LEARNING EFFICIENT DEEP FEATURE EXTRACTION FOR MOBILE OCULAR BIOMETRICS

A Dissertation
IN
Electrical and Computer Engineering
and
Telecommunications and Computer Networking

Presented to the Faculty of the University
of Missouri–Kansas City in partial fulfillment of
the requirements for the degree

DOCTOR OF PHILOSOPHY

by
NARSI REDDY

M.S., University of Missouri-Kansas City, Kansas City, USA, 2020

Kansas City, Missouri
2020

LEARNING EFFICIENT DEEP FEATURE EXTRACTION FOR MOBILE OCULAR BIOMETRICS

Narsi Reddy, Candidate for the Doctor of Philosophy Degree

University of Missouri–Kansas City, 2020

ABSTRACT

Ocular biometrics uses physical traits from eye regions such as iris, conjunctival vasculature, and periocular for recognizing the person. Ocular biometrics has gained popularity amongst research and industry alike for its identification capabilities, security, and simplicity in the acquisition, even using a mobile phone's selfie camera. With the rapid advancement in hardware and deep learning technologies, better performances have been obtained using Convolutional Neural Networks(CNN) for feature extraction and person recognition. Most of the early works proposed using large CNNs for ocular recognition in subject-dependent evaluation, where the subjects overlap between the training and testing set. This is difficult to scale for the large population as the CNN model needs to be re-trained every time a new subject is enrolled in the database. Also, many of the proposed CNN models are large, which renders them memory intensive and computationally

costly to deploy on a mobile device. In this work, we propose CNN based robust subject-independent feature extraction for ocular biometric recognition, which is memory and computation efficient. We evaluated our proposed method on various ocular biometric datasets in the subject-independent, cross-dataset, and cross-illumination protocols.

APPROVAL PAGE

The faculty listed below, appointed by the Dean of the School of Graduate Studies, have examined a dissertation titled "Learning Efficient Deep Feature Extraction For Mobile Ocular Biometrics," presented by Narsi Reddy, candidate for the Doctor of Philosophy degree, and hereby certify that in their opinion it is worthy of acceptance.

Supervisory Committee

Reza Derakhshani, Ph.D., Committee Chair
Department of Computer Science & Electrical Engineering

Cory Beard, Ph.D., Co-discipline Advisor
Department of Computer Science & Electrical Engineering

Ghulam Chaudhry, Ph.D.
Department of Computer Science & Electrical Engineering

Zhu Li, Ph.D.
Department of Computer Science & Electrical Engineering

Praveen Rao, Ph.D.
Department of Computer Science & Electrical Engineering

CONTENTS

ILLUSTRATIONS

TABLES

ACKNOWLEDGEMENTS

Foremost, I would like to thank my advisor Dr. Reza Derakhshani for providing my teaching, guidance, and support through my graduate studies. From the start, he gave me the freedom to experiment with different ideas while providing me with advice if I needed it.

I am very grateful to my thesis committee members Dr. Cory Beard, Dr. Praveen Rao, Dr. Ghulam Chaudhry, and Dr. Zhu Li, for being flexible in accommodating me. I want to thank Dr. Ajita Rattani, for supporting me through knowledge and guidance in my research and writing.

I have been very fortunate to work with many great people at Eyeverify (dba ZOLOZ), who guided me in understanding the biometrics industry's workings and requirements. I would also like to thank all friends, especially Daniel Lopez, Ahmad Mohammad, Ala-Addin Nabulsi, Jesse Lowe, and Mark Nguyen, to provide me technical and moral support through my research.

Lastly, I would like to thank my family for supporting me through my endeavors.

CHAPTER 1

INTRODUCTION

With the increasing popularity of mobile devices, biometrics plays a vital role in protecting personal information and data. Several efforts have been made in developing biometric authentication systems using physical traits such as the face, ocular(eye regions), and fingerprint for mobile devices [6, 7]. In specific, ocular biometrics in the visible spectrum has gained attraction in mobile devices as it can be easily acquired by using front-facing RGB ”selfie” cameras. Ocular patterns in visible light include vascular arcades seen on the white of the eye, eyebrows, and the periocular region (mostly skin texture and wrinkles) encompassing the eye, as shown in Figure 1.

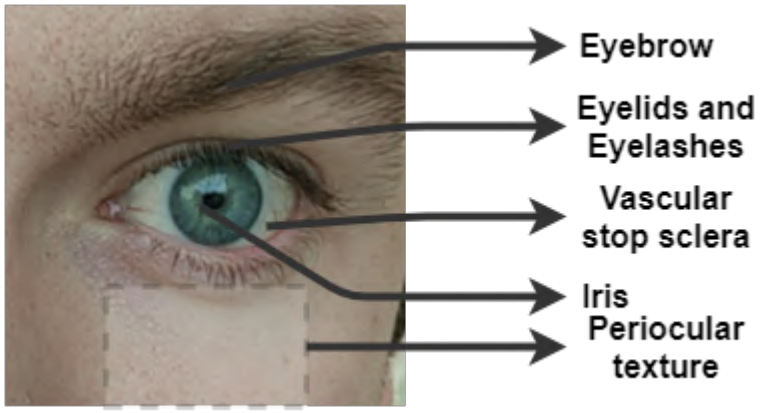


Figure 1: An ocular image labeled with vasculature pattern, eyebrow, eyelids, eyelashes, and periocular skin texture.

1.1 Mobile Biometrics System

A biometric system, in general, uses physiological or behavioral characteristics to recognize an individual. Physiological biometrics systems use the shape or structure of the individual body, such as the face, fingerprint, iris, palm veins, periocular, palm print, and ear shape. When it comes to behavioral biometric systems, an individual is recognized based on the behavior patterns such as gain, keystrokes, signature, facial expressions, and voice.

The first stage of any biometric system is **data acquisition**. In the mobile ocular biometrics system, the data - which in this work are eye images - are acquired using the front-facing ”selfie” camera. Then, in the **preprocessing** stage, the region of interest (ROI) extraction and (or) illumination correction is applied to the acquired eye image samples. Then the preprocessed sample is used to **features extraction** for matching.

Matching in the biometric system is a two-step process: Enrollment and recognition. In the **Enrollment process**, the extracted features of an individual are stored in the database as templates. These databases can be local to the device or in a secure cloud service. When it comes to mobile biometric systems, for security and protecting an individual's biometric data, the templates are store locally with encryption. **Recognition process** takes place when an individual needs to gain access or claim one's identity. In this step, the features extracted from the eye sample are matched with templates in the database to verify the individual's identity.

A biometric system operates in either identification mode or verification mode in the recognition process. In **Identification mode**, given individual's biometric features is

with templates of many users in the database to identify if the individual is in the system. This type of model is useful in large group operations such as multi-user access systems or office buildings. Whereas in the **verification mode**, extracted features of an individual are matched with their own templates in the database to verify their identity. For security and authentication efficiency reasons, the templates are stored locally in mobile biometrics. So, the matching process operated in verification mode.

Figure 2 shows the overall outline of the mobile biometric system pipeline. An individual can gain access or verify identity by matching extracted features from their eye samples with the stored templates in the local database.

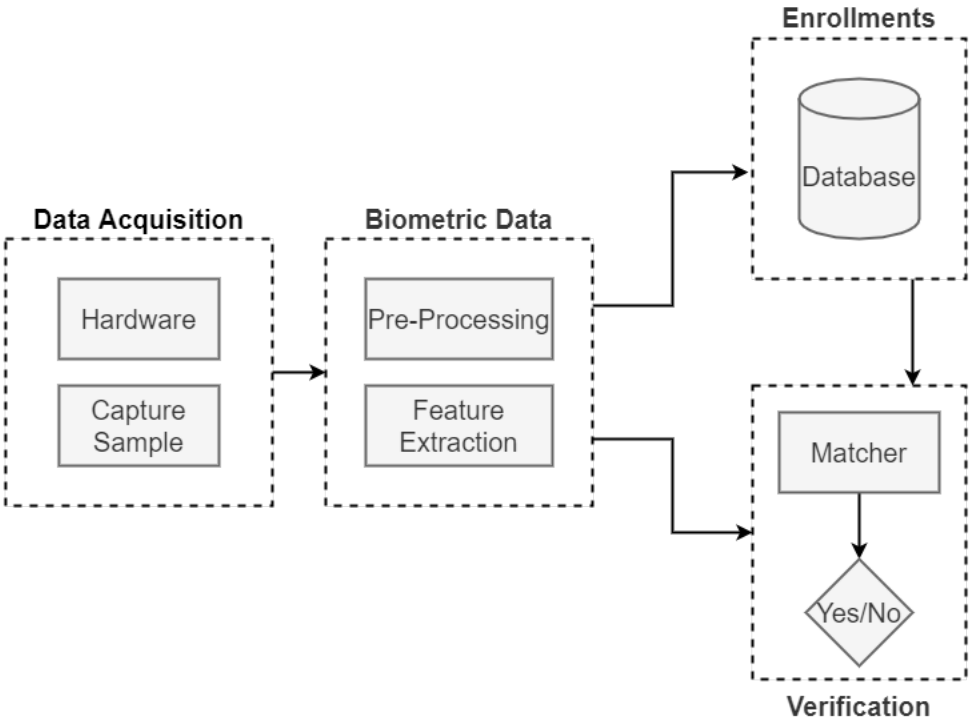


Figure 2: Mobile biometrics system pipeline.

1.2 Problem Statement

Detailed surveys of ocular and periocular biometrics in the visible spectrum have been provided by Alonso-Fernandez, and Bigun [8], Rattani and Derakhshani [9] and Kumar and Seeja [10]. Earlier studies have mostly proposed the use of handcrafted features such as histograms of oriented gradients (HOG) and local binary patterns (LBP), along with simple distance metrics for matching such as Euclidean distance or learning-based methods such as Support Vector Machines (SVM) [11].

With the recent advances deep-learning focused advances in software and hardware, Convolutional Neural Networks (CNN) and alike are now capable of real-time biometric feature extraction and matching, even on mobile and embedded hardware [12]. Most of the earlier ocular recognition work used CNNs under subject-dependent evaluation protocols, where the subjects overlap between the training and test sets, which may over-estimate performance and generalizability. However, recently, researchers have been focusing on CNN-based methods for ocular recognition in the unconstrained environment and under the subject-independent protocol [13, 14].

Subject-dependent vs Subject-independent: Many studies proposed [15–18] using multi-class classifiers for recognition and obtained state of the are performance. However, multi-class classifiers can operate in the only subject-dependent protocol, which creates a problem when a new subject's identity needs to be added, making the model to re-train. Therefore, a model needs to train in a subject-independent protocol, where subject identities do not overlap between the training and test set.

Unconstrained Environments: The challenges of unconstrained environments

may lead to substantial variations in the samples due to factors such as lighting conditions, distance, motion blur, glasses and hair occlusions, front-facing imaging sensor and optic aberrations (including smudged lenses), and other imaging issues such as inaccurate white balance and exposure metering. Many recent techniques proposed data augmentation [18], and enhanced region of interest (ROI) detection methods [4] for better performance in an unconstrained environment, but proposed methods are large and (or) require large computational requirement to be able to implement in mobile devices.

Computational Efficiency: For a better user experience in a mobile device, a biometric system, along with being secure, it must be fast. For a deep learning-based system to be fast, it must be efficient computationally and lower memory size. Many of the deep learning based purposed methods used architectures such as VGG [19], ResNet [20], and Inception-net [21], which are large [22] and also computationally slower [23] to implement in the mobile biometric system.

This work proposes a robust deep learning-based model that works in a subject-independent unconstrained environment and computationally efficient for mobile devices.

1.3 Contributions

Contributions made in this work are as follows:

1. Conducted, a large scale evaluation of different CNN based architectures to evaluate the performance in terms of accuracy vs. size vs. speed. Based on the finding, we propose two incrementally updated models.

2. OcularNet: First proposed model is a collection of small CNN models using local

patches from the eye images. As the proposed method is a patch-based technique, one can extract features based on the region's availability in the eye image.

3. OcularNet-v2: To extract eye region patches, OcularNet depends on the shelf eye region of interest(ROI) detector and localization models. If these fail to localize the eye region accurately, OcularNet fails. To overcome this, in OcularNet-v2, we proposed to use a built-in ROI detector that is trained along with an efficient single CNN model for better feature extraction.

4. Finally, we propose a new large ocular dataset in the visible spectrum (LOD-V) for training robust deep learning models and showing significant improvements in performance for the OcularNet-v2 model.

1.4 Thesis Outline

Chapter 2 provides a literature review on the existing works in ocular biometrics methods from early handcrafted features based methods to current deep learning based methods in subject-dependent and subject-independent evaluation protocols.

In **Chapter 3**, a brief explanation of how we calculate computation complexity and size of the model to better understand and design efficient CNN models for mobile ocular biometrics.

In **Chapter 4**, analyzed popular CNN architectures to evaluate their efficient subject-independent performance in mobile ocular biometrics. We evaluate how well models perform when trained from scratch and after fine-tuning. Based on the theoretical and practical knowledge gained from comparing different architectures, we proposed our custom

model designed to be computationally efficient while achieving comparable performance.

Chapter 5 propose our first version of the OcularNet model, a patch-based CNN architecture for mobile ocular biometrics. We showed that the proposed OcularNet model, which was smaller than the popular ResNet architecture model, outperforms on multiple datasets in subject-independent and cross-dataset evaluations.

In **Chapter 6**, based on the fail cases from the OcularNet, we propose multiple improvements for our model, OcularNet-v2. This method consists of a much efficient feature extraction model, which is trained along with an ocular ROI detector. While being very computationally efficient, the proposed method can easily generalize to other datasets, including those captured in near-infrared illumination.

In **Chapter 7** proposes a new large ocular dataset in visible lighting (LOD-V), which consists of more than 772 unique subjects with over $200K$ eye samples. We show that by training OcularNet-v2 on the LOD-V dataset, which has $3.8\times$ more unique subjects, can improve performance significantly.

In **Chapter 8**, a conclusion is provided, then we identify challenges remaining and provide a future path.

CHAPTER 2

PREVIOUS WORK

Before the the rise of today's popular CNN based methods, studies proposed using feature extraction techniques such as histograms of oriented gradients (HOG) [24], color histograms [25], local binary patterns (LBP) [26], local phase quantization (LPQ) [27], binarized statistical image features (BSIF) [28], phase-only correlation (POC) [29] and maximum response sparse filter [30], along with simple distance based match metrics such as Euclidean distance or cosine similarity for ocular recognition. Others proposed traditional learning based methods such as Support Vector Machines (SVM) [11], and shallow neural networks [16, 30]) for ocular recognition.

Below we review notable CNN-based ocular recognition methods under subject-dependent and subject-independent protocols.

2.1 Subject-dependent Evaluation

Raghavendra et al. in [30] combined texture features extracted by Maximum Response (MR) filters with deeply coupled autoencoders for deriving features from ocular images along with a softmax based classifier trained on VISOB [3] dataset. The proposed method outperformed others submitted to the ICIP-2016 mobile ocular biometrics competition [3]. Ahuja et al. in [31] proposed a CNN model, VisobNet, with a

reported error rate of less than a 1% when trained and tested on the VISOB and MICHE-II [32] datasets, significantly outperforming their own method based on Scale Invariant Feature Transform(SIFT) [33] keypoint descriptor. Using the VISOB dataset, Rattani et al. in [15] evaluated the transfer learning technique where the CNN models pre-trained on ImageNet [12] were fine-tuned for ocular recognition. Alahmadi et al. [17] used a CNN model pre-trained on ImageNet for feature extraction, followed by a sparse classifier (SA-SRC) for classification, trained and tested on VISOB dataset. Table-1 summarises deep learning techniques proposed in only close-set protocols.

Table 1: Existing deep learning based techniques proposed for ocular recognition in an only subject-dependent protocol.

Method	**Summary**	**Datasets**
MR Filter [30]	Autoencoder with Softmax for feature size reduction.	VISOB [3]
VisobNet [31]	Supervised learning with softmax classifier.	VISOB, MICHE-II [32]
Fine-tuned VGG [15]	Transfer Leanring on pre-trained CNN mdoels.	VISOB
ConvSRC [17]	Pre-trained CNN model with sparse classifier	VISOB

2.2 Subject-independent Evaluation

Nie et al. [11] proposed an unsupervised convolutional radial basis function (C-RBF) network for efficient feature extraction from ocular images. The authors also proposed a supervised metric learning technique to achieve better matching accuracy on the UBIPR [34] dataset when compared to conventional methods such as cosine similarity and Euclidean distance. Zhao et al. [13] used explicit semantic information such as gender and left/right eye in addition to the ocular image and trained a CNN model for feature extraction. The matching was performed using subject-independent and cross dataset scenarios

using a Bayesian scheme [35]. In the visible spectrum, the model was trained on UBIPR and tested on UBIRIS-V2 [**?**] and FRGC [36]. For evaluation in the infrared spectrum, the model was trained on a subset of FOCS [37] and tested on CASIA.v4-distance [38] and the remaining FOCS dataset. Garg et al. [39] proposed a modified triplet loss for training CNN models to learn heterogeneity aware robust features for an unconstrained environment. They reported significant improvement in subject-independent and cross dataset analysis using VISOB, CSIP [40], and IITD IMP [41] datasets. Tiong et al. [14] proposed a dual-stream CNN model with shared weights to extract features from the original RGB eye image along with local binary-coded patterns (OC-LBCP) image. OC-LBCP combines local binary patterns (LBP) and local ternary patterns (LTP), which increases the invariance to confounding factors such as noise and illumination variations. They performed the subject-independent evaluation on their new ethnic-ocular dataset and cross dataset evaluation on the UBIPR dataset [34].

Reddy et al. [42] proposed a fully unsupervised autoencoder-based CNN model for feature extraction and evaluated using a cross-dataset protocol. To learn robust features in an unsupervised manner during training, the authors proposed a loss function that reduces the difference between the encoded features of two randomly augmented images from a single original image, along with auto-encoders for input image reconstruction loss. The model was then trained on all the images from UBIRIS-V2, UBIPR, and MICHE-II datasets and tested on the VISOB dataset.

Proenca et al. [18] proposed a custom data augmentation pipeline where the ocular ROI confined within the eyelids is replaced with a different subject randomly. So the CNN

model focuses only on the periocular region for feature extraction and avoids features inside the eye. In this study, experiments conducted on the UBIRIS-v2 dataset show significant performance improvement.

Zhao et al. [4] proposed a supervised learning method to produce a mask for extracting ocular features from eyebrows and eye regions. A semantic segmentation model with only $0.1M$ parameters learned from 100 images is used to generate the ROI mask. All the experiments were carried out using a subject-independent evaluation on UBIRIS-V2, UBIPR, FRGC, FOCS, and CASIA.v4-distance datasets.

Table 2 summarizes the above mentioned deep learning techniques for ocular recognition in an unconstrained environment evaluated using subject-independent and cross-dataset analyses.

Table 2: Existing deep learning based techniques proposed for ocular recognition using subject-independent, subject-dependent, and cross-dataset analyses. *CD stands for cross dataset analysis.*

Method	**Summary**	**Datasets**	**Data Protocol**
Metric Learning [11]	Unsupervised convolutional RBF for feature extraction with supervised metric learning.	UBIPR [34]	subject-independent
Semantics-Assisted CNN [13]	Training feature extraction model with additional semantic information.	UBIPR [34], UBIRIS-V2 [**?**], FRGC [36], FOCS [37], and CASIA.v4-distance [38]	subject-independent (also CD)
Heterogeneity Aware Deep Embedding [39]	Custom triplet loss for robust feature extraction.	VISOB [3], CSIP [40], and IITD IMP [41]	subject-independent and subject-dependent
Explicit Critical Regions Attention [4]	Feature extraction from eyebrow and eye region using semantic segmentation mask.	UBIRIS-V2 [**?**], UBIPR [34], FRGC [36], FOCS [37], and CASIA.v4-distance [38]	subject-independent and subject-dependent
Dual Stream CNN [14]	Shared weights CNN model for feature extraction and OC-LBP image.	UBIPR [34] and ethnic-ocular dataset	subject-independent (also CD)
Deep-PRWIS [18]	Proposed a data augmentation pipeline by interchanging ocular region of a sample with a different subject's.	UBIRIS-v2 [**?**]	subject-independent and subject-dependent
Encoded Feature Loss [42]	Custom autoencoder for robust feature extraction using unsupervised learning.	VISOB [3], UBIRIS-V2 [**?**], UBIPR [34] and MICHE-II [32]	subject-independent (also CD)

CHAPTER 3

CALCULATING CNN MODELS COMPUTATIONAL EFFICIENCY

As the input data is image-based in ocular biometrics, CNN-based models are the most commonly used deep learning techniques for feature extraction and matching. This chapter provides a brief explanation of different layers in the CNN model, followed by how we calculate the size and computational cost of a CNN model.

3.1 Basic Building Blocks Of CNN Model

Essentially a CNN model consists of Convolutional and fully connected (FC) layers, along with non-linear activation and pooling layers. In this section, a brief explanation of for each layer in the CNN model,

Fully connected layer connects every input feature to every output feature in a given layer by performing matrix multiplication between input features and learnable weight matrix as shown in Figure 3. Let, x be the input feature vector of $1 \times F_{in}$ size and y be the output feature vector of size $1 \times F_{out}$. W_i be the weight matrix of i^{th} layer with $F_{in} \times F_{out}$ size. Then, fully connected (FC) layer is implemented as follows,

$$y = W_i * x \tag{3.1}$$

In **Convolutional Layer** output features are generated by convolving multiple learnable filter (or kernels) onto input features as shown in Figure 4. Let x be C_{in} input

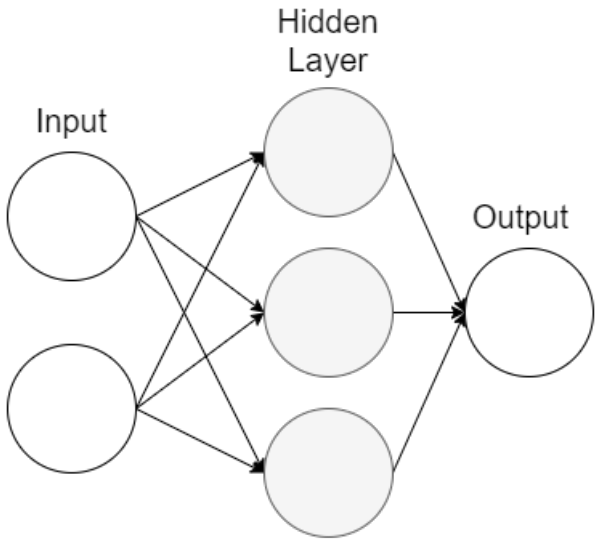


Figure 3: Simple fully connected layer with one learnable layer.

features with 2D spatial size of $m \times n$ and y be the generated C_{out} out put features with 2D spatial size of $m' \times n'$. Here convolution operation is given as,

$$y_{ij} = w_{ij} \circledast x_i \tag{3.2}$$

Where y_{ij} is j^{th} output feature generated when i^{th} input feature, x_i, is convolved with kernel w_{ij} of size $k \times k$.

Then to generate one out of C_{out} output feature of size $m' \times n'$, we need C_{in} number of kernels of size $k \times k$. So, total of kernels requires to generate C_{out} number of features is given as $C_{in} \times C_{out}$. The convolution output spatial size, m' n', depend on the stride s, the kernel size k, and any padding p applied to the input features as follows,

$$m' = (m - k + 2p)/s + 1, n' = (n - k + 2p)/s + 1 \tag{3.3}$$

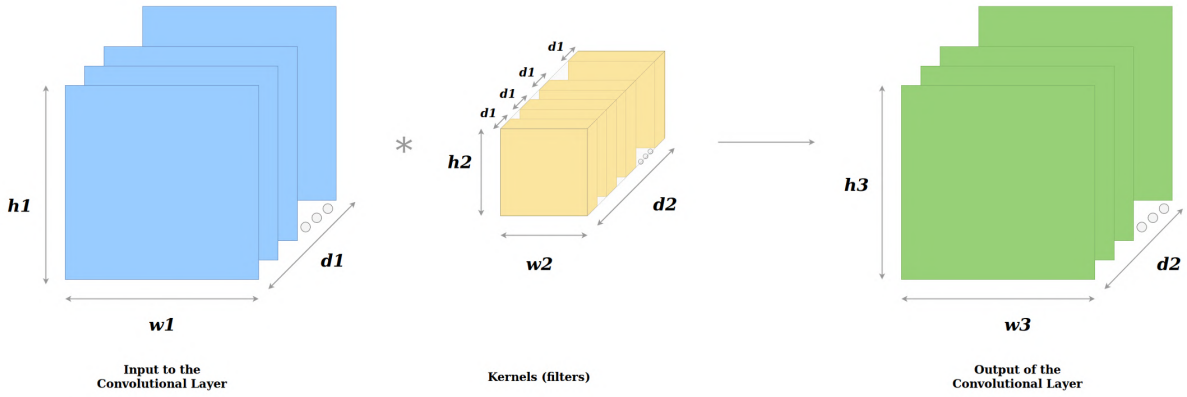


Figure 4: Convolutional Layer [1]

In neural networks, **Activation layers** are used to introduce non-linearity into the model to execute non-trivial operations. The most commonly used activation functions in neural networks are hyperbolic tangent (TanH) and Sigmoid. However, when it comes to CNN models, ReLU - Rectified Linear Unit - is the most popular activation function used for its simplicity in implementation, as shown in equation 3.4. Unlike TanH and Sigmoid, ReLU avoids the vanishing gradient problem, making it easier to building deeper models.

$$y = max\,(0, x) \tag{3.4}$$

Pooling layer reduces the spatial size of the input features to reduce the number of parameters and computational cost of the CNN model, which in turn helps reduce the model overfitting to the input data. The most commonly used pooling layer in CNN models is max pooling, which outputs maximum values from the input features in spatial window k. Along with max pooling, average pooling is also used in CNN models mainly to perform global pooling on the final convolutional layer to reduce the feature size in

models such as ResNet and MobileNet.

3.2 How To Calculate Models Computational Cost And Size

It is essential to understand how we calculate the evaluated models' size and computational complexity to build efficient deep learning models for mobile ocular biometrics. This is because a large model with high computational cost requires more memory and increases execution latency. In contrast, user experience requires fast and battery-friendly biometric authentication, which could be triggered frequently, such as unlocking the phone.

As the input data is image-based, CNN-based models are most commonly used in ocular biometrics. Essentially a CNN model consists of Convolutional and fully connected layers, along with non-linear activation, pooling, and normalization layers. The **size** of the models is calculated by taking the number of learnable parameters in the CNN models. The **computational cost** is calculated as the number of multiply-addition (MAdd) operations present in all learnable layers such as convolutional and fully connected layers [43, 44].

For a normal convolutional layer with $k \times k$ convolution with c_{in} input feature channels and c_{out} output feature channels, the total number of parameters is calculated as:

$$k \times k \times c_{in} \times c_{out} \tag{3.5}$$

MAdd operation with input features spatial size of $m \times n$, assuming stride $s = 1$, is calculated as:

$$k \times k \times c_{in} \times c_{out} \times m \times n \tag{3.6}$$

In a fully connected layer with F_{in} number of input features and F_{out} number of output features, the total number of parameters is calculated as:

$$F_{in} \times \ F_{out} \tag{3.7}$$

the the number of MAdd operations are given as

$$F_{in} \times \ F_{out} \tag{3.8}$$

To reduce the size and computational cost of models, methods such as parameter pruning [44], lower bit quantization [45], squeezed convolutional networks, and the use of separable convolutions [43, 46, 47] are proposed.

From the above equation 6.4 and equation 6.5, in a standard conventional layer, to extract one new feature $k \times k$ convolution is applied on all the input features. In the case of separable convolutions, only one $k \times k$ convolution is performed for each input, generating the same number of channels as the input (i.e., $c_{out} = c_{in}$), then is generally followed by 1×1 standard convolutional layer for feature extraction. For separable convolutions, the number of parameters and MAdd operations is calculated as:

$$k \times k \times c_{in} \times 1 + 1 \times 1 \times c_{in} \times c_{out} \tag{3.9}$$

$$\begin{aligned} &m \times n \times k \times k \times c_{in} \times 1 + m \times n \times 1 \times 1 \times c_{in} \times c_{out} \\ &= m \times n \times (k \times k \times c_{in} \times 1 + 1 \times 1 \times c_{in} \times c_{out}) \end{aligned} \tag{3.10}$$

To understand how computationally efficient are separable convolutional, let consider a concrete example. Let 64×64 be the input size with 32 features and using 3×3 convolutional layer to generate 128 feature channels, then for standard convolutional

layer, by substituting these number in equation-3.5 and equation-3.6, gives $36,864(= 32 \times 128 \times 3 \times 3)$ number of parameters with $151M(= 64 \times 64 \times 32 \times 128 \times 3 \times 3)$ of MAdd operations. In the case of separable convolutions by substituting above values in equation-3.9 and equation-3.10, we obtain $2,336(= [3 \times 3 \times 32] + [32 \times 128])$ number of parameters with $9.5M(= [64 \times 64] \times [3 \times 3 \times 32 + 32 \times 128])$ MAdd operations. From the above example, we can see a considerable reduction in the number of parameters and MAdd operations making CNN models designed with separable convolutions more computationally efficient.

CHAPTER 4

CASE STUDY OF DEEP LEARNING MODELS IN OCULAR BIOMETRICS

4.1 Introduction

Deep learning has provided significant improvements in many applications, such as image classification, object detection, and segmentation [48]. With the mobile technology revolution and wide-scale integration of biometric technologies for user authentication, deep learning solutions have successfully ported into mobile phones for accurate user authentication [49].

This chapter provides a comparative analysis of different deep learning models in terms of their efficacy in biometric user authentication on mobile devices. To this aim, we evaluate existing deep learning architectures such as VGG [19], ResNet [20], DenseNet [50] and mobile device centric architectures such as MobileNet-v1 [43], MobileNet-v2 [47] and NasNet-mobile [2] in terms of their matching performance and computational cost on mobile ocular biometrics dataset. In addition, we present and benchmark our custom compact deep learning model and show it has comparable performance to existing deep learning models while being competitive in model size and computational cost.

In summary, the contribution of this chapter are as follows:

1. Comparative evaluation of deep learning architectures such as VGG [19], ResNet [20], DenseNet [50] and mobile device based architectures such as MobileNet-v1 [43], MobileNet-v2 [47] and NasNet-mobile [2] in terms of their performance and cost

for operation on mobile devices.

2. Also proposes a custom deep learning architecture that is better suited for mobile biometrics due to its compact size, lower computational cost, and competitive accuracy compared to existing models.

The rest of this chapter is organized as follows: Section 2 provides details on different deep learning CNN models used in this study. Dataset and experimental protocol are discussed in section 3. Experimental results on ocular biometrics case studies are presented in section 4. Conclusions are drawn in section 5.

4.2 Convolutional Neural Network Models

The deep learning models evaluated in this study are mostly CNNs pre-trained on a large scale ImageNet [12] dataset comprising 1.2 million training images, a standard when it comes to large-scale image classification. Following, we discuss the selected pre-trained models and our proposed custom model in terms of their architecture, parameters, and the number of MAdd operations, which are compiled into a Table-3.

1. VGG: The VGG [19] architecture was introduced by the Visual Graphics Group research team at Oxford University. The architecture consists of sequentially stacked 3×3 convolutional layers with intermediate max-pooling layers followed by a couple of fully connected layers for feature extraction. Usually, VGG models have 13 to 19 layers. Our experiments used VGG-19 as our test model, which has a $140M$ number of parameters with $19.5G$ MAdd operations.

2. ResNet: ResNet [20] is a short form of residual networks based on the idea of

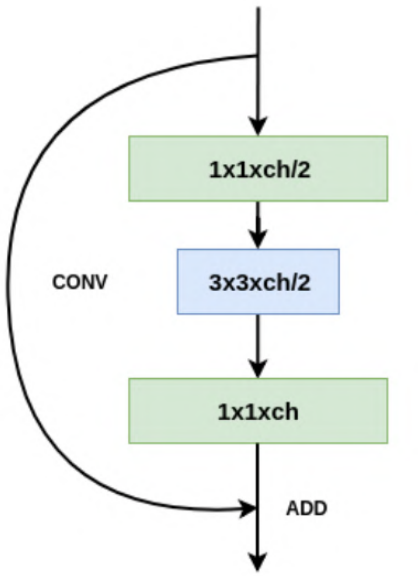


Figure 5: Building blocks of ResNet architecture

"identity shortcut connection" where input features may skip certain layers as shown in the Figure 5. This study uses ResNet-50, which has $23.5M$ parameters with $4G$ MAdd operations.

3. DenseNet: DenseNets [50] are inspired by residual networks, where all the previous layers' features are transferred to the current layer, as shown in Figure 6. Apart from tackling the vanishing gradients problem, this architecture also strengthens feature propagation and feature reuse while reducing the required parameters. This study uses densenet-121, which has $3.2M$ parameters with $2.8G$ MAdd operations.

4. MobileNet-V1: As can be seen from the Table 3, the architectures such as VGG and ResNet, although achieved very high accuracies in the Imagenet dataset, are either large in size or require a lot of MAdd operations. Therefore, these architectures may not be efficient for implementation on mobile devices. MobileNet-v1 [43] is one of the most

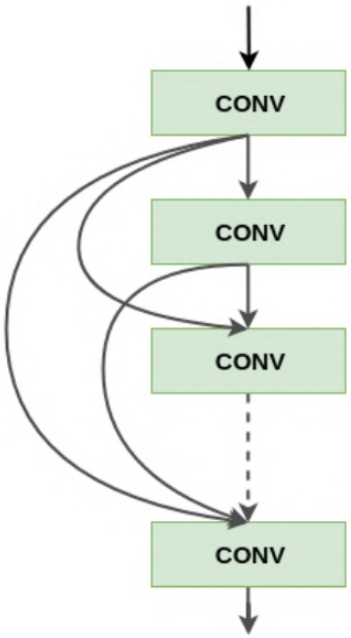


Figure 6: Building blocks of DenseNet architecture

popular mobile-centric deep learning architectures, which is small in size and computationally efficient while achieving high performance. The main idea of MobileNet is that instead of using regular 3×3 convolution filters, depth-wise separable 3×3 convolution filters are followed by 1×1 convolutions. While achieving the same filtering and combination process as a regular convolution, the new architecture requires fewer operations and parameters. This study uses MobileNet-v1 with 0.5x and 1x channels multiplier with input size of 224×224 for testing. These models are denoted as $MobileNet\_V1\_0.5\_224$ and $MobileNet\_V1\_1.0\_224$, respectively, in Table 3.

5. MobileNet-V2: The newer version of MobileNet architecture combines depth-wise separable 3×3 convolution with inverted ResNet architecture. In ResNet architecture, as shown in Figure 5, the 3×3 convolution is performed on the reduced number of

Table 3: Comparison of extracted feature size, number of parameters and number of MAdd operations for each CNN model evaluated in this study. The number of parameters and MAdd operations are the dominant factors in evaluating a model's size and computational cost.

Model	Feature Size	Parameters	No. of MAdd
VGG - 19	4096	140M	19.6G
ResNet - 50	2048	23.5M	4G
DenseNet - 121	1024	7M	2.8G
MobileNet_v1_1.0_224	1024	3.2M	568M
MobileNet_v1_0.5_224	512	819K	149M
MobileNet_v2_1.0_224	1280	2.2M	300M
MobileNet_v2_0.5_224	1280	688K	96M
NasNet-mobile	1056	4.2M	567M
Proposed Model	256	672K	256M

channels whereas in MobileNet-v2 [47] architecture the 3×3 convolution layer is replaced with depth-wise separable 3×3 convolution layer and increased number of channels, as shown in Figure 7. if ch_{in} are the number of feature channels provided as an input to the residual layer, resnet architecture extracts features at 3×3 convolution on half of the input feature channels i.e., $ch_{in}/2$. Whereas in the case of MobileNet-V2, the feature channels are increased by an expansion factor t i.e., $ch_{in} * t$. Experiments are conducted on MobileNet-v2 with 0.5x and 1x channels multiplier with input size of 224×224 for testing. These models are denoted as *MobileNet_V2_0.5_224* and *MobileNet_V2_1.0_224*, respectively, in Table 3.

6. NasNet-Mobile: Unlike other models presented in this paper, this model is not hand-designed. Instead, a reinforcement learning technique known as AutoML [51] was used to generate this model and specifically designed to perform well over Imagenet [12]

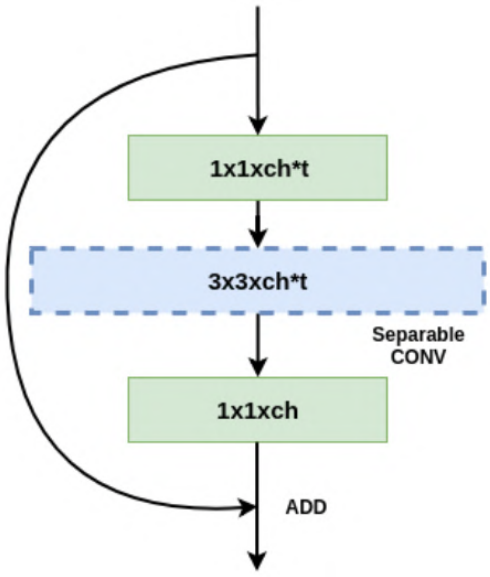


Figure 7: Building blocks of MobileNet-V2 architecture.

dataset. AutoML searches for the best convolutional layer (or âcellâ) on a small dataset and then transfers the block to a larger dataset. By changing the number of the convolutional cells and the number of filters in the convolutional cells, different versions of NASNet [2] were developed. In this paper, we considered the smallest of all the NasNets, called NASNet-mobile, targeted for mobile devices.

7. Proposed Model: Our custom model is based on MobileNet-v2 architecture, as shown in Figure 7, where we keep expansion factor $t = 1$. Table 4 show the complete architecture of the proposed model. As it can be seen from Table 4, the spatial resolution is dropped 4X with-in the first two layers to reduce the computational complexity. The convolution layers $conv2$, $conv3$ and $conv4$ use the residual module shown in Figure 9. At $conv2b$ and $conv3b$ same module is used but without the skip connection to reduce the spatial resolution by half and to increase the number of feature channels. The proposed

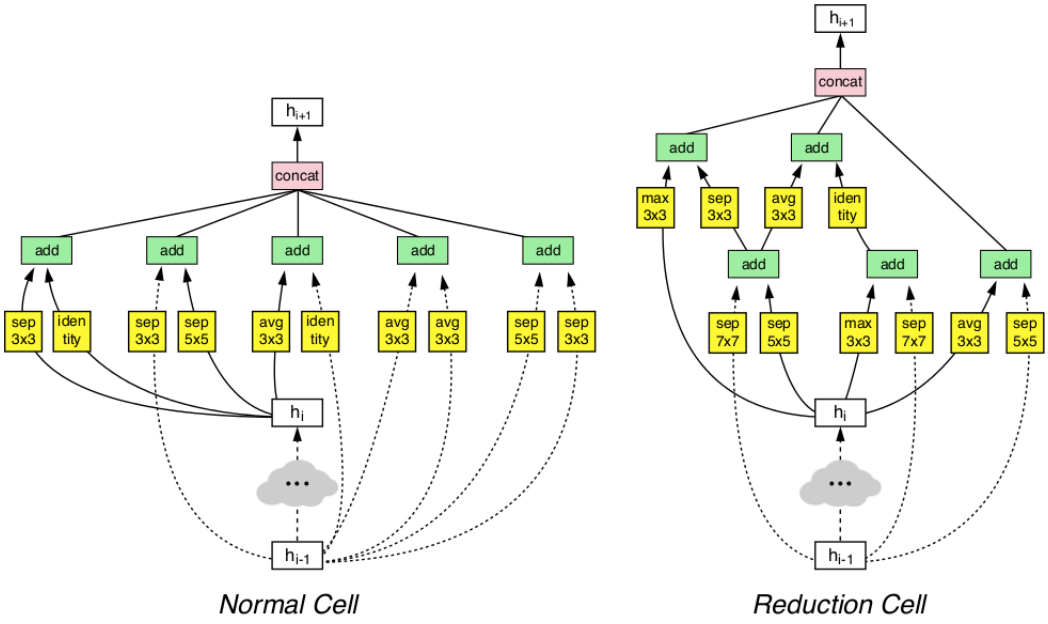


Figure 8: Building blocks of NasNet architecture. Normal Cell: For feature new features extraction, and Reduction Cell: For feature extraction and spatial size reductions. **Figure reference [2]*

model has only $672K$ parameters and $256M$ MAdd operations. The feature size and number of parameters in our proposed model are the least among all the models (see Table 3).

4.3 Experimental Setup

Dataset: VISOB [3] dataset consists of ocular images [9] from over 550 healthy subjects. This publicly available dataset was collected using front-facing cameras of different mobile devices (iPhone 5s, Samsung Note 4 and Oppo N1) under varying lighting conditions (office, daylight, and dim indoors). Participants' data were collected in two visits, visit 1, and visit 2, 2 to 4 weeks apart. During each visit, participants took a selfie

Table 4: The proposed custom deep learning models are based on MobileNet-V2 architecture with expansion factor $t = 1$. The input and output shapes are described in height x width x channels.

Input Shape	**Layer**	**Output Shape**	**Parameters**	**MAdd Operations**
160x240x1	ConvBNReLU(5x5x64, s=2)	80x120x64	1792	15M
80x120x64	MaxPooling(3,2)	40x60x64	-	-
40x60x64	3 X (Residual[t=1, ch=64, s=1])	40x60x64	28,032	60M
40x60x64	ConvBNReLU(1 X [t=1, ch=128, s=2])	20x30x128	26,752	30M
20x30x128	3 X (Residual[t=1, ch=128, s=1])	20x30x128	105,216	61M
20x30x128	ConvBNReLU([t=1, ch=256, s=2])	10x15x256	102,656	30M
10x15x256	3 X (Residual[t=1, ch=256, s=1])	10x15x256	407,040	60M
10x15x256	Global Average Pooling	1X1X256	-	-
	Total		671,488	256M

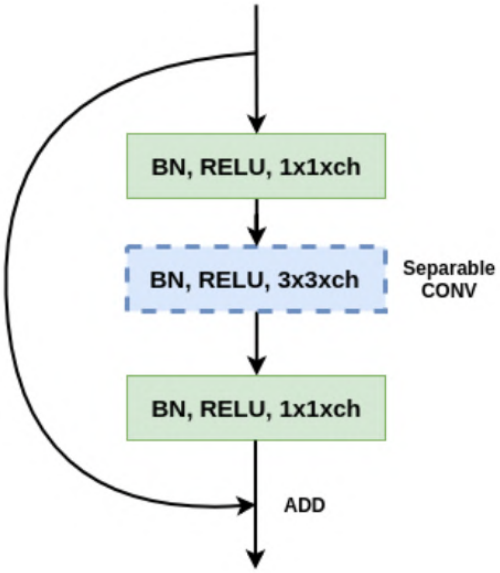


Figure 9: Custom layer module is similar to MobileNet-V2 architecture with expansion factor $t = 1$ and normalization and activation before each convolutional layer.

like captures in two different sessions (session 1 and session 2) about 10 to 15 minutes apart, under all lighting conditions and using all the three devices. From the collected data, eye crops were generated using Viola-Jones based eye detector, and the cropped eye images were resized to 160×240 pixel resolution. Variations such as motion blur, specular reflections, and different lighting conditions are captured in this dataset. Figure 10 shows example ocular images from VISOB dataset.

In our experiments, we divided VISOB dataset into three sets as follows:

1. ***DATA-A:*** This set consists of ocular images from 200 participants from Visit 1 for all the devices, all lighting conditions, and sessions. This subset is used for training all deep learning models (discussed in 4.2), and it consists of $39,732$ images from the left and the right ocular regions. The models trained using this subset are used

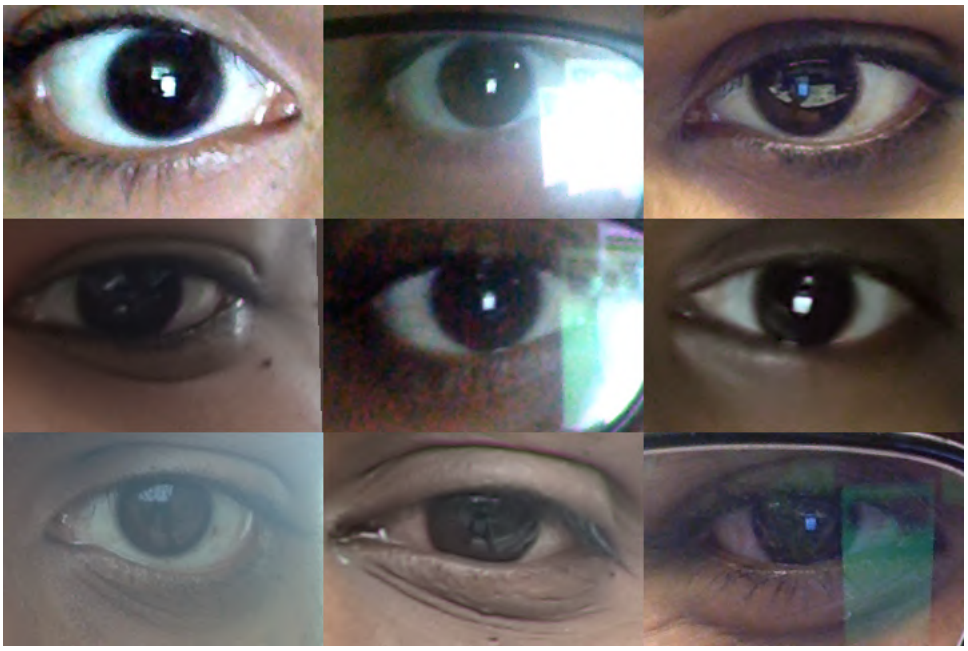


Figure 10: Example of ocular images from VISOB dataset [3]

for ocular feature extraction.

2. ***DATA-B:*** This subset consists of remaining 350 participants with $55,314$ images from the left and right ocular regions from Visit 1, which were not used in training or fine-tuning the deep learning models. This subset is used to evaluate the subject independent, subject-independent verification.

3. ***DATA-C:*** This subset consists of all the 295 participants from visit 2 with $63,089$ images. This subset consists of 92, overlapping participants with DATA-A. This subset is used to test the subject-independent verification with few overlaps in training and validation.

For the purpose of this study, We flipped the right ocular images horizontally and

considered these images as belonging to unique subject identities. This way number of unique subjects are doubled in all the dataset, and all the results were computed for the left ocular images.

Training Protocol: We trained our proposed model using Adam optimizer [52] with the learning rate of $lr = 0.001$ for 150 epochs with early stopping and with the batch size of 32 on DATA-A using softmax with categorical cross-entropy as a loss function. The pre-trained models (introduced in section 4.2) were fine-tuned for ocular recognition using transfer learning. For transfer learning of the pre-trained models, the classification layer is changed from 1000 outputs to 400 outputs associated with the number of DATA-A subjects. The transfer learning process is divided into two stages, as follows:

1. Stage 1: All the layers except the final classification layer (consisting of 400 outputs) are set as non-trainable, and the whole network was trained for 10 epochs. We used Adam optimizer with $lr = 0.001$ and with batch size of 32.

2. Stage 2: In this stage, all the layers were trained for another 20 epochs with a reduced learning rate of $lr = 0.0001$.

This two-stage transfer learning process ensures that the newly initialized classification layer will not distort model weights too quickly and drastically as they are considered to be at desirable ranges.

Evaluation Protocol: All the deep learning models trained or fine-tuned on DATA-A were evaluated on DATA-B and DATA-C for verification in the subject-independent scenario. For datasets DATA-B and DATA-C, images belonging to the session 1 were

used as enrollment, and those belonging to the session 2 were used for verification. Features from enrollment-verification image pairs were computed using the corresponding CNN models. The verification performance of all models was computed for one-vs-all lighting condition matching and reported via equal error rate (EER%) and genuine match rate at 0.0001 false match rate (GMR@10^{-4}FMR). This means that for each device and for each lighting condition at enrollment, we calculate EER and GMR@10^{-4}FMR for all the lighting conditions in the verification set. Cosine similarity metrics (cos), as shown in equation 4.1, was used to generate the match scores in the [0,1] range. The final match score for a claimant was calculated as the maximum of all the scores between enrollment and test image pairs.

$$cos(\boldsymbol{x}, \boldsymbol{y}) = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{||\boldsymbol{x}|| \cdot ||\boldsymbol{y}||} \tag{4.1}$$

4.4 Experimental Results

Tables 5 - 7 and Tables 8 - 10 show the equal error rate (EER%) and genuine match rate at 0.0001 false match rate (GMR(%)@10^{-4}FMR) for both datasets DATA-B and DATA-C, respectively. Enrollment is from office lighting, daylight, and dim lighting conditions in session 1. Verification samples come from all lighting conditions across all mobile devices (iPhone 5s, Samsung Note 4, and Oppo N1) in session 2. The results are shown for all the pre-trained models and our proposed model trained from scratch on DATA-A. Note that VGG-19 was not fine-tuned due to memory issues.

It can be seen from Tables 5 - 7 that overall our proposed model outperformed

Table 5: EER(%) and GMR@10^{-4}FMR of all the models were evaluated on DATA-B dataset for enrollment images under specific lighting conditions and verification images under all lighting conditions, for iPhone-5s device.

Models	**Transfer Learning**	**iPhone 5s**		
		Office Light	**Day Light**	**Dim Light**
DenseNet-121		8.90/66.91	8.99/65.76	9.76/65.52
	Yes	5.72/63.87	5.29/65.46	5.69/62.03
MobileNet_V1_0.5_224		7.67/68.21	7.67/67.42	8.14/66.68
	Yes	6.55/64.64	6.67/64.56	7.28/62.38
MobileNet_V1_1.0_224		8.44/68.07	7.82/67.17	9.18/66.68
	Yes	6.22/65.16	6.36/65.61	7.33/62.73
MobileNet_V2_0.5_224		7.59/66.53	7.65/65.66	8.54/63.46
	Yes	7.21/62.75	7.10/61.75	7.89/60.09
MobileNet_V2_1.0_224		7.75/67.61	7.67/68.27	8.69/66.21
	Yes	7.04/61.34	6.86/60.25	7.67/59.51
NasNet-mobile		12.39/59.24	13.39/57.39	12.90/57.61
	Yes	9.81/53.22	10.07/51.93	10.82/52.31
ResNet-50		10.07/63.76	10.10/63.41	11.05/62.69
	Yes	5.96/68.73	5.26/69.97	6.35/65.98
VGG-19	No	11.97/62.04	12.11/61.55	13.04/60.98
Proposed model	**NA**	**5.28/73.28**	**4.71/71.33**	**4.87/70.09**

Table 6: EER(%) and GMR@10^{-4}FMR of all the models were evaluated on DATA-B dataset for enrollment images under specific lighting conditions and verification images under all lighting conditions, for Samsung Note-4 device.

Models	**Transfer Learning**	**Note-4**		
		Office Light	**Day Light**	**Dim Light**
DenseNet-121		9.80/62.53	9.00/65.00	9.32/63.66
	Yes	5.68/58.60	4.95/62.45	5.91/61.23
MobileNet_V1_0.5_224		8.60/64.38	7.54/66.56	8.32/65.06
	Yes	6.70/59.55	6.09/62.77	6.82/61.59
MobileNet_V1_1.0_224		8.57/64.14	8.06/66.00	8.88/65.12
	Yes	6.32/60.62	6.19/63.53	6.63/63.97
MobileNet_V2_0.5_224		7.43/62.71	7.06/64.84	7.66/62.14
	Yes	7.44/56.54	6.82/59.54	7.12/60.13
MobileNet_V2_1.0_224		7.90/63.80	7.13/67.64	8.09/65.00
	Yes	7.60/55.38	6.78/58.22	7.79/57.52
NasNet-mobile		13.23/54.08	12.52/55.87	14.07/55.08
	Yes	10.40/48.25	10.03/50.96	11.11/49.67
ResNet-50		10.33/59.52	10.09/61.45	10.82/59.04
	Yes	5.82/66.47	5.28/66.96	5.54/67.92
VGG-19	No	12.84/58.63	11.85/59.74	12.83/59.34
Proposed model	**NA**	**5.37/68.80**	**4.67/71.07**	**4.90/69.57**

Table 7: EER(%) and GMR@10^{-4}FMR of all the models were evaluated on DATA-B dataset for enrollment images under specific lighting conditions and verification images under all lighting conditions, for Oppo device.

Models	**Transfer Learning**	**Oppo**		
		Office Light	**Day Light**	**Dim Light**
DenseNet-121		12.28/62.07	11.88/63.48	9.78/65.43
	Yes	7.60/62.07	7.43/62.55	5.18/61.52
MobileNet_V1_0.5_224		11.63/64.36	10.73/65.91	7.56/68.45
	Yes	9.11/62.62	8.50/64.01	5.78/64.02
MobileNet_V1_1.0_224		12.34/64.31	11.54/64.74	8.69/68.62
	Yes	9.89/63.19	8.87/63.50	6.01/66.13
MobileNet_V2_0.5_224		10.37/63.27	10.09/64.63	7.49/66.56
	Yes	10.02/60.04	9.19/60.82	6.62/60.08
MobileNet_V2_1.0_224		10.87/63.56	10.63/65.86	7.55/68.49
	Yes	10.38/58.97	9.42/58.90	6.64/60.12
NasNet-mobile		16.49/56.85	15.50/56.86	14.50/56.66
	Yes	12.47/51.48	11.40/51.09	9.68/49.87
ResNet-50		13.83/60.67	12.94/61.62	10.40/61.83
	Yes	7.79/66.17	7.95/67.83	5.30/69.54
VGG-19	No	15.88/57.89	14.66/58.68	12.35/59.90
Proposed model	**NA**	**6.57/67.00**	**6.24/68.47**	**4.65/72.74**

all the rest by about 0.88% lower EER. This performance was followed by fine-tuned ResNet-50 and DenseNet-121 models on DATA-B. Whereas in the case of DATA-C, as it can be seen from the Tables 8 - 10, the fine-tuned ResNet-50 is the best performing model followed closely by DenseNet-121 and our proposed model.

Table 8: EER(%) and GMR@10^{-4}FMR of all the models were evaluated on DATA-C dataset for enrollment images under specific lighting conditions and verification images under all lighting conditions, for iPhone-5s device.

Models	**Transfer Learning**	**iPhone 5s**		
		Office Light	**Day Light**	**Dim Light**
DenseNet-121		10.35/56.13	8.61/61.84	8.38/61.89
	Yes	5.97/61.67	4.38/65.14	4.04/67.05
MobileNet_V1_0.5_224		9.45/63.64	7.56/67.92	6.48/69.33
	Yes	6.88/63.18	4.92/67.27	4.22/67.99
MobileNet_V1_1.0_224		9.57/62.22	7.57/66.96	7.07/67.49
	Yes	7.08/63.50	4.97/67.48	4.47/68.24
MobileNet_V2_0.5_224		8.63/61.92	7.55/66.17	6.72/67.99
	Yes	7.77/57.57	5.88/62.21	5.55/62.43
MobileNet_V2_1.0_224		9.29/62.16	7.64/67.08	7.16/68.03
	Yes	7.96/58.31	5.83/62.38	5.55/64.34
NasNet-mobile		14.59/52.18	13.54/57.91	12.69/56.79
	Yes	10.64/46.63	9.15/52.22	8.39/50.58
ResNet-50		11.82/55.19	10.05/60.90	9.87/60.40
	Yes	**5.56/65.28**	**3.83/70.22**	**3.43/71.64**
VGG-19	No	14.34/54.05	12.85/59.10	12.41/60.04
Proposed model	NA	**6.89/58.91**	**5.98/64.86**	**5.61/64.99**

However, taking the model size and number of operations (MAdd) into consideration, ResNet-50 is $35X$ larger in size and requires $15.6X$ more MAdd operations than our proposed model. Similarly, compared to DenseNet-121, the proposed model is $10X$ smaller in size and requires $11X$ fewer number of MAdd operations. The proposed model

Table 9: EER(%) and GMR@10^{-4}FMR of all the models were evaluated on DATA-C dataset for enrollment images under specific lighting conditions and verification images under all lighting conditions, for Samsung Note-4 device.

Models	**Transfer Learning**	**Note-4**		
		Office Light	**Day Light**	**Dim Light**
DenseNet-121		10.74/56.67	9.29/58.66	9.71/58.38
	Yes	6.01/60.47	4.40/64.26	4.55/63.79
MobileNet_V1_0.5_224		9.67/62.56	7.79/63.74	7.84/64.64
	Yes	6.82/62.20	5.28/66.84	5.44/65.50
MobileNet_V1_1.0_224		9.88/61.02	7.78/63.58	8.48/63.12
	Yes	6.80/63.31	5.06/68.82	5.26/65.65
MobileNet_V2_0.5_224		8.99/60.69	7.38/62.93	7.42/63.19
	Yes	7.68/55.99	5.72/59.43	6.20/59.68
MobileNet_V2_1.0_224		9.71/61.32	7.66/64.42	7.96/62.90
	Yes	7.68/57.52	5.92/62.93	6.26/61.89
NasNet-mobile		15.27/52.62	14.47/53.71	14.65/54.33
	Yes	10.70/46.20	9.34/48.15	9.39/49.08
ResNet-50		12.17/55.46	11.04/56.33	11.10/57.81
	Yes	**5.13/65.60**	**4.03/67.61**	**4.11/67.11**
VGG-19	No	15.02/53.77	14.38/56.61	13.75/56.36
Proposed model	NA	**6.02/61.90**	**5.55/62.21**	**5.63/63.12**

Table 10: EER(%) and GMR@10^{-4}FMR of all the models were evaluated on DATA-C dataset for enrollment images under specific lighting conditions and verification images under all lighting conditions, for Oppo device.

Models	**Transfer Learning**	**Oppo**		
		Office Light	**Day Light**	**Dim Light**
DenseNet-121		10.49/60.70	8.09/62.69	10.47/61.75
	Yes	6.00/62.48	3.62/67.66	6.29/65.53
MobileNet_V1_0.5_224		9.36/65.49	6.42/69.00	8.92/67.35
	Yes	6.82/64.13	4.51/69.16	7.37/68.28
MobileNet_V1_1.0_224		9.46/65.38	6.46/67.66	8.74/67.49
	Yes	6.68/64.49	4.23/68.51	7.14/66.09
MobileNet_V2_0.5_224		8.88/62.90	5.98/66.68	8.27/65.39
	Yes	7.51/60.15	5.61/62.16	7.77/59.56
MobileNet_V2_1.0_224		9.39/64.68	6.51/67.49	8.86/66.84
	Yes	7.24/60.17	5.12/63.14	7.46/62.45
NasNet-mobile		15.17/56.42	12.99/57.20	14.26/56.20
	Yes	10.37/51.48	9.03/52.52	10.35/50.05
ResNet-50		12.11/58.77	9.39/60.90	11.81/60.07
	Yes	**5.55/66.84**	**3.29/70.34**	**6.07/69.22**
VGG-19	No	14.50/57.86	12.59/61.47	13.94/58.40
Proposed model	NA	**6.57/62.05**	**5.01/65.46**	**7.00/64.60**

with $672K$ parameters is the smallest model in size than all other models tested, as shown in Table 3. However, compared to the MAdd operations, the proposed model is $2.7X$ larger than the smallest model(MobileNet_V2_0.5_224).

While comparing the performance of the proposed model to the mobile-centric deep learning architectures (MobileNet-v1, MobileNet-v2, and MobileNet-v3), the proposed model outperformed other models on DATA-B dataset as shown in Tables 5 - 7. On dataset DATA-C, the proposed model performed better than both MobileNet-V1 and MobileNet-V2 architectures. However, in daylight and dim lighting conditions (for enrollment templates), MobileNet-V1 and MobileNet-V2 architectures outperformed the proposed model.

4.5 Conclusion

In this chapter, we evaluated popular deep learning architectures relevant to camera-based mobile biometrics, as well as our custom proposed model, in terms of their accuracy, size, and computational cost measured for mobile ocular biometrics. To this aim, we performed subject-independent verification on VISOB ocular biometrics dataset using the fine-tuned and compact custom-designed model for mobile device-centric applications. Experimental results show that ResNet-50, DenseNet-50, and our custom models performed consistently better across all experiments. However, the proposed custom model is $35X$ smaller in size and has $15.6X$ fewer MAdd operations than the ResNet-50 model. Thus, it offers the best trade-off between performance and computational cost within the

bounds of our test use case. Further, the proposed model outperformed most of the compact mobile-centric architectures such as MobileNet-V1 and MobileNet-V2 in most of the experiments.

One main limitation considering the proposed custom models would the amount of time taken to train, 150 epochs with early stopping, is very high compared to performing fine-tuning on pre-trained models, which is only 30 epochs.

CHAPTER 5

OCULARNET MODEL

5.1 Introduction

In the previous chapter, from large scale analysis on different deep learning models conducted on mobile ocular biometrics, we can have the following findings:

1. Fine-tuning a pre-trained model will increases performance by a large margin on the ocular biometrics dataset.

2. It is possible to achieve better performance even with smaller CNN models. This can be seen from the matching performance of MobileNet architectures and proposed models being at least $10\times$ smaller than architectures such as ResNet.

3. Custom models trained from scratch can perform similar to that of popular models, such as MobileNet-V2 and ResNet-50, but with increased training time.

Based on these findings, we can build an efficient CNN model for mobile biometrics without losing any significant matching performance. However, there are also a few additional variables that need to be considered in mobile biometrics:

1. How does a change in the region of interest (ROI) of the eye affect the CNN models' performance? How can it be avoided? That is the variation in how eye image is generated. Depending on different eye detectors, the ROI of the eye will change and can affect the final matching performance.

2. How does a model perform not only in subject-independent evaluation but also in cross-dataset experiments? All the experiments conducted in the previous chapter are in subject-independent evaluation but with-in the same dataset collection.

3. How well does the model perform in cross-spectrum evaluations? That is a model trained on the visible spectrum and tested on cross-spectrum or completely different spectrum, like the near-infrared spectrum.

Taking these new variables into consideration, in this chapter, we propose a deep learning method to perform patch-based ocular biometric recognition. These patches are extracted from six overlapping windows of ocular and periocular regions. To ensure that the extracted patches are well defined for any input eye image, we incorporate an ROI detector for landmark localization. For each patch, a computationally small deep learning model named PatchCNN is trained to generate feature descriptors. Feature matching is performed by calculating the euclidean distance between each patch of enrollment and verification image pair. The final score for this is calculated using different score fusion techniques such as minimum, mean, and median of all patch scores. Finally, to get the score for the input verification image for a given enrollment subject, a minimum of all the scores are considered. We evaluate the proposed OcularNet models' performance compared to the popular CNN model, ResNet-50 [20], on large scale mobile VISOB [3] dataset in an open-world subject independent verification process where the model is trained on a subset of data that is not used in verification. We also evaluated the performance of the model on datasets with data acquisition methods such as UBIRIS-I [53], UBIRIS-II [54], and CROSS-EYED [55].

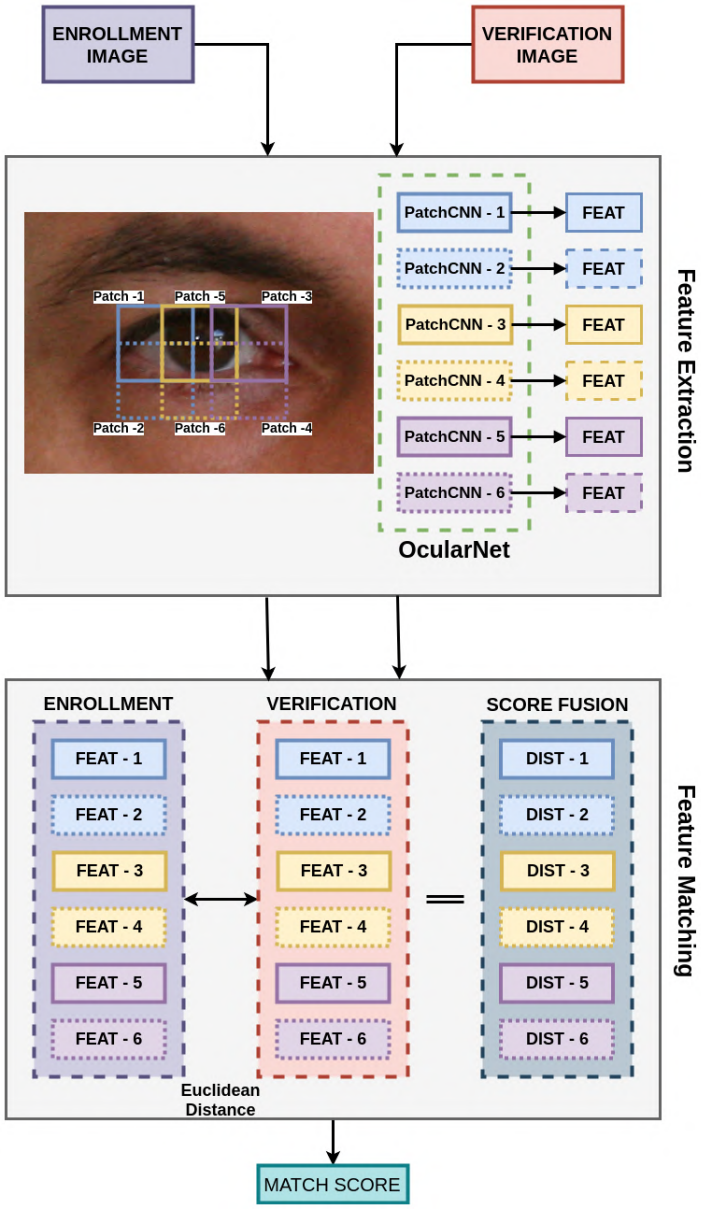
The rest of this chapter is organized as follows: Section 2 provides details on the proposed OcularNet model. Datasets, experimental protocol, and experimental results are presented in section 3. Conclusions are drawn in section 4.

5.2 Proposed Method

The block diagram of the proposed OcularNet model is shown in Figure - 11. First, we extract six overlapping patches from the ocular and periocular region for the given eye image. Second, for each patch, we trained a small CNN model to extract and generate feature descriptors. We explain our patch extraction process in section 5.2.1, the proposed PatchCNN architecture in section 5.2.2 and feature matching in section 5.2.3.

5.2.1 Eye Patches Extraction

In our experiments, to obtain robust ROI for eye images, we Incorporated Drishti's [56] eye landmark localization techniques. Figure - 12 depicts the eye landmark annotations generated using Drishti eye. Firstly we align the eye image horizontally using the left and right eye corner (landmarks 0 and 8). Now, patches 1-4 are extracted by scaling the image such that the width of the eye is 100 pixels. Then the patches of 64×64 pixel are cropped as shown in Figure - 11. Similarly, we scale the image such that the iris's diameter is 50 pixels to extract the patch numbers 5 and 6. This is done so that all the patches are approximately registered with the same scale for all the subjects. The feature extraction model is trained on these patches to extract features that are differentiable between the subjects.



1

Figure 11: Block Diagram of the Proposed OcularNet Model.

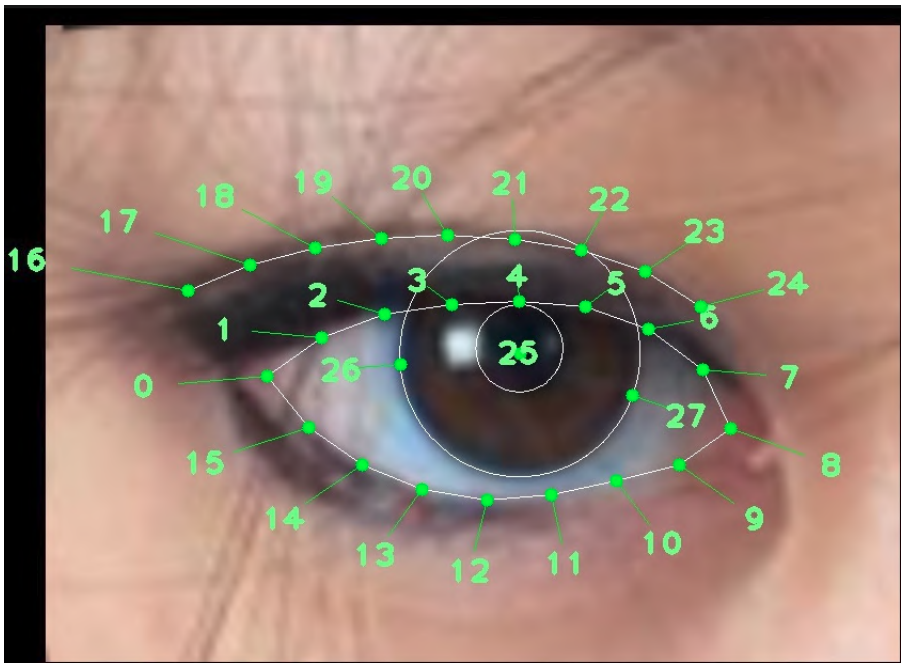


Figure 12: Annotations generated using Drishti Eye landmark localization.

5.2.2 PatchCNN

We used a ResNet [20] architecture, short for a residual network, based deep learning architecture for PatchCNN as shown in Figure - 13. In ResNet architecture, the input features skipped a certain number of layers and were shorted with the output layer. Using this idea of identity shortcut connection, we can build deeper CNN models without vanishing gradient problem. In our model, batch normalization followed by a rectified linear unit (ReLU) non-linearity is applied before each convolution layer, as shown in the figure. Also, when the stride is applied on a 3x3 convolutional layer or (and) increase the number of feature channels in the ResNet block, the skip connection is removed.

Table - 11 depicts the architecture of the proposed PatchCNN model with a total of only $253K$ parameters. Our model's input is a $2D$ image with a single color channel

(grayscale image). Considering we train six PatchCNN for each eye patch, the proposed OcularNet has $1.5M$ total parameters. The proposed model has 31 convolutions layers, followed by a multi-layer perception for embedded feature extraction.

Each PatchCNN is trained as a multi-class classifier by an additional classification layer at the end of the model in Table - 11. As a loss function, we considered center loss [57] with softmax loss to extract discriminative features. Each PatchCNN is trained for 50 epochs using Stochastic Gradient Descent(SGD) optimizer with momentum $(= 0.9)$. The training starts with an initial learning rate of 0.1, and then every 20 epochs learning rate is reduced by a factor of 10. A subset of a dataset from VISOB Visit - I, which will be explained in detail in the section - 5.3.1 is used as training data. After training, the classification layer is removed, and 128 embedded features are used as the feature descriptor.

5.2.3 Feature Matching

For a given enrollment and verification image pair, after extracting embedded features for each patch using respective PatchCNN, we use the Euclidean distance between each patch's embedded feature. As there are multiple patches, the final score will be the score level fusion of all the patches' distances from enrollment and verification image pair. We evaluated the mean, median, and minimum of patches scores as the final score for given enrollment and verification image pairs in our experiments.

Table 11: Structure of the Proposed PatchCNN for Feature Extraction and comparing a total number of parameters for OcularNet with ResNet-50 model with a single input channel.

Input Shape	**Layer**	**Output Shape**	**Parameters**
[1, 64, 64]	CONV 7x7, S2, 32	[32, 32, 32]	1,568
[32, 32, 32]	CONV (1x1x16, 3x3x16, 1x1x64, stride2)	[64, 16, 16]	3,968
[64, 16, 16]	ResNet 3X (1x1x16, 3x3x16, 1x1x64)	[64, 16, 16]	13,632
[64, 16, 16]	CONV (1x1x32, 3x3x32, 1x1x128, stride2)	[128, 8, 8]	15,616
[128, 8, 8]	ResNet 3X (1x1x32, 3x3x32, 1x1x128)	[128, 8, 8]	53,376
[128, 8, 8]	CONV (1x1x64, 3x3x64, 1x1x256, stride2)	[256, 4, 4]	61,952
[256, 4, 4]	ResNet 1X (1x1x64, 3x3x64, 1x1x256,)	[256, 4, 4]	70,400
[256, 4, 4]	Global Pooling	[256]	-
[256]	Dense Layer	[128]	32,896
For each PatchCNN, Total parameters			253,408
OcularNet: 6X PatchCNN			**1,520,448**
[1, 224, 224]	**ResNet-50 w/ single input channel**	[2048]	**23,501,760**

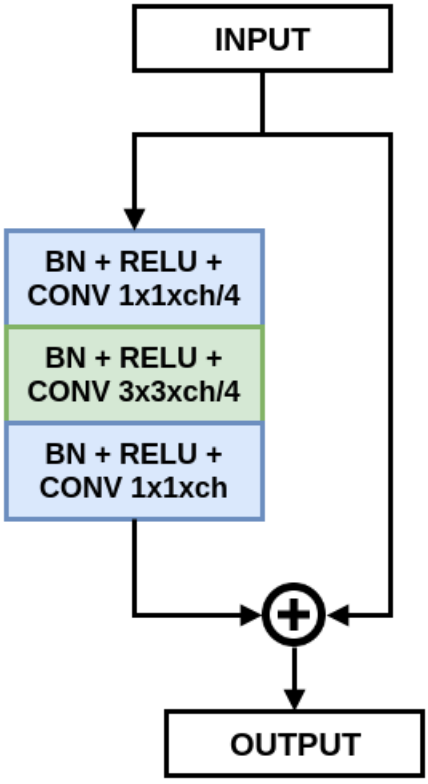


Figure 13: Block Diagram of ResNet architecture variant used in our model.

5.3 Experimental Evaluation

5.3.1 Datasets

We evaluated the performance of the OcularNet multiple datasets. We used VISOB [3] for training the models and subject independent verification evaluation. Also, to evaluate the performance of the model in different data acquired methods, we performed dataset independent experiments on UBIRIS-I [53], UBIRIS-II [54] and CROSS-EYED [55] datasets where we evaluate the feature extraction and matching performance of the proposed method trained on a subset of VISOB dataset on different datasets without

Figure 14: The sample of eye images from the VISOB dataset depicting variations such as motion blur, specular reflections, and different lighting conditions that are captured in this dataset.

any retraining.

VISOB [3]: VISOB is a publicly available dataset that was collected using front-facing cameras of different mobile devices (iPhone 5s, Samsung Note 4 and Oppo N1) under varying lighting conditions (office, daylight, and dim indoors) from over 550 healthy subjects. The data is collected from the subject in two visits (Visit - I and Visit - II) at 2 to 4 weeks apart, and in each visit, selfie like captures are taken from subjects under all lighting conditions and using all the three devices in two different sessions (Session - I and Session - II) that were about 10 to 15 minutes apart. Eye crops were generated using Viola-Jones based eye detector and resized to 160×240 pixel resolution. Variations such as motion blur, specular reflections, and different lighting conditions are captured in this dataset. Figure 14 shows example ocular images from VISOB dataset.

We used Visit - I of the dataset for our experiments. We divided the dataset into

Figure 15: The sample of eye images from UBIRIS-I dataset.

two subject-independent sets. The first set of the data consists of 200 subjects with $39,732$ images from the left and the right eye regions from all the mobile devices, lighting conditions, and sessions to train the OcularNet model. The remaining dataset of 350 subjects with $55,314$ images is used to perform subject independent verification testing, where session - I images are used in enrollments and session - II are used in verification.

UBIRIS-I [53]: Dataset consists of $1,877$ images collected from 241 subjects in two sessions using a DSLR camera. Figure 15 shows samples from UBIRIS-I. The dataset is mainly used in iris segmentation and matching evaluation in the visible spectrum. However, it is also used in other ocular biometric applications [58]. As we can see in figure - X, the UBIRIS-I dataset consists of the only ocular region with less to none periocular region. So, in our experiments, we evaluated the performance of the OcularNet for only $1,\ 3,\ \&\ 5$ patch ids.

Figure 16: The sample of eye images from UBIRIS-II dataset depicting variation in captured distance and poses.

UBIRIS-II [54]: In this version of UBIRIS, the data is collected from 261 subjects at multiple distances from 4 to 8 meters with varying poses from a DSLR camera in two sessions. As not all eye images at all distances have a periocular region, in this work, we performed experiments on samples collected at only $6,\ 7,\ \&\ 8$ meters. For this dataset, we will be evaluating OcularNet for all patch ids. Figure 16 shows samples with variation in captured distance and pose.

CROSS-EYED [55]: Unlike other datasets tested in this work, the CROSS-EYED dataset consists of samples captured in a visible spectrum and near-infrared spectrum from 120 subjects, as shown in Figure 17. As the evaluation for CROSS-EYED is across the spectrum, we considered visible spectrum images as enrollment set and near-infrared spectrum images as verification set. As the proposed model uses 2D single color channel images, we converted the RGB images to gray-scale in other datasets. However, in

Figure 17: The sample of eye images from CROSS-EYED dataset depicting samples from visible spectrum on top row and near-infrared spectrum images in the bottom.

the CROSS-EYED dataset, as it is across the spectrum, we choose the red color channel from visible spectrum images as the red color channel is closest in wavelength to the near-infrared spectrum. The CROSS-EYE dataset consists of two types of eye regions. One is only an ocular region image with the significantly less periocular region, and the other is periocular images with the ocular region masked. Because of the masked ocular region in periocular images, it is challenging to extract registered patches. We performed experiments only on ocular region images, as shown in Figure 17, and evaluated the performance of the OcularNet for only $1,\ 3,\ \&\ 5$ patch ids.

In our experiments, we flipped all the right eye images horizontally and considered these images belonging to the new unique user. For the purpose of this study, We flipped the right ocular images horizontally and considered these images as belonging to unique subject identities. This way number of unique subjects were doubled in all the dataset,

and all the results were computed for the left ocular images.

5.3.2 Evaluation Protocol:

To compare the performance of the OcularNet, which is a patch-based deep learning model, we considered training a single channel input ResNet-50 [20] model for comparison. The ResNet-50 model takes the input size of 224x224 and has $23.5M$ parameters, excluding the final classification layer. We trained the Resnet-50 model with the same dataset and training procedure used for OcularNet for better comparison. Compared to OcularNet with $1.5M$ parameters, ResNet-50 has 15.6X more parameters as shown in table - 11.

The verification performance of both models is evaluated via an equal error rate (EER$\%$) and genuine match rate at 0.0001 false match rate (GMR@1^{-4}FMR). For OcularNet, as we have multiple patches scores, we showed the final performance of minimum, mean, and median based score fusion techniques.

In the VISOB dataset, enrollments are considered in office lighting of the session - I, and verification are performed in all three lighting conditions from the session - II. For the rest of the dataset, UBIRIS-I, UBIRIS-II, and CROSS-EYE, the verification performance is shown for all the samples with enrollments from the session - I and verification images from the session - II.

5.3.3 Results:

Table - 12 show the verification performance of OcularNet model in comparison with ResNet-50 for VISOB dataset with enrollments in office lighting from the session

- I and verification from all three lighting conditions, office, daylight, and dim indoors, from the session - II. In all three devices, OcularNet achieved around 1% improvement in EER(%) compared to ResNet-50 and at least 10% improvement in GMR@1^{-4}FMR.

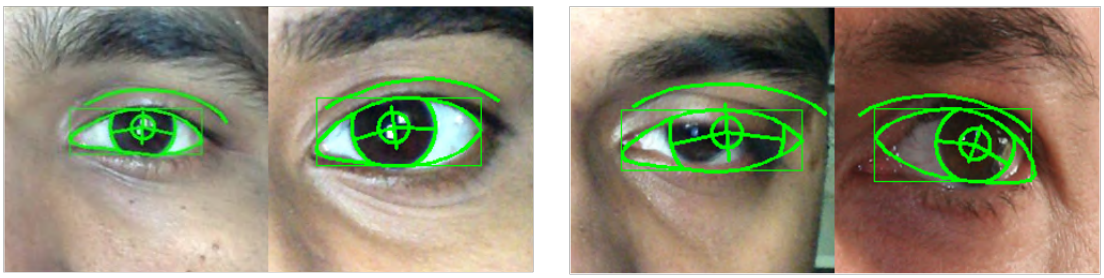
Table - 12 show the verification performance for UBIRIS-I, UBIRIS-II, and CROSS-EYE using the OcularNet model in comparison with ResNet-50. Just like in VISOB, OcularNet with mean based score fusion technique outperformed by at least 8% in EER(%) and 11% in GMR@1^{-4}FMR compared to ResNet-50.

Table 12: EER(%) AND GMR(%)@10^{-4}FMR for OcularNet and ResNet-50 evaluated on VISOB Visit - I dataset with enrollment set contains office light images from the session - I and verification set contains all the lighting conditions from the session - II.

Device	**Ocular Net (score fusion type)**			**ResNet-50**
	min	**mean**	**median**	
iPhone 5s	2.42/74.72	**1.89/76.12**	1.93/75.42	2.59/57.32
Oppo N1	1.97/65.62	**1.17/67.22**	1.23/66.63	2.62/56.51
Samsung Note 4	1.75/70.79	**1.23/72.53**	1.30/72.33	2.31/57.29

In the case of UBIRIS-I, UBIRIS-II, and CROSS-EYE datasets, there is a large difference in performance compared to OcularNet to ResNet-50. This is mainly because of registered patches extracted from the given eye image, which is explained in section 5.2.1.

From the above results, it can be seen that overall, the OcularNet model with the mean of all patch scores is outperforming other score fusion schemes and also the ResNet-50 model.



(a) (b)

Figure 18: (a) and (b) show examples of accurately generated eye landmarks and when the ROI detector failed, respectively.

While conducting visual analysis on the results, we noticed that the matching performance of OcularNet is dependent on how well the ROI detector performs. Figure 18(a) shows eye images where the Drishti eye landmark system correctly detected the ROI, and Figure 18(b) shows where it failed to do so and generates miss-aligned eye patches.

Table 13: EER(%) AND GMR(%)@10^{-4}FMR for OcularNet and ResNet-50 evaluated on CrossEyed, UBIRIS - V1, and UBIRIS - V2 datasets.

Dataset	**Ocular Net (score fusion type)**			**ResNet-50**
	min	**mean**	**median**	
CrossEyed	15.98/6.67	**14.95/11.82**	16.04/12.14	21.52/0.73
UBIRIS - V1	12.62/16.56	**9.86/26.03**	10.22/17.67	23.96/0.82
UBIRIS - V2	12.49/4.86	**9.77/14.18**	10.41/11.91	17.91/3.31

5.4 Conclusion

In this chapter, we proposed OcularNet, a patch-based CNN architecture for mobile ocular biometrics. We trained a small CNN, named patchCNN, on six overlapping patches extracted from ocular and periocular regions of the eye images for feature descriptor extraction. We showed that the proposed OcularNet model, which is 15.6X smaller than the popular ResNet-50 model, outperforms by at least 11% GMR at 1^{-4} FMR in subject independent verification setting in mobile VISOB dataset. We also showed that the proposed model achieved at least 8% lower EER compared to ResNet-50 on UBIRIS-I, UBIRIS-II, and CROSS-EYE datasets, which have different acquisition protocols compared to the VISOB dataset on which models are trained.

One of the main limitations of OcularNet is that it depends on the off-the-shelf ROI detector for eye images. If the ROI detector failed, then the extracted eye patches are miss-aligned, which leads to lower matching performance. Secondly, even though the OcularNet performed better than ResNet-50 in cross-spectrum evaluation on the CROSS-EYED dataset, the error rate reductions are not the same as in UBIRIS-I and UBIRIS-II datasets, which are in the visible spectrum same as the training dataset.

CHAPTER 6

OCULARNET-V2: SELF-LEARNED ROI DETECTION WITH DEEP FEATURES

6.1 Introduction

In Chapter 5, we proposed our first mobile ocular biometrics model, OcularNet. Small CNN models, named patchCNN, trained on six overlapping patches extracted from detected ROI from eye images. Even though we show significant performance improvement in terms of the cross dataset and cross-spectrum evaluation, we ran into the following limitations:

1. In OcularNet, depending on off-the-shelf ROI detectors for eye region detection, matching performance is affected if the ROI detector fails.

2. As OcularNet is a patch-based method; it is required to train multiple patchCNN's for each patch. Even though they are tiny, they are very much specialized in working on only the patch region. If the ROI detector fails, patchCNN fails to match.

3. As the models are not illumination normalization technique incorporated, OcularNet model had higher error rates in the CROSS-EYED dataset, which is a cross-spectrum (NIR vs. Visible) matching dataset. Whereas OcularNet is only trained using visible spectrum data.

This chapter proposes an OcularNet-v2 with an ocular region of interest (ROI)

detection model trained along with the feature extraction model in a self-supervised manner. That is, *the proposed ROI is learned in conjunction with the feature extraction such that the learned features maximize identifiability in the unconstrained environment*. This contrasts with the existing methods where the ROI model is learned separately and in a supervised manner. We also used a custom data augmentation pipeline to simulate the unconstrained environment of the acquired eye images for model training.

The **contributions** of this chapter are as follows:

1. We introduce an ocular ROI detection model based on spatial transformer networks (STN), which is trained along with the feature extraction model without any supervised learning to obtain robust and generalizable features.

2. A customized version of the MobileNet-v2 architecture is proposed with the input layer changed from a 3-channel input to a single-channel input for adaptability across different sensors and spectra. Further, we show that the last convolutional layers could be removed from the original implementation without affecting the accuracy while reducing the model size by $3.4\times$ compared to the original MobileNet-v2 and $36\times$ compared to the popular ResNet-50.

3. We introduce a data augmentation pipeline with variations such as random illumination, blur, zoom, and rotation to simulate the unconstrained mobile environment further. The data augmentation pipeline facilitates the extraction of robust features from the predicted ROI in non-ideal imaging scenarios.

4. We conduct a thorough, large scale cross-dataset evaluation. The proposed model

was trained only on 200 subjects from the VISOB dataset (in visible light) and tested on UBIRIS-V2, UBIPR, and FERET for cross dataset evaluation in the visible spectrum. Finally, the CASIA-TWINS dataset is used for near-infrared (NIR) evaluation, and CROSS-EYED for the cross dataset and cross-spectral evaluations.

The rest of the Chapter is organized as follows: The proposed method is discussed in section 6.2. Section 6.3 provides the details on the datasets used and the experimental protocol followed. In section 6.4, experimental results are discussed. Finally, the conclusion is drawn in section 6.5.

6.2 Proposed Method

Figure 19 illustrates the training and testing pipeline of our proposed model, which consists of an ROI detector based on a spatial transformer network aligned with the CNN-based feature extraction model. The modules of the proposed pipeline are discussed next.

6.2.1 Data Augmentation

To obtain a robust ROI detection and feature representation in an unconstrained environment, we first apply random photometric augmentations. Then, we resize all the images to 180×180 pixels and convert them into single-channel grayscale images to perform geometric data augmentation methods as follows:

- For photometric augmentation, we augmented illumination by randomly varying the brightness, saturation, and contrast of the image. We limited maximum and minimum values of brightness, saturation, and contrast to 50% of the original value

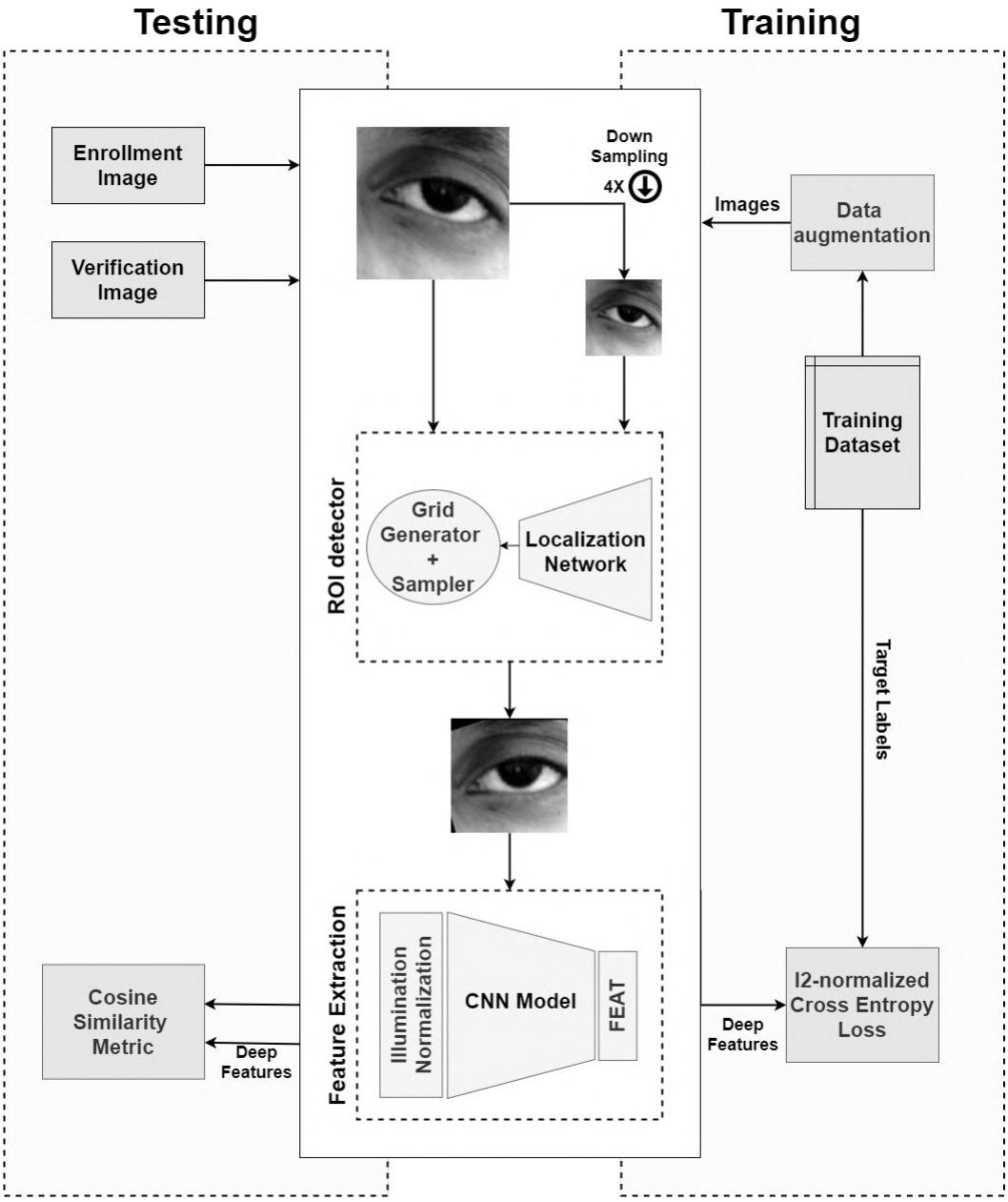


Figure 19: The proposed deep feature learning pipeline consisting of ROI detection and feature extraction modules.

to avoid unnatural distortions. We also applied a small amount of random Gaussian blur to the image to simulate the effect of de-focus. For Gaussian blur augmentation, we limited the blur kernel value to a maximum of $\sigma = 1.0$.

- For geometric augmentation, we randomly scaled the image from $0.8\times$ to $1.5\times$ of the original scale. Then, we rotated the image randomly between ± 20 angles and translated each randomly up to 40 pixels. Further, we extracted a center crop of the eye with a 160×160 pixel crop window, which is then fed to the proposed model for training and testing.

Our proposed data augmentation is applied on-the-fly to every image in a batch before being fed to the network, which is more memory-efficient during training. Figure 20 shows an example of the input ocular image along with the augmented image pairs generated using a combination of the above photometric and geometric augmentation.

6.2.2 Region of Interest (ROI) Detection

Padole et al. [34] showed that substantial improvement in matching accuracy could be had by aligning the eye images. Well-registered and normalized eye images are shown to match more efficiently and accurately in general. Other studies proposed ROI-based object detectors [59], supervised semantic mask generators [4], and much deeper models with more parameters to better counter spatial misalignment into feature extraction module [18]. However, these techniques either require large labeled datasets for robust ROI detection or have a sizeable computational footprint.

To overcome these problems, here we propose using a simple spatial transformer

Figure 20: Random augmentations for sample eye images (leftmost) obtained using our proposed augmentation pipeline consisting of photometric and geometric augmentation.

network (STN) [60], which is trained in conjunction with the feature extraction model to produce ocular ROIs for robust feature extraction. STN's main application is to reduce the spatial variance of the input images to achieve better recognition. Such a model can be trained without the need for labeled ROI data [60].

The **spatial transformer network** can be divided into three parts, explained below, along with our customizations for our proposed ROI detection model:

1. First, a localization network is used to predict the transformation matrix Θ. For a localization network, one can use an MLP or CNN model to predict Θ. Table 14 shows our proposed small localization network using a CNN with less than $110K$ parameters were, *ConvBNReLU*, convolution layer is used for feature extraction,

followed by batch normalization and ReLU, to normalize and add non-linearity to the features. In general, localization networks can be used to predict any transformation matrix. However, in our application, we use a 6 degrees-of-freedom affine transformation matrix (Θ_A), shown in equation 6.1.

$$\Theta_A = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} \tag{6.1}$$

2. Second, we generate coordinate grid samples by transforming the location of each pixel (x_i^t, y_i^t) in the input image I with $H \times W$ spatial dimensions to the detected ROI pixel coordinates (x_i^s, y_i^s) in the output image $I^{'}$ per equation 6.2.

$$\begin{bmatrix} x_i^s \\ y_i^s \end{bmatrix} = \Theta_A . \begin{bmatrix} x_i^t \\ y_i^t \\ 1 \end{bmatrix} = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} . \begin{bmatrix} x_i^t \\ y_i^t \\ 1 \end{bmatrix} \tag{6.2}$$

3. Finally, bi-linear interpolation is used to map the input image pixels I_{mn} to the new detected ROI pixel coordinates (x_i^s, y_i^s) in the output image $I_i^{'}$ using equation 6.3.

$$I_i^{'} = \sum_m^H \sum_n^W I_{mn}.\max(0, 1 - |x_i^s - m|). \max(0, 1 - |y_i^s - n|) \tag{6.3}$$

6.2.3 Feature Extraction

We used MobileNet-V2 like architecture [61] due to it is lower computational cost and memory size. An inverted residual block is proposed to reduce the size and

Table 14: Structure of the CNN-based localization network used for predicting affine transformation matrix A_Θ with 6 parameters. ConvBNReLU represents the convolution layer followed by batch normalization and ReLU. MAXPOOL represent max pooling layer.

Input	**Layer**	**Parameters**
[40×40×1]	ConvBNReLU (3×3×16, stride = 2)	192
[20×20×16]	MAXPOOL 2,2	-
[10×10×16]	ConvBNReLU (3×3×32, stride = 2)	4,704
[5×5×32]	Linear 128	102,528
128	Linear 6	774
Total Parameters		**108,198**

computational cost of the feature extraction model in MobileNet-V2 architecture, with separable 3×3 convolutional kernels, as shown in Figure 21.

The memory size and computational cost of a model can be calculated as the total number of parameters and the total number of MAdd (multiply-add) operations in all learnable layers, such as convolutional and fully connected layers. For a standard convolutional layer with k convolution with c_{in} input feature channels and c_{out} output feature channels, the total number of parameters is calculated as:

$$k \times k \times c_{in} \times c_{out} \tag{6.4}$$

MAdd operation with input features spatial size of $H \times W$, assuming stride $s = 1$, is calculated as:

$$k \times k \times c_{in} \times c_{out} \times H \times W \tag{6.5}$$

From the above equation 6.4 and equation 6.5, it can be seen that in a standard conventional layer, to extract one new feature, k convolution is applied on all the input features.

Table 15: The proposed feature extraction model based on MobileNet-V2 architecture, with the input layer changed from 3 to 1 channel. The input and output shapes are described in height × width × channels.

Input	ID	Layer	Output	Parameters	MAdd Ops (Millions)
160×160×1	1	ConvBNReLU (3×3×32, stride-2)	80×80×32	384	1.84
80×80×32	2	1 × InvertedResidual(t = 1, ch = 16)	80×80×16	896	5.12
80×80×16	3	2 × InvertedResidual(t = 6, ch = 24, stride = 2)	40×40×24	13,968	28.03
40×40×24	4	3 × InvertedResidual(t = 6, ch = 32, stride = 2)	20×20×32	39,696	19.10
20×20×32	5	4 × InvertedResidual(t = 6, ch = 64, stride = 2)	10×10×64	183,872	19.64
10×10×64	6	3 × InvertedResidual(t = 6, ch = 96, stride = 1)	10×10×96	303,168	29.64
10×10×96	7	3 × InvertedResidual(t = 6, ch = 160, stride = 2)	5×5×160	1,269,184	23.76
5×5×160	8	1 × InvertedResidual(t = 6, ch = 320, stride = 1)	5×5×320	473,920	11.74
5×5×320	9	ConvBNReLU ([3×3×1280])	5×5×1280	412,160	10.24
160×160×1	-	$MOD-0$ (all layers)	5×5×1280	2,223,328	149.12
160×160×1	-	$MOD-1$ (layers 1 to 8)	5×5×320	1,811,168	138.88
160×160×1	-	$MOD-2$ (layers 1 to 7)	5×5×160	1,337,248	127.14
160×160×1	-	$MOD-3$ (layers 1 to 6)	10×10×96	541,984	103.39

In the case of separable convolutions proposed in MobileNet-V2, only one k convolution is performed for each input, generating the same number of channels as the input (i.e., $c_{out} = c_{in}$). For separable convolutions, the number of parameters and MAdd operations is calculated as:

$$k \times k \times c_{in} \times 1 \tag{6.6}$$

$$k \times k \times c_{in} \times 1 \times H \times W \tag{6.7}$$

MobileNet-V2 proposed an inverted residual block shown in Figure 21, where larger kernel operations with $k = 3$ are performed on separable convolutions, which helps reduce the computational cost and size of the model.

We made two major modifications to MobileNet-V2:

1. We modified the input channel of the model to use only a single-channel illumination normalized image rather than an RGB image. Accordingly, the first convolutional layer is changed from 3 input channels to 1.

2. We removed the last three layers to reduce the size and number of computations. This was achieved without suffering any significant drop in matching performance.

Table 15 shows the complete architecture of our proposed CNN model. We used self quotient image (SQI) [62] followed by a simple contrast stretching for illumination normalization. Let σ be the Gaussian blur kernel that is applied to the input image I, then the SQI image, I_Q, is given as:

$$I_Q = \frac{I}{\sigma(I) + \epsilon} \tag{6.8}$$

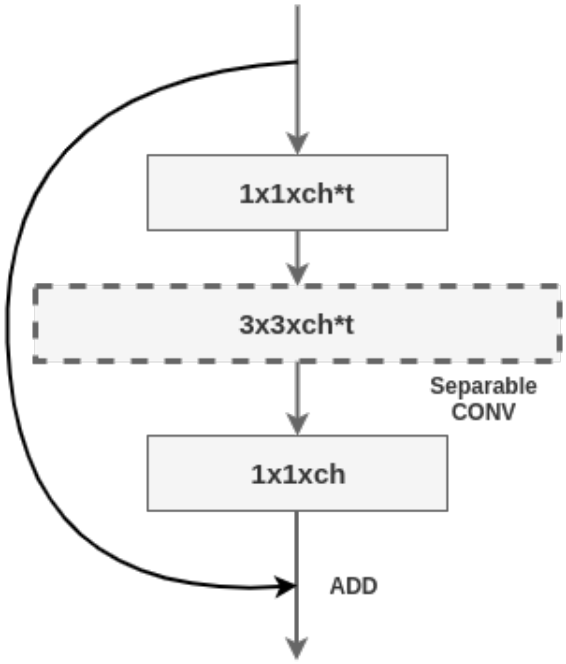


Figure 21: Inverted Residual block for MobileNet-V2 architecture. t is the channel expansion factor.

Where ϵ is a small constant to avoid division by zero. Afterward, contrast stretched image I_{QC} is computed from SQI image as follows:

$$I_{QC} = \frac{\max(0, \min(I_Q + \phi, \phi))}{\phi} \tag{6.9}$$

ϕ is the constant stretch parameter for the SQI image. We set $\phi = 1.5$ experimentally and based on visual analysis for our study. Figure 22 shows a sample ocular image obtained using the proposed SQI illumination normalization followed by contrast stretching.

Table 15 shows the proposed feature extraction model with modified input along

Figure 22: From left to right: 1) input RGB ocular image, 2) illumination normalization using self quotient image (SQI), and 3) the image after applying contrast stretch from equation 6.9

with input feature size, number of parameters, and the number of MAdd operations at each layer. With computational efficiency as one of our primary goals, we evaluated our MobileNet-V2 architecture with 4 different modifications denoted as $MOD-0$ to $MOD-3$ in Table 15. $MOD-0$ to $MOD-3$ indicate the number of removed ending layers (ranging from 0 to 3) in the architecture. The final embedded feature vector is obtained by simply performing global average pooling on the output features obtained from the feature extraction model.

6.3 Dataset And The Protocol

In this section, we discuss our study datasets and the training parameters of the proposed model.

6.3.1 Datasets

VISOB [3]: This dataset was collected using the front-facing camera of three mobile devices: iPhone 5s, Samsung Note 4, and Oppo N1 under three different lighting

Table 16: Summary of the datasets and their data division splits for the training and testing. Except for VISOB, the rest of the datasets were used for testing.

Dataset	Spectrum	Subjects (Train/Test)	Samples (Train/Test)
VISOB [3]	Visible	200/350	39,732/55,314
UBIRIS-V2 [54]	Visible	-/261	-/11,101
UBIPR [34]	Visible	-/344	-/10,257
CROSS-EYED (Iris) [5]	Visible and NIR	-/120	-/3,840
CROSS-EYED (Periocular) [5]	Visible and NIR	-/120	-/3,840
CASIA-TWINS [38]	NIR	-/100 pairs of twins	-/3,183
FERET [64]	Visible	-/994	-/7,196

conditions: office, daylight, and dim indoors. The data was collected in two visits, 2 to 4 weeks apart, with two capture sessions per visit from over 550 healthy adult volunteers. We used the Dlib library [63] for facial landmark detection. For eye localization, Dlib's 5-point face landmarks detection model was used. Eye crops were generated such that the eye is in the center of the crop, and the width of the eye (from eye corner to eye corner) is 60% of the aforesaid crop as shown in Figure 20 (leftmost images).

We used a subject-independent protocol and divided the dataset randomly into training and testing sets. This random division of dataset is once before starting the experiments. In the training set, we used $39,732$ images of the left and right eyes from only 200 subjects captured using all devices, lighting conditions, and sessions in VISIT-I. The remaining 350 subjects from VISIT-I, totaling $55,314$ left and right eye images were used for testing. Session-I samples were used for enrollment, and session II samples were used for verification.

UBIRIS-V2 [54]: This dataset was collected from 261 individuals using a DSLR

camera from 4 to 8 meters in two different sessions. The dataset contains $7,731$ samples for session 1 and $3,370$ for session 2, focusing on iris and ocular biometric recognition in visible light. We used all the samples for our evaluation in our experiments, with session-I samples set as enrollment and session-II samples used for verification.

UBIPR [34]: This dataset contains RGB periocular images acquired at multiple distances with varying poses. The dataset has 344 subjects with a total of $10,257$ samples acquired at 5 different distances (D1 - D5), which we used for cross distance matching. For example, if the samples in the distance D1 were in the enrollment set, then the remaining samples of each subject from D2-D5 were used for the corresponding verification set.

CROSS-EYED [5]: This is a cross-spectral dataset for periocular and ocular biometric evaluation. The dataset consists of eye samples from 120 subjects collected in visible and near-infrared (NIR) spectrum. Our experiments evaluated the dataset for cross-spectral matching with samples from the visible spectrum used as enrollment and those from the NIR spectrum for verification. This dataset consists of two types of eye crops. One is a close up of the eye with no periocular region focusing on iris recognition. The other crop focuses on the periocular region with the within the eyelids ROI masked out. We evaluated our model on both of these eye crops.

CASIA-TWINS [38]: This dataset contains NIR ocular images from 100 pairs of identical twins collected during the annual twins festival in Beijing. The dataset comprises of $3,183$ left and right ocular images in total. We performed our evaluation using all the samples.

FERET [64]: FERET is mainly a face recognition dataset. However, it has also

been used for ocular biometrics [10]. The dataset was collected from 994 subjects with different poses and facial expressions. In our evaluation, we used ocular images belonging to frontal (*fa* and *fb*) and slightly tilted faces (*rb* and *rc*). $2,670$ ocular images from the frontal *fa* pose were used for our enrollment set and all the remaining samples, totaling $4,526$ from frontal *fb* pose, and all samples with slight tilting face poses i.e., *rb*, and *rc*, were used for the verification set. The ocular region was localized and segmented using Dlib facial landmark detection library [63].

Table 16 shows a summary of datasets used, along with the samples in training and testing splits. All the models proposed in this work were trained on the training subset of the VISOB dataset. The remaining datasets, including the testing subset of VISOB, were utilized for subject-independent analysis and cross dataset analysis across different spectra. ***To further increase the level of matching difficulty, all the right-eye images were flipped horizontally and considered as new subjects, similar to identical twins case.*** This doubles the total number of subjects in both training dataset and testing datasets given in Table 16, resulting in 400 subjects for training the models.

6.3.2 Network Training

In our experiments, MobileNet-V2 model pre-trained on ImageNet dataset [12] was fine-tuned on our training dataset using transfer learning. Adam [52] optimizer was used for training our models. As mentioned in section 6.2.3, we modified MobileNet-V2 with a new input layer with channel 1, and the remaining layers were used with pre-trained weights from ImageNet dataset [12]. For this reason, we used two different learning rates.

For the new initial layer-1 in Table 15 and ROI detection model from Table 14, we chose an initial learning of 0.001. For layers 2 to 9 with pre-trained weights in Table 15, we chose an initial learning rate of 0.0001 since they already have useful common information reflected in their values. The learning rate was reduced by a factor of 10 after the first 5 epochs and then after every 50 epoch. Models were trained using L2-normalized categorical cross-entropy loss [57] for a total of 150 epochs with early stopping. The loss is calculated as:

$$loss = -\frac{1}{M}\sum_{i=1}^{M} log\frac{\alpha \cdot e^{\hat{W}_{y_i}^T \cdot f(\hat{x}_i)}}{\sum_{j=1}^{C} e^{\hat{W}_{y_j}^T \cdot f(\hat{x}_i)}} \tag{6.10}$$

In equation 6.10, $f(x_i)$ denotes the L2-normalized deep features extracted from the model for given input image x_i in batch M belonging to class y_i with target weights W_{y_i}. C denotes the total number of classes and α is the relaxation constant used for speeding up the training process. Here ($\hat{}$) denotes L2-normalization as shown in equation 6.11. In our experiments, we set $C = 400$ which is the number of subjects in training dataset (see section 6.3.1).

$$\hat{x} = \frac{x}{\|x\|} \tag{6.11}$$

6.3.3 Matching Metric

Since this is a subject-independent, user-independent evaluation, we used cosine similarity to generate the matching score between enrollment and verification feature vectors, which has been very successful and widely used in popular deep learning based

biometric systems like face recognition systems [65, 66]. Cosine similarity between two inputs x_i and x_j is given as follows:

$$s_{cos} = \hat{x_i} \cdot \hat{x_j} = \frac{x_i}{\|x_i\|} \cdot \frac{x_j}{\|x_j\|} \tag{6.12}$$

In equation 6.12, we can see that cosine similarity is the dot product between two L2-normalized inputs and is bound between -1 and 1. We considered cosine similarity because, during training, the loss from equation 6.10 is minimized when the cosine similarity between deep features $f(x_i)$ and corresponding class weights w_{y_i} is maximized. Similarly, during the evaluation, we obtain the max bound of the cosine similarity, 1, when enrollment and verification images belong to the same subject, and vice versa.

During all our verification experiments, the match score was computed as the maximum cosine similarity between the given verification sample and all the enrollment samples for a given subject (multi-template matching), which is a common practice in many biometric systems to improve accuracy and robustness.

6.3.4 Model Architecture Evaluation

To design an efficient feature extraction model, we evaluated feature matching performance of all 4 reduced versions ($MOD - 0$ to $MOD - 3$) of the MobileNet-V2 architecture as mentioned in Table 15. Using the proposed augmentation pipeline in section 6.2.1 and the training subset of VISOB dataset (section 6.3.1), we evaluated subject-independent performance of all the four modifications from $MOD-0$ to $MOD-3$. The architecture was evaluated on the samples from the testing subset of VISOB.

Session I and II samples were used for enrollment and validation, respectively. Since this experiment was to find an efficient feature extraction model, the region of interest detector mentioned in section 6.2.2 was not used.

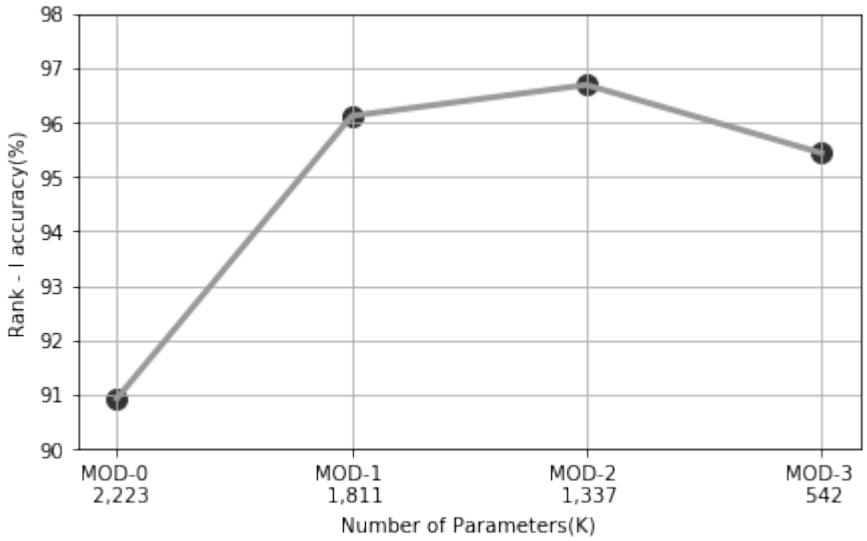


Figure 23: Rank-1 identification accuracy (%) of all modifications to the MobileNet-V2 architecture with respect to the number of parameters.

As this is a subject-independent user-independent evaluation, we used cosine similarity to generate the matching score between enrollment and verification feature vectors. Rank-1 identification rate was used as our performance metric. Figure 23 shows the rank-1 performance of each modification with respect to the number of parameters. Rank-1 performance evaluation in an identification setting was conducted to analyze the potential of different architectures more rigorously. This is because identification (one to many matching) is a more difficult task than verification (one-to-one matching) with a higher chance of false matches, especially when probing a large gallery pertaining to all enrolled samples from all the users in the dataset.

Examining Figure 23, it is interesting to note that the rank-1 accuracy of the full MobileNet-V2 ($MOD-0$) is 5% lower than $MOD-1$ (with its layer-9 removed) . From $MOD-1$ to $MOD-3$, there is only 1% increase in rank-1 accuracy. Even after removing the last 3 layers in $MOD-3$, the performance is only dropping by around 0.5% compared to $MOD-1$. Further, compared to $MOD-1$, $MOD-3$ has $3.3\times$ fewer parameters and $3.3\times$ smaller embedded feature size as shown in Table 15 and Table 22, respectively.

Therefore, for the rest of our experiments, we chose and used $MOD-3$ modified architecture of MobileNet-v2 model (referred to as OcularNet-v2) (Figure 19) along with ROI detector (section 6.2.2). In total, the proposed pipeline (OcularNet-v2) has only $650K$ parameters, which is $3.4\times$ smaller than the full MobileNet-V2 ($MOD-0$) and around $36\times$ smaller than the popular ResNet-50 model.

For the localization network in our proposed STN model in section 6.2.2, we resized the input image from 160×160 pixels to $\frac{1}{4}\times$ of its original size, i.e., 40×40 pixels, before feeding it to the localization network. This was done to reduce the computational cost and to facilitate using a smaller ROI prediction model.

6.4 Experimental Results

In our experiments, training was performed using only the VISOB's training subset, while testing was carried out on all the datasets using the subject-independent protocol discussed in section 6.3.1. This challenging data division was designed to evaluate the generalizability of the proposed method in a subject-independent, cross dataset, and

Table 17: VISOB test set verification results with the genuine match rate at 0.001 false match rate (GMR (%) @ 0.1% FMR) for each lighting condition: office, daylight, and dim indoors; for all three mobile devices: iPhone, Note-4 and Oppo N1.

iPhone			
Model	**Office**	**Day light**	**Dim**
MOD-1	93.00	96.12	95.96
MOD-3	93.94	96.52	96.00
OcularNet-v2	**94.53**	**95.97**	**96.13**
Note-4			
	Office	**Day light**	**Dim**
MOD-1	93.79	96.30	94.58
MOD-3	94.26	94.59	94.31
OcularNet-v2	**91.94**	**95.22**	**95.40**
Oppo N1			
	Office	**Day light**	**Dim**
MOD-1	91.57	96.20	96.19
MOD-3	91.27	96.20	95.91
OcularNet-v2	**90.93**	**95.85**	**95.74**

cross-spectral matching scenarios.

For all the experiments, results are evaluated using the Equal Error Rate (EER$\%$) [67]. Also, Genuine Match Rate (GMR$\%$) at fixed False Match Rates (FMR$\%$) [67] are reported following the published measurement practices over the utilized datasets.

We compared the performance of the proposed model, OcularNet-v2, with $MOD-1$ and $MOD-3$ models which are discussed in section 6.2.3. All the models were trained on the same subset of the VISOB for fair performance comparison.

We evaluated the VISOB's test set using a verification protocol where the genuine match rate at 0.001 false match rate (GMR$\%$ @ 0.1% FMR) is set as the performance metric, with the enrollment samples coming from session-I and verification samples from session-II subsets. As shown in Table 17, all the models' evaluated genuine match rates are within a $\pm 1\%$ of each other. This could be due to models being trained on the same VISOB subset, with samples drawn from all the lighting conditions and devices. Thus these models generalized well even though the VISOB test set excludes the identities that appear in the training data (subject-independent evaluation).

6.4.1 Cross Dataset Results

Table 18 shows equal error rate (EER$\%$) and GMR at 1% FMR for UBIRIS-V2 dataset. We evaluated the trained models using two data splits, $D1$, and $D2$. $D1-set$, with samples drawn from distances within 6 to 8 meters, where periocular regions had consistent quality. $D2-set$, where samples from all distances are considered, including closer distance samples with large variation in the acquired periocular region. From the

Table 18: EER(%) and GMR(%) at 1% FMR for the Ubiris-V2 dataset, comparing the proposed method with reported methods in the literature on two different datasets. *Note: Results for PRIWIS are from [4]*

Model	EER (%)	GMR (%) @ 1% FMR
Distance : 6 to 8 meters (D1-set)		
ResNet-50 [59]	17.91	-
OcularNet [59]	9.77	68.58
MOD-1	7.78	73.74
MOD-3	8.69	70.76
OcularNet-v2	**7.65**	**72.3**
Distance : all distances (D2-set)		
PRWIS by Proenca et al. [18]	22.95	-
Model by Zhao et al. [4]	10.05	-
MOD-1	10.00	65.06
MOD-3	10.33	65.36
OcularNet-v2	**9.1**	**69.82**

Table 19: EER(%) for multi-distance evaluation on the UBIPR dataset, with enrollment samples from one distance and verification samples from the other four distances.

Enrollments at Distance	MOD - 1	MOD - 3	OcularNet-v2
D1	4.16	5.54	**4.43**
D2	3.49	4.57	**3.87**
D3	3.73	5.24	**4.18**
D4	4.18	5.87	**4.35**
D5	3.97	5.21	**4.18**

Table 18, in $D1 - set$ evaluations, it can be seen that all trained models outperformed our earlier OcularNet [59]. In the case of our proposed model, we obtained 2% lower EER and close to 4% improvement in GMR(%) at 1% FMR. Also, it can be seen that in $D1 - set$, the match rate of the proposed model OcularNet-v2 was comparable to the much larger $MOD - 1$ model. However, as large variations in the periocular region are introduced in $D2 - set$, the proposed model outperformed the $MOD - 1$ model showing 1% lower EER$\%$ and 4% increase in GMR(%) at 1% FMR. It can also be seen that the proposed model outperformed the PRWIS model proposed by Proenca et al. [18] by more than 13% in terms of EER, and the model proposed by Zhao et al. [4] by 1% reduction in EER using subject-independent evaluation.

For the UBIPR dataset, we evaluated the performance in terms of EER(%) with enrollment (but not necessarily verification) samples acquired from a certain distance. For example, enrollments coming from a distance of $D1$ meters and verification samples from all the remaining distances. It can be seen from Table 19 that the proposed model $STN + MOD - 3$ is better than $MOD - 3$ with 1% lower EER, and lagging only by 0.3% in EER compared to $3.3\times$ larger model $MOD - 1$.

Cross spectral testing was performed on the CROSS-EYED dataset for both periocular and iris samples. We compared our trained models with the top three methods from the CROSS-EYED 2017 competition using EER and GMR @ 1% FMR as performance metrics. From Table 20, we see that the performance of our proposed methods (MOD-1, MOD-3, and OcularNet-v2) is only second to the best models from the competition using iris and periocular competition data. When it comes to our trained models, the proposed

Table 20: EER(%), and GMR(%) at 1% FMR for CrossEyed dataset. The first three methods for the iris and periocular dataset are the top 3 performing from CROSS-EYED 2017 competition [5]

Model	EER (%)	GMR (%) @ 1% FMR
Iris Dataset		
NTNU4	0.05	0.00
NTNU3	5.58	8.43
NTNU1	6.19	8.81
ResNet-50 [59]	21.52	-
OcularNet [59]	14.95	-
MOD-1	3.80	13.91
MOD - 3	3.75	14.74
OcularNet-v2	**2.71**	**7.86**
Periocular Dataset		
HH1	0.82	0.74
NTNU1	1.59	1.86
IDIAP2	1.65	2.03
MOD-1	2.40	5.21
MOD - 3	2.19	4.38
OcularNet-v2	**0.94**	**0.83**

Table 21: EER(%), and GMR at 0.1% FMR, for FERET dataset and CASIA-TWINS dataset using (a) all data and (b) twins only data.

-	**FERET dataset**		**CASIA-TWINS (All data)**		**CASIA-TWINS (Only Twins)**	
Model	**EER**	**GMR @ 0.1% FMR**	**EER**	**GMR @ 0.1% FMR**	**EER**	**GMR @ 0.1% FMR**
MOD-1	6.79	72.43	12.02	56.73	11.75	60.07
MOD-3	7.43	69.85	12.83	58.70	13.07	55.70
OcularNet-v2	**6.06**	**72.96**	**11.29**	**65.18**	**9.41**	**69.18**

model (OcularNet-v2) achieved a 6% lower GMR compared to a much larger $MOD-1$ model over the iris dataset. Similarly, using the periocular dataset, the OcularNet-v2 model achieved 4.38% lower GMR compared to $MOD-1$.

For the FERET dataset, the proposed method, $STN+MOD-3$, performed slightly better than other evaluated models, with approximately 0.5% improvement in EER and GMR (at 0.1% FMR), as shown in Table 21. For NIR spectrum CASIA-TWINS, the proposed model's one-to-one matching performance was calculated over (a) all the datasets and (b) between twin pairs only. From Table 21, it can be seen that the proposed model outperforms by 8.45% higher GMR at 0.1% FMR for all data comparison (a), and by 9.11% higher GMR at 0.1% FMR in case of twins only comparison (b).

6.4.2 Visual Analysis of the ROI Model

Our proposed ROI model is based on STN architecture and trained using self-supervision along with the feature extraction model $MOD-3$. Thus the ROIs detected by the model are entirely dependent on the training dataset. After training the model for

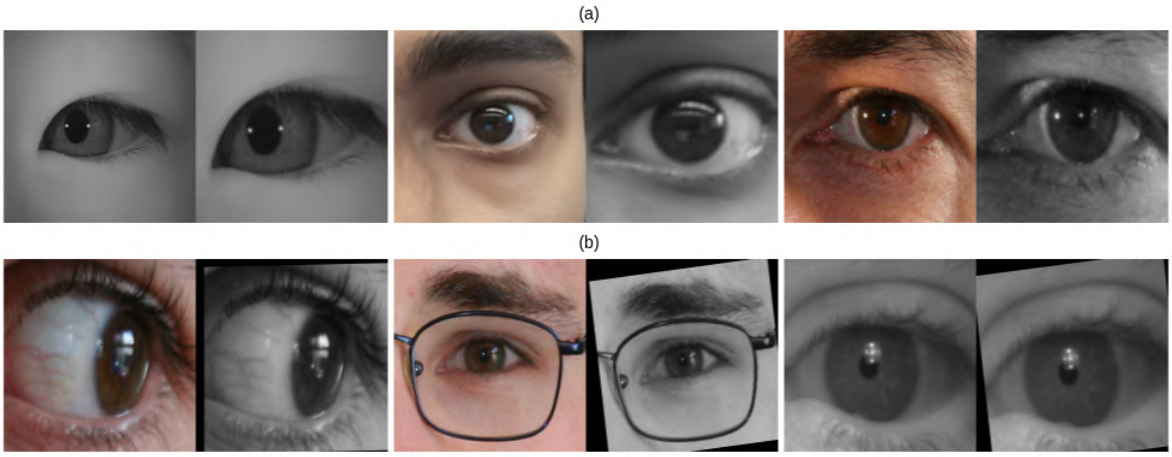


Figure 24: (a) Shows eye samples where the STN model extracted good ocular ROIs in the different wavelengths. (b) It shows eye samples, with bad ocular crops or eyeglasses frames showing, where the STM model failed to crop the input correctly. Note: for each image pair, the left image is STN model input, and the right one is the output.

$40 - 60$ epochs, the STN model starts to learn to detect the proper ROI from the input eye images. These ROIs are expected to have better classifiability, as shown in Figure 25. It can be seen from Figure 25 that the feature extraction model learned to obtain better classifiable features from the crop with the eye at the center, horizontally aligned, and the width of the eye covering about 90% of the crop.

As for the test set, Figure 24 (a) shows samples where the STN model successfully obtained ROIs with the eye properly centered. However, as shown in Figure 24 (b), in the case of eye images with little to no periocular region, the model failed to align. The model also failed in samples with prescription eyeglasses. However, this is due to the fact that VISOB visit I, used for training, did not contain subjects with glasses.

Figure 25: STN model output with the learned ocular ROI using the augmented samples from training.

6.4.3 Execution Time On Embedded Device

To evaluate the computation cost of the proposed model on mobile and Embedded systems, we choose Nvidia Jetson Nano [68] for standardized testing and used execution time (in milliseconds) as the performance metric. The experiments are conducted on the deep learning framework Pytorch-1.6 at 32bit floating-point precision with just-in-time (JIT) compilation to achieve consistency and the best performance. To compare the performance of the proposed model, we chose ResNet-50, MobileNet-v2 and modification which are proposed for MobileNet-v2 in section-6.2.3, $MOD-0$ to $MOD-3$. Table-x shows the execution time (ms) and the size of the all the evaluated models. It can be seen that the proposed model requires only $37.6ms$ with model size being $36\times$ smaller compared to ResNet-50. Also, compared to full MobileNet-v2, the proposed model is $3.4\times$

Table 22: Comparison of execution times and parameter sizes using the proposed model OcularNet-v2 with proposed feature extraction models $MOD-\{1,3\}$, OcularNet-v1 ($6\times$ PatchCNN), and popular CNN architectures such as MobileNet-v2 and ResNet-50.

Model	# Parameters	Execution time (ms)
ResNet-50	23.50M	92.1
OcularNet	1.52M	$6 \times 27.72 = 166.33$
MobileNet-V2	2.22M	37.1
MOD-0	2.22M	36.7
MOD-1	1.81M	35.5
MOD-2	1.34M	33.5
MOD-3	542K	30.2
OcularNet-v2	650K	37.6

smaller with the same execution time. OcularNet-v2 uses $MOD-3$ as a feature extractor with an STN model for ROI detection. As $MOD-3$'s execution time is $30.2ms$, it should be noted that the STN model in OcularNet-v2 requires around $7ms$ for ROI detection.

6.4.4 Key Findings of the Study

1. For VISOB, UBIPR, FERET, and farther distance subset of UBIRIS-V2 ($D1-set$), the eye samples are already well aligned and centered; thus, little needs to be done in terms of image alignment or ROI detection. Therefore applying STN along with $MOD-3$ did not provide significant improvements. Overall, the equal error rate and genuine match rates stay within $\pm 1\%$ of non-ROI-detecting models.

2. However, with the introduction of data with more considerable variations, as seen in CASIA-TWINS or UBIRIS-V2($D2-set$), the proposed method shows significant

improvements in accuracy compared to $3\times$ larger (and also deeper) CNN model, $MOD-1$. Overall, we see a 9% improvement in GMR at 0.1% FMR using the proposed ROI detection model.

3. From visual inspection of ROIs in the section - 6.4.2, we can see that the self-supervised STN model is capable of centering and aligning the eye as well as proper rotation to level the eye image.

4. With a further visual inspection, one can see that large pose variations, over-cropped inputs, and samples with visible glass frames cause the proposed ROI model to fail. This issue may be mitigated by improving the augmentation pipeline and introducing samples with large pose variations and eyeglasses into the training set.

6.5 Conclusion

This chapter introduced OcularNet-v2: an efficient feature extraction model for ocular biometrics in unconstrained environments. OcularNet-v2 consists of an ocular ROI detector trained in a self-supervising manner and the feature extraction model. The feature extraction model is a modified version of MobileNet-V2 architecture, which is $36\times$ smaller than the popular ResNet-50. We also proposed a custom data augmentation pipeline and illumination normalization technique to learn robust features even in the presence of an adverse imaging environment and in cross-spectral matching, not to mention mirrored eyes that simulate subjects akin to identical twins. Our experimental results show that our model, which was trained using only 200 subjects from the visible light VISOB dataset, can easily generalize to other datasets, including those captured in NIR. We

evaluated our model on UBIRIS-V2, UBIPR, FERET, CROSS-EYED, and CASIA-IRIS-TWINS datasets and could obtain error rates up to $7\times$ lower than the existing models' as reported in the literature.

CHAPTER 7

LOD-V: LARGE OCULAR BIOMETRICS DATASET IN VISIBLE SPECTRUM

7.1 Introduction

In the previous chapter, we proposed OcularNet-v2, an efficient feature extraction model for ocular biometrics in unconstrained environments, and shown significant improvements in verification performance in subject-independent, cross dataset evaluation datasets with samples captured in different lighting spectrum(NIR).

In OcularNet-v2 and in the literature, evaluations for subject-independent mobile verification protocol are conducted by creating a training and testing set by dividing the dataset subject-wise. In our experiments, we choose 200 subjects from the VISOB dataset as the training set and reaming samples for subject-independent evaluation.

However, there two limitations to this approach:

1. In the real world, the number of subjects (N) using the biometric system is large. By dividing the evaluation, dataset purpose will make it challenging to evaluate real-world performance.

2. The proposed biometrics model needs to work on many mobile devices from different manufactures with a wide variety of selfie camera parameters. Training and testing on the same dataset will make it difficult to evaluate the model's real-world performance.

To ensure the evaluation is conducted in a subject-independent environment without the limitation mentioned above, models are trained on our new proposed dataset LOD-V (Large Ocular dataset in Visible Spectrum), which consists of more than 750 subjects with around $200K$ samples. LOD-V collects existing high-quality face datasets for face recognition and anti-spoofing purposes, from which periocular images are extracted for this evaluation. Testing is conducted on datasets from Chaprter 6, UBIRIS-V2 [54], UBIPR [34], CROSS-EYED [5], FERET [64], and CASIA-TWINS [38] datasets and show significant performance improvements in cross-dataset and cross spectrum evaluation.

The rest of the chapter is as follows, In Section-7.2, we provide details of the new LOD-V dataset. Experimental setup for training and testing OcularNet-v2 are provided in Section-7.3. In Section-7.4, experimental results are discussed. Finally, the conclusion is drawn in Section-7.5.

7.2 Creating LOD-V Dataset

LOD-V data is a collection of 8 datasets containing high-quality face data created for anti-spoofing and facial expressions research. We extract eye images from the subjects from each of these face datasets such that there are at least 4 samples per eye are available. In total, the proposed LOD-V dataset has 772 subjects with a total of $217K$ samples. Table 23 shows the number of subjects and samples per face datasets present in the LOD-V dataset.

Figure 26: Eye samples generated from CASIA Face anti-spoofing dataset.

7.2.1 Face Databases

In the following section, we introduce all the 8 face datasets used in creating LOD-V briefly.

Casia Face anti-spoofing [69] dataset consists of 50 unique subjects. The dataset is video samples collected from live subjects at three different video qualities. In our experiments, the video samples from only high-quality videos are used for extracting eye images. The dataset also consists of the spoof video samples; however, we did not use them for our biometric recognition experiments.

The **Chicago Face Database** [70] is a facial expressions dataset consisting of face images from 597 volunteers with different ethnicities one of five other facial expressions for fear, anger, and happiness close-mouthed, happy open-mouthed, and neutral. The

Table 23: Number of samples and subjects for all the databases in LOD-V dataset.

Database	All samples		4 samples/subject	
	# Subjects	# Samples	# Subjects	# Samples
Casia Face anti-spoofing	50	4242	50	4242
Chicago Face Database	597	2414	156	1524
FACES	171	4104	171	4104
KDEF	70	1960	70	1960
Oulu-NPU	55	50518	55	50518
RaFD	67	3216	67	3216
Replay-Mobile Database	39	30790	39	30790
SiW database	164	121496	164	121496
Total	**1213**	**218740**	**772**	**217850**

dataset consists of 2414 samples; however, as we only considered subjects with at least 4 samples available, we used 1524 samples from 156 subjects in our experiments.

FACES [71] dataset contains high-quality face images collected from 171 Caucasian volunteers for facial expressions recognition. The dataset is divided into three age groups, young ($n = 58$), middle-aged ($n = 56$), and older ($n = 57$), all expressing six facial expressions: neutral, sadness, disgust, fear, anger, and happiness. The dataset comprises two samples per expression per person with a total of $2,052$ images.

The **Karolinska Directed Emotional Faces** (KDEF) [72] is a set of 4900 in pictures from 70 representative across ethnicity, race, sex, and gender with 7 primary facial expressions captured in five different facial poses. We only used samples from frontal poses in our experiments and ended up with 1960 samples from all 70 individuals.

The **Oulu-NPU** [73] is face anti-spoofing detection database consists real and

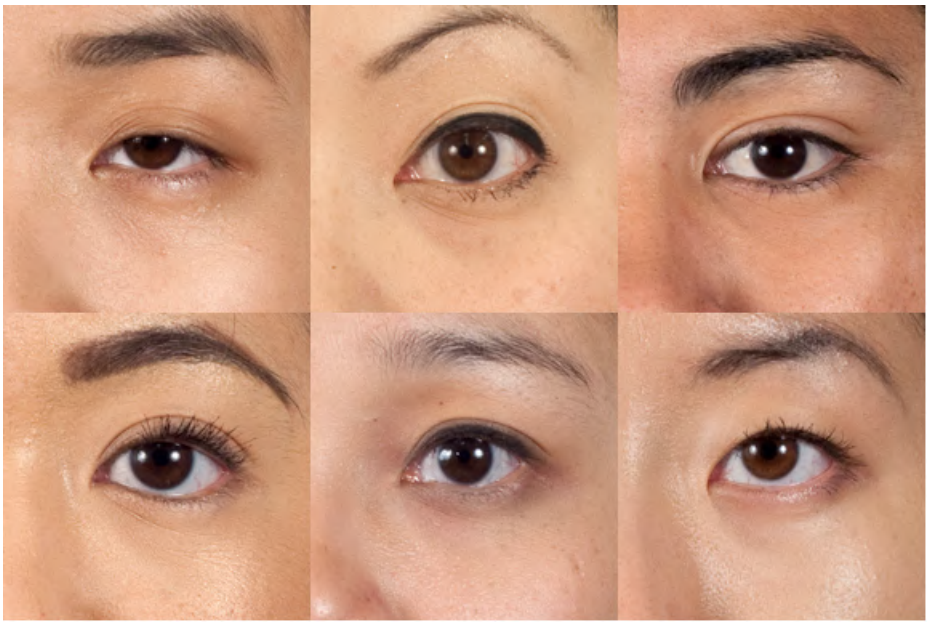


Figure 27: Eye samples generated from Chicago Face Database.

attack videos collected from 45 subjects. These video samples for each subject were captured in different illumination conditions and different places using the front cameras of six different mobile devices. In our experiments, we used only real video samples from all the mobile devices and individuals.

The **Radboud Faces Database** (RaFD) [74] is a high-quality face database containing pictures of eight emotional expressions from 67, primarily Caucasian individuals. The dataset consists of x number of samples equally distributed among all the subjects with 8 facial expressions: anger, disgust, fear, happiness, sadness, surprise, contempt, and neutral.

The **Replay-Mobile Database** [75] for face anti-spoofing detection consists of video clips from 40 subjects in different lighting conditions. The dataset consists of video

Figure 28: Eye samples generated from FACES facial expressions recognition database.

samples collected using a selfie camera of iPad Mini2 and an LG-G4 smartphone under five lighting conditions (controlled, adverse, direct, lateral, and diffuse). Our experiments used high-quality samples from 39 subjects only under controlled illumination conditions for our experiments.

Spoof in the Wild (SiW) [76] database provides live and spoof videos from 165 individual in a different distance, pose illumination, and expressions variations. For each subject, 8 live videos and 20 spoof videos are collected with a total of 4478 videos in the database. In our experiments, we considered eye samples from only live videos.

7.2.2 Generating Eye Crops

To generate the eye crops from the face images, we used the Dlib library [63] for face localization and detection. For eye localization, Dlib's 5-point face landmarks

Figure 29: Eye samples generated from Karolinska Directed Emotional Faces (KDEF) database.

detection model was used, and eye crops were generated such that the eye is in the center of the crop, and the width of the eye (from eye corner to eye corner) is 50% of the crop as shown in Figure 34.

In the case of face databases with video clips, we sampled 5 frames per second of the data for extracting eye crops.

In our training dataset, we flipped all the right-eye images horizontally and considered them as the new subjects, resulting in double the number of subjects. Total of $1544 (= 2 \times 772)$ subjects are used for training OcularNet-v2 model.

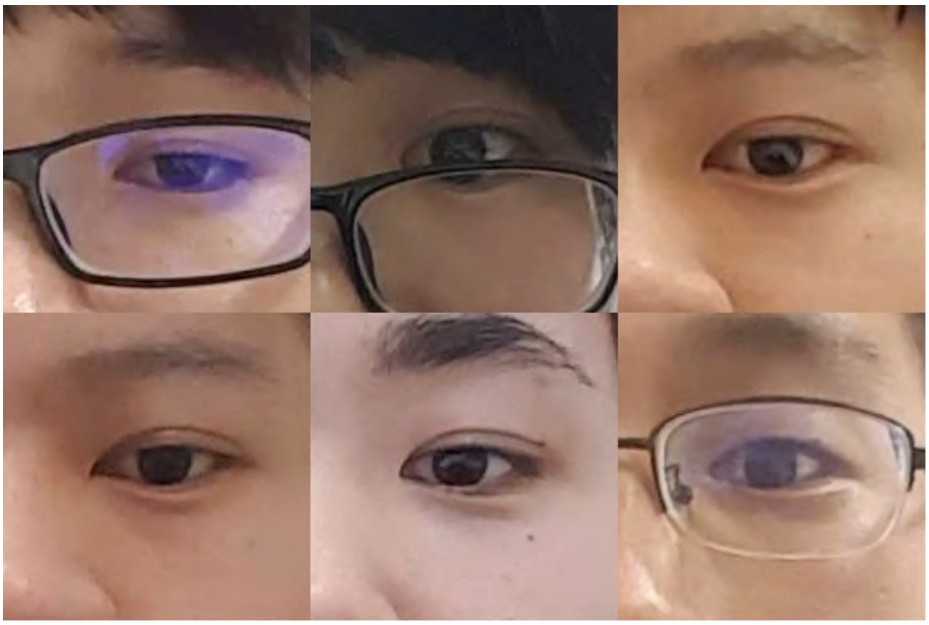


Figure 30: Eye samples generated from Oulu-NPU database.

7.3 Experimental Setup

To train and evaluate the OcularNet-v2 model with the new LOD-V dataset, we used the same experimental setup proposed in the previous chapter Section-6.3.

7.3.1 Training Protocol

The OcularNet-v2 model is trained with two different learning rates for layers trained from scratch and pre-trained layers from MobileNet-v2 trained with a lower learning rate for fine-tuning. Adam [52] optimizer was used for training our models with initial learning rates of 0.001 and 0.0001 for layers trained from scratch and for fine-tuning layers, respectively. The learning rate was reduced by 10 after the first 5 epochs and then after every 50 epoch. Models were trained using L2-normalized categorical cross-entropy

Figure 31: Eye samples generated from Radboud Faces Database (RaFD).

loss [57] for a total of 150 epochs with early stopping.

7.3.2 Test Datasets

Experiments are conducted on 6 datasets: VISOB, UBIRIS-V2, UBIPR, FERET, CROSS-EYED, and CASIA Twins datasets. Details of all the datasets are provided in Section-6.3.1. All the datasets follow similar testing protocol from Section-6.3.1 except for VISOB dataset. In our previous chapter, to train OcularNet-v2, we divided the dataset randomly for subject-independent evaluation into training and testing sets. However, as we train the model with the new LOD-V dataset, we used the whole VISOB VISIT-I dataset to test and compare the proposed model's performance with different literary techniques.

We evaluated the results using Equal Error Rate (EER%) [67], and Genuine Match

Figure 32: Eye samples generated from Replay-Mobile Database.

Rate (GMR%) at fixed False Match Rates (FMR%) [67] following the published measurement practices over the utilized datasets.

7.4 Results

We compared the performance of the OcularNet-v2 model trained on proposed LOD-V dataset with OcularNet-v2 trained on VISOB datasets along with $MOD-1$ and $MOD-3$ models which are discussed in section 6.2.3. To differentiate OcularNet-v2 trained on LOD-V with the one trained on VISOB, we use **OcularNet-v2 + LOD-V** for the model trained on LOD-V dataset.

All the experimental results on the tests datasets discussed in section-7.3.2 are divided into same spectrum cross-dataset evaluation and cross-spectrum evaluations.

Figure 33: Eye samples generated from Spoof in the Wild (SiW).

7.4.1 Cross-dataset Results

First, we report the cross-dataset evaluation results on the VISOB dataset in the same illumination and cross-illumination. In the same illumination evaluation, enrollment samples and verification samples are from the same lighting conditions. In contrast, in cross-illumination evaluation, enrollment samples are from office light, and verification samples are from all three lighting conditions (Office, dim light, daylight).

Tables 24 - 26 shows the same illumination evaluation results in GMR at 0.1% FMR. The table is divided into three categories, first, are results based on handcrafted features [30], second is the results from models trained in the subject-dependent environment, and finally, the results of the cross-dataset assessment. It can be seen that the

Table 24: Verification performance (GMR% at FAR= 10^{-3}) of existing models compared to our proposed model OcularNet-v2 on iPhone samples from VISOB Visit-I dataset.

Models	**iPhone**		
	Office light	**Day light**	**Dim light**
Hand Crafted Features [30]			
Block BSIF	30.09	44.23	37.99
Block HoG	0.45	0.15	1.13
BSIF	43.30	60.82	46.18
HoG	0.31	0.04	0.47
LPQ	3.12	1.82	7.22
Closed-Set Protocol			
MR Filters	89.46	91.69	92.54
Deep SparseFilters	87.62	89.65	89.55
VisobNet	99.67	99.71	99.82
ConvSRC	99.69	99.86	99.62
Open-Set Protocol			
ResNet + TL	81.15	87.76	90.17
Best Model from [42]	85.68	82.06	80.26
OcularNet-v2 + LOD-V	**94.26**	**94.82**	**96.96**

Table 25: Verification performance (GMR% at FAR= 10^{-3}) of existing models compared to our proposed model OcularNet-v2 on Samsung Note-4 samples from VISOB Visit-I dataset.

Models	**Note 4**		
	Office light	**Day light**	**Dim light**
Hand Crafted Features [30]			
Block BSIF	27.36	48.10	46.50
Block HoG	0.31	0.13	0.19
BSIF	36.97	58.66	58.88
HoG	0.29	0.10	0.25
LPQ	1.85	5.65	9.29
Closed-Set Protocol			
MR Filters	90.29	92.72	93.01
Deep SparseFilters	85.32	92.63	92.12
VisobNet	98.76	99.21	99.48
ConvSRC	98.85	99.39	99.61
Open-Set Protocol			
ResNet + TL	62.83	79.16	73.10
Best Model from [42]	82.38	79.55	75.03
OcularNet-v2 + LOD-V	**93.85**	**95.70**	**96.85**

Table 26: Verification performance (GMR% at FAR= 10^{-3}) of existing models compared to our proposed model OcularNet-v2 on Oppo N1 phone samples from VISOB Visit-I dataset.

Models	**Oppo**		
	Office light	**Day light**	**Dim light**
Hand Crafted Features [30]			
Block BSIF	24.96	47.81	39.20
Block HoG	0.63	0.43	0.52
BSIF	29.34	53.93	42.38
HoG	0.35	0.19	0.31
LPQ	3.42	2.70	4.02
Closed-Set Protocol			
MR Filters	93.03	92.63	93.92
Deep SparseFilters	83.79	97.10	87.29
VisobNet	99.23	99.65	99.85
ConvSRC	99.31	99.17	99.32
Open-Set Protocol			
ResNet + TL	69.34	75.32	81.62
Best Model from [42]	85.35	83.98	76.35
OcularNet-v2 + LOD-V	**95.73**	**95.75**	**97.05**

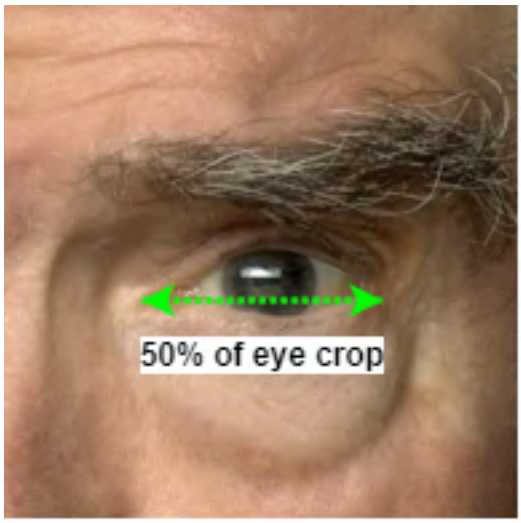


Figure 34: Generated eye crops for LOD-V dataset with width of eye of 50% of the eye crop size.

proposed model outperformed all the handcrafted feature models and cross-dataset evaluation by a large margin. Compared to subject-dependent environment models, even though they are trained and tested on the same dataset, the proposed model still achieves substantial performance with an average GMR of 96.67% at 0.1% FMR across all devices and illumination conditions. We calculated EER($\%$) for VISOB cross-illumination evaluation, as shown in Table 27. It can be seen that the proposed model outperformed all the cross-dataset evaluation models with a significant reduction in error rate. Compared to the models trained in a subject-dependent environment, the proposed model achieves the third-best error next to the MR filters [30] and deep sparse filters [16] methods.

Table 28 shows EER($\%$), and GMR at 1% FMR for the trained model on two data splits in the UBIRIS-V2 dataset. $D1 - set$ contains samples collected from the person standing within 6 to 8 meters from the camera, capturing eye images with a consistent

Table 27: Cross illumination performance (EER%) on VISOB dataset with enrollments from Office lighting and verification samples from all the lighting conditions.

Models	iPhone			Oppo			Samsung		
	O-O	**O-DAY**	**O-DIM**	**O-O**	**O-DAY**	**O-DIM**	**O-O**	**O-DAY**	**O-DIM**
subject-dependent Protocol									
MR Filters	0.06	0.13	0.20	0.04	0.10	0.09	0.05	0.13	0.10
Deep Sparse Filtered	0.48	1.82	1.45	0.63	1.90	3.34	0.49	2.50	4.25
ANU	10.36	11.03	16.64	16.01	14.75	18.24	9.10	13.69	19.57
IIITG	19.29	32.93	45.34	19.79	38.24	42.59	18.65	34.29	40.21
subject-independent and Cross-dataset Protocol									
ResNet + TL	3.72	12.50	14.72	7.29	21.64	19.36	7.75	17.18	25.94
Best Model from [42]	5.46	10.38	14.76	8.33	14.82	16.69	7.35	12.99	19.33
OcularNet-v2 + LOD-V	**1.83**	**4.66**	**9.66**	**1.40**	**4.74**	**11.84**	**2.16**	**5.42**	**9.55**

Table 28: EER(%) and GMR(%) at 1% FMR for the Ubiris-V2 dataset, comparing the proposed method with reported methods in the literature on two different datasets. *Note: Results for PRIWIS are from [4]*

Model	**EER (%)**	**GMR (%) @ 1% FMR**
Distance : 6 to 8 meters (D1-set)		
ResNet-50 [59]	17.91	-
OcularNet [59]	9.77	68.58
MOD-1	7.78	73.74
MOD-3	8.69	70.76
OcularNet-v2	7.65	72.3
OcularNet-v2 + LOD-V	**4.75**	**86.21**
Distance : all distances (D2-set)		
PRWIS by Proenca et al. [18]	22.95	-
Model by Zhao et al. [4]	10.05	-
MOD-1	10.00	65.06
MOD-3	10.33	65.36
OcularNet-v2	9.1	69.82
OcularNet-v2 + LOD-V	**7.81**	**63.11**

Table 29: EER(%) for multi-distance evaluation on the UBIPR dataset, with enrollment samples from one distance and verification samples from the other four distances.

Enrollments at Distance	**MOD - 1**	**MOD - 3**	**OcularNet-v2**	**OcularNet-v2 + LOD-V**
D1	4.16	5.54	4.43	**2.63**
D2	3.49	4.57	3.87	**2.44**
D3	3.73	5.24	4.18	**2.94**
D4	4.18	5.87	4.35	**3.27**
D5	3.97	5.21	4.18	**2.64**

quality periocular region. Whereas $D2 - set$ contains all the samples from the UBIRIS-V2 dataset. From Table 28, we can see that compared OcularNet-v2 trained on partial VISOB dataset, in $D1 - set$, the same model trained on the LOD-V dataset achieved close to 3% reduction in error rate with more than 14% improvement in GMR at 1% FMR. In $D2 - Set$, OcularNet-v2 trained on LOD-V reduced the error rate by 1.3%, and however, GMR at 1% FMR is reduced by 6.71%.

UBIPR dataset contains samples collected at five distances. In our experiments, we chose samples from a certain distance and verification samples from the remaining distances. Table 29 shows the performance of all the trained models in EER($\%$). It can be seen that the OcularNet-v2 trained with LOD-V reduces EER by 1.42% compared to the model trained on a partial VISOB dataset. It can also be seen that the OcularNet-v2 + LOD-V outperforms $MOD - 1$ model which is $3.3\times$ larger in parameters.

As shown in the Table 30, in FERET dataset evaluation, the OcualrNet-V2 + LOD-V reduces error rate by $2.3\times$ compared to the same model trained on VISOB dataset, while GMR at 0.1% FMR increase by 17.24%.

Table 30: EER(%), and GMR at 0.1% FMR, for FERET dataset and CASIA-TWINS dataset using (a) all data and (b) twins only data.

-	FERET dataset		CASIA-TWINS (All data)		CASIA-TWINS (Only Twins)	
Model	**EER**	**GMR @ 0.1% FMR**	**EER**	**GMR @ 0.1% FMR**	**EER**	**GMR @ 0.1% FMR**
MOD-1	6.79	72.43	12.02	56.73	11.75	60.07
MOD-3	7.43	69.85	12.83	58.70	13.07	55.70
OcularNet-v2	6.06	72.96	11.29	65.18	9.41	69.18
OcularNet-v2 + LOD-V	**2.46**	**90.20**	**9.53**	**67.12**	**9.95**	**61.49**

7.4.2 Cross-Spectral Results

EER($\%$) and GMR at 1% FMR for the CROSS-EYED dataset in Iris only and periocular only dataset are shown in Table 31. It can be seen that for Iris only dataset, the OcularNet-v2 model trained on the LOD-V error rate increased by 4.11%, and while GMR remained the same. However, in the periocular dataset, OcularNet-v2 trained on LOD-V reduced EER($\%$) to 0.73% from 0.94% for the same model trained on VISOB and GMR is reduced by more than $2.5\times$. From Table 31, it can also be seen that in for periocular dataset, the proposed model outperformed the best model $HH1$ in both EER($\%$) and GMR at 1% FMR.

In the NIR spectrum test dataset, CASIA-TWINS, we calculated EER($\%$) and GMR at 0.1% FMR for (a) all the dataset and (b) between only twin pairs. The Table 30 shows that the EER reduced by 1.76% with a 1.94% increase in GMR. However, in the case of twins, only dataset, model trained on the LOD-V dataset achieved slightly lower

Table 31: EER(%), and GMR(%) at 1% FMR for CrossEyed dataset. The first three methods for the iris and periocular dataset are the top 3 performing from CROSS-EYED 2017 competition [5]

Model	**EER (%)**	**GMR (%) @ 1% FMR**
Iris Dataset		
NTNU4	0.05	0.00
NTNU3	5.58	8.43
NTNU1	6.19	8.81
ResNet-50 [59]	21.52	-
OcularNet [59]	14.95	-
MOD-1	3.80	13.91
MOD - 3	3.75	14.74
OcularNet-v2	2.71	7.86
OcularNet-v2 + LOD-V	**6.82**	**8.86**
Periocular Dataset		
HH1	0.82	0.74
NTNU1	1.59	1.86
IDIAP2	1.65	2.03
MOD-1	2.40	5.21
MOD - 3	2.19	4.38
OcularNet-v2	0.94	0.83
OcularNet-v2 + LOD-V	**0.73**	**0.31**

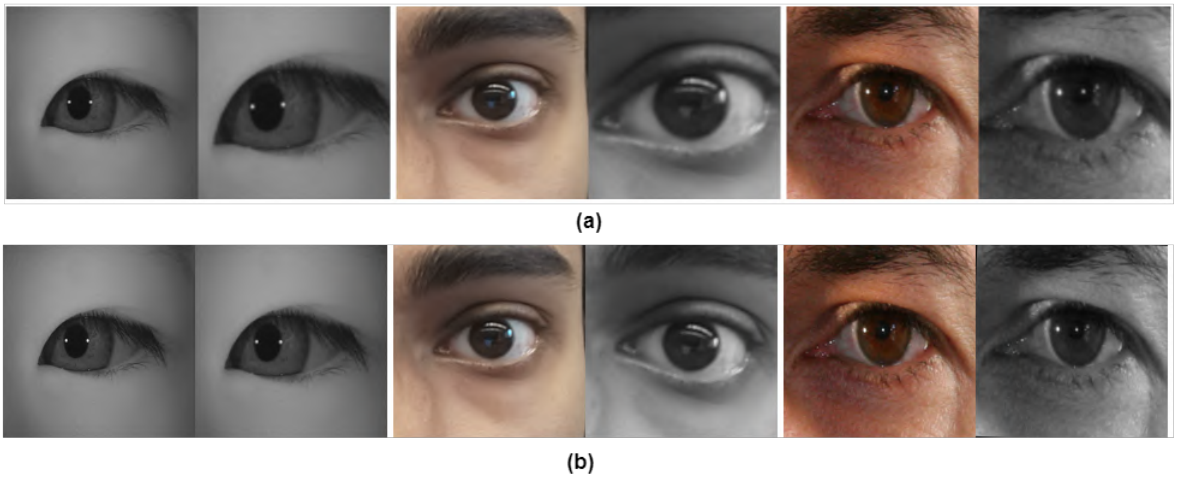


Figure 35: (a) Shows ROI's generated by OcularNet-v2 when trained on VISOB dataset. (b) Shows ROI's generated by OcularNet-v2 when trained on LOD-V dataset where the model preferred larger periocular region. Note: for each image pair, the left image is STN model input, and the right one is the output.

performance than the model trained on the VISOB dataset.

7.4.3 Visual Analysis of the ROI Model

From Figure 35, it can be seen that the proposed model, when trained with a partial VISOB dataset with only 200 subjects generated ROI's with the eye at the center, horizontally aligned, and the width of the eye covering about 90% of the crop. However, when we train the OcularNet-v2 model from scratch with a larger LOD-V dataset with 772 subjects, the model generated ROI's with similarly eyes centered and with a larger periocular region. This shows that the model needed a larger eye region to improve the generalizability with a large number of subjects.

7.4.4 Key Findings

1. OcularNet-v2 ($MOD - 3 + STN$) model trained on VISOB did not show any performance improvements compared to only $MOD - 3$ on UBIPR, FERET, and $D1 - set$ in UBIRIS-V2 datasets, where images are well aligned and centered. However, with the introduction of a large training dataset, LOD-V, we can see considerable performance improvements.

2. However, we saw a drop in performance in datasets such as CROSS-EYED iris only and $D2 - set$ in UBIRIS-V2, where some sample's eye crop have the only eye and with the less-to-no periocular region is available. Training models on much tighter crops can mitigate this issue.

7.5 Conclusion

This chapter proposed a new dataset LOD-V with 772 unique subjects and more than $200K$ samples to avoid division of test databases for subject-independent evaluation and provide access to Large scale verification evaluation to show real-world performance. We show that the OcularNet-v2 model trained on the LOD-V dataset achieved significant performance improvements compared to the same model trained on partial VISOB dataset various datasets in the cross-dataset evaluation and the cross-illumination evaluations showing up to 3% reduction in error rates.

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 Summary Of Contributions

With advancements of deep learning technology mobile devices, many proposed improving matching accuracy of ocular biometrics. Many of these early works implemented using deep learning techniques to achieve higher matching performance with really low error rates. However, the drawback of these implementations is that they operated on a subject-dependent dataset where the model is trained and tested on the same subjects. These models also tend to be large, making them difficult to deploy on a mobile device efficiently.

Our work conducted an extensive evaluation to propose a deep learning-based matching model achieving higher matching performance in subject in-depended evaluation while being efficient enough for operating on mobile phones in real-time. Our proposed model shows comparative performance, even in cross-dataset evaluation.

Chapter 4 conducted a large scale evaluation on different deep learning models in subject-independent evaluation to find a model that achieves better matching performance while being computationally efficient. Experimental results on the VISOB dataset show that it is possible to attain larger models such as ResNet-50 even with smaller models based on MobileNet architectures. From these findings, In Chapter-5, we proposed OcularNet, a patch-based CNN architecture for mobile ocular biometric matching. The

proposed model OcularNet consists of 6 small CNN models named PatchCNN, which extract features from overlapping patches extracted from the eye region. OcularNet, while being $15.6\times$ smaller in size compared to ResNet-50, is our performance with up to 8% lower error rates in subject-independent, cross-dataset and also in cross-illumination evaluations.

However, the main limitation of OcularNet models is that it requires an ROI detector to obtain accurate eye region localization to extract the overlapping regions for feature extraction. As the model depends on an off-the-shelf ROI detector, the model fails to match if the ROI detector fails. To overcome this problem, we propose OcularNet-v2 in Chapter 6, where the feature extraction model is trained along a size ROI detection model to achieve better performance while being computationally efficient. OcularNet-v2 is consists of STN based ROI detector trained in conjunction with a modified MobileNet-V2 model while being $36\times$ smaller than the ResNet-50 model and takes only $37.6ms$ of execution on an embedded device. The proposed is evaluated on multiple subject-independent, cross-dataset, and cross-illumination datasets showing up to $7\times$ lower error rates than the models presented in the literature.

Finally, we proposed a large ocular biometrics dataset in the visible spectrum (LOD-V) for large-scale deep learning model training. LOD-V dataset is prepared by combining high-quality face datasets used for facial expression and anti-spoofing analysis. The proposed dataset consists of $217K$ image samples from 772 subjects. We shown the OcularNet-v2 model trained using the LOD-V dataset achieves up to $2.5\times$ reduction in error rates.

8.2 Limitations

As a larger dataset is introduced in training OcularNet-v2 model in Chapter-7 with LOD-V, it can be noticed that ROI's detection in the model preferred to have more periocular region than the model trained on 200 subject VISOB dataset, as shown in Figure 35. Because of this, in the iris only CROSS-EYED dataset, with samples having no periocular region, the error rate increased by 4%. This shows that the model has difficulty generalizing well with the dataset iris only datasets with less to no periocular region.

8.3 Future Works

In Chapter 6, the OcularNet-v2 model consists of a modified MobileNet-V2 model where we were able to change an input and remove a significant amount of layers while maintaining the performance. In our future work, we will be performing this model modification analysis on the different types of architectures to study the performance effect of removing multiple layers and trying various input features. With advancements in AutoML for custom model search from scratch, we want to conduct a thorough evaluation to see if it is possible to construct efficient models for ocular biometrics, which competes with the existing architectures designed for general-purpose vision datasets such as ImageNet.

In our work, experiments are conducted on subject independent, cross-dataset, and cross-illumination. However, most of the datasets available have samples collected on the same day, with few datasets collected in multiple sessions focusing on short-term

verification evaluation. Even though the VISOB dataset is collected in multiple visits, the dataset concentrates on same-day matching performance only. To overcome this, we recently proposed VISOB 2.0 [77], mainly focusing on long-term verification evaluation of biometric models. In future work, we will be conducting a large-scale assessment of different methods and the performance change while going from short-term verification to long-term verification.

APPENDIX A

SUPPLEMENTARY MATERIALS FOR CHAPTER 4

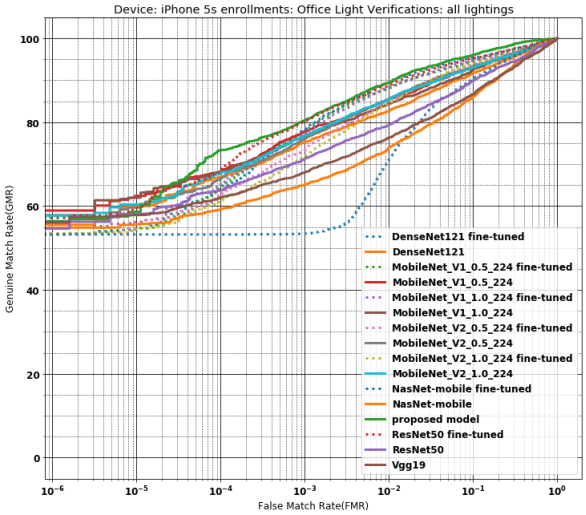


Figure 36: ROC curves for various deep learning models for enrollments in office lighting and verification samples from all the lighting conditions for samples in DATA-B set from iPhone 5s device.

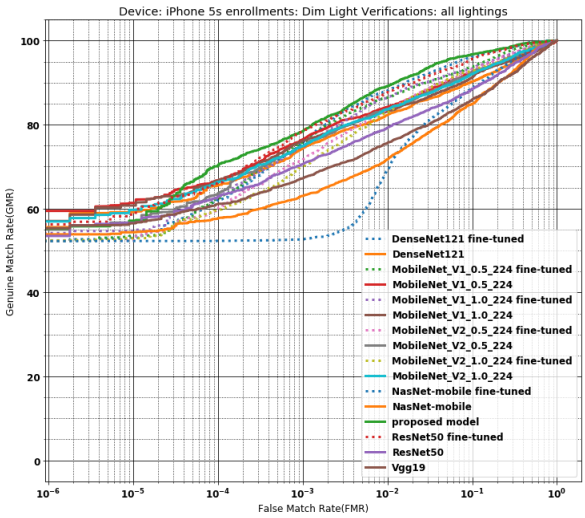


Figure 37: ROC curves for various deep learning models for enrollments in dim office lighting and verification samples from all the lighting conditions for samples in DATA-B set from iPhone 5s device.

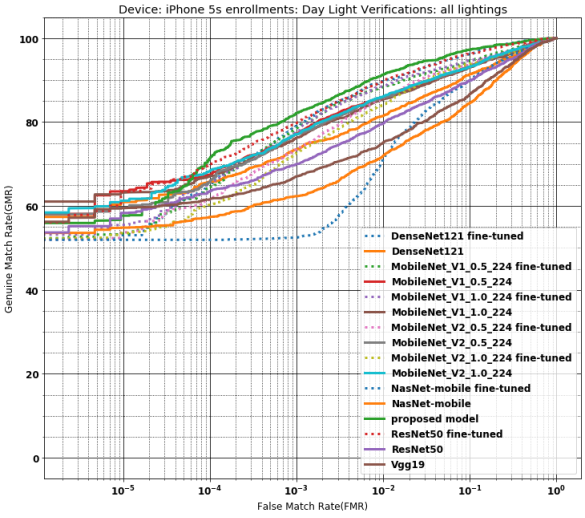


Figure 38: ROC curves for various deep learning models for enrollments in outdoor day lighting and verification samples from all the lighting conditions for samples in DATA-B set from iPhone 5s device.

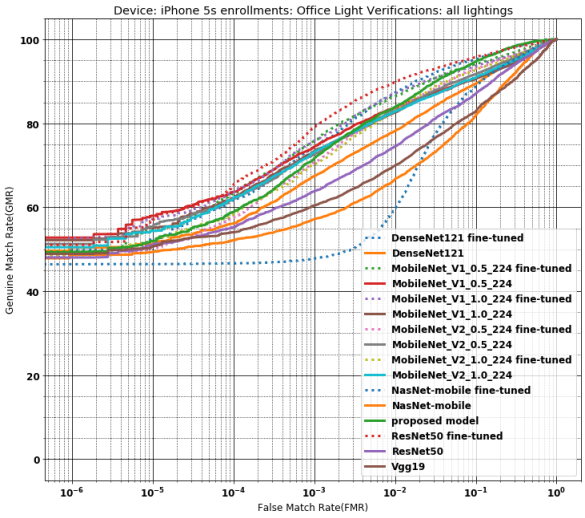


Figure 39: ROC curves for various deep learning models for enrollments in office lighting and verification samples from all the lighting conditions for samples in DATA-B set from iPhone 5s device.

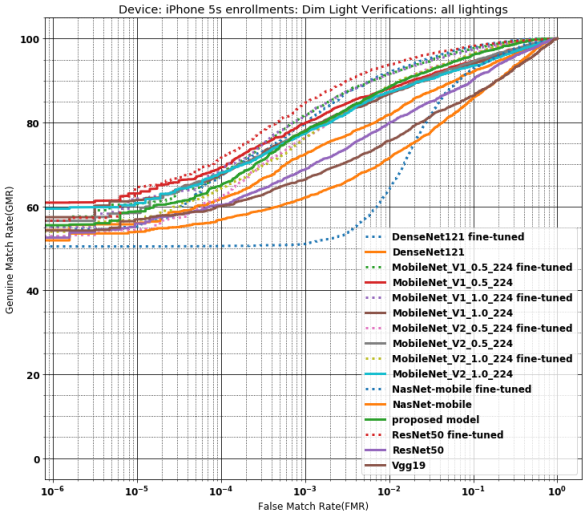


Figure 40: ROC curves for various deep learning models for enrollments in dim office lighting and verification samples from all the lighting conditions for samples in DATA-C set from iPhone 5s device.

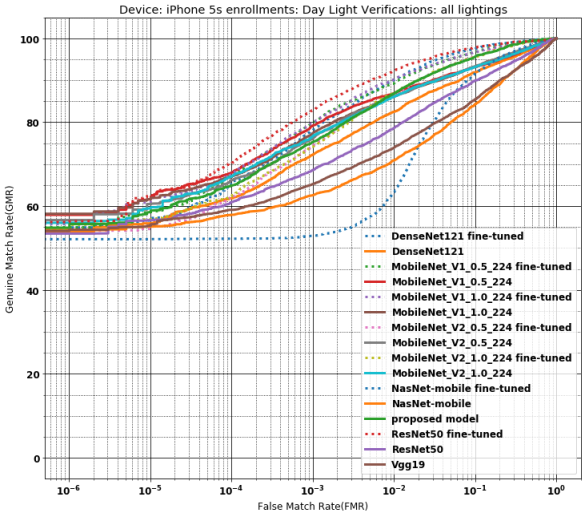


Figure 41: ROC curves for various deep learning models for enrollments in outdoor day lighting and verification samples from all the lighting conditions for samples in DATA-C set from iPhone 5s device.

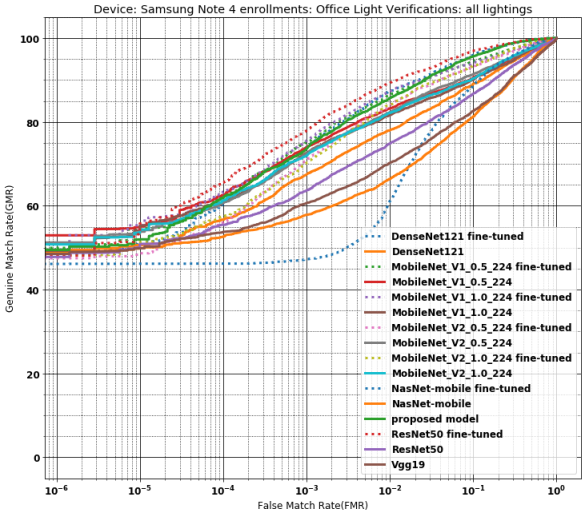


Figure 42: ROC curves for various deep learning models for enrollments in office lighting and verification samples from all the lighting conditions for samples in DATA-B set from Samsung Note 4 device.

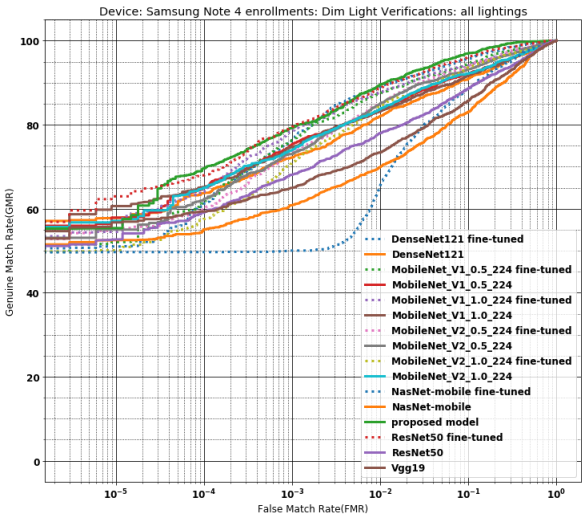


Figure 43: ROC curves for various deep learning models for enrollments in dim office lighting and verification samples from all the lighting conditions for samples in DATA-B set from Samsung Note 4 device.

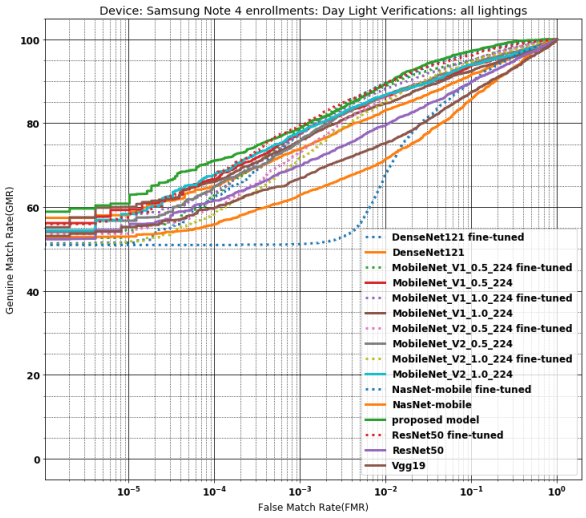


Figure 44: ROC curves for various deep learning models for enrollments in outdoor day lighting and verification samples from all the lighting conditions for samples in DATA-B set from Samsung Note 4 device.

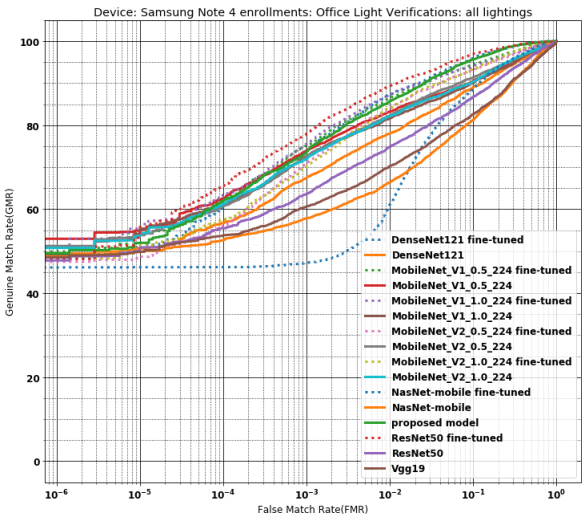


Figure 45: ROC curves for various deep learning models for enrollments in office lighting and verification samples from all the lighting conditions for samples in DATA-B set from Samsung Note 4 device.

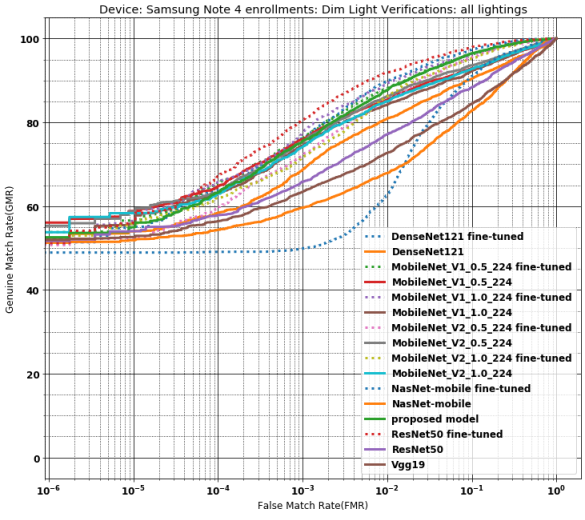


Figure 46: ROC curves for various deep learning models for enrollments in dim office lighting and verification samples from all the lighting conditions for samples in DATA-C set from Samsung Note 4 device.

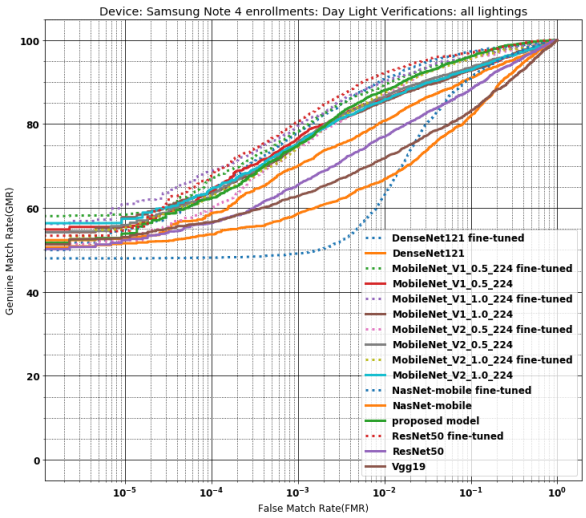


Figure 47: ROC curves for various deep learning models for enrollments in outdoor day lighting and verification samples from all the lighting conditions for samples in DATA-C set from Samsung Note 4 device.

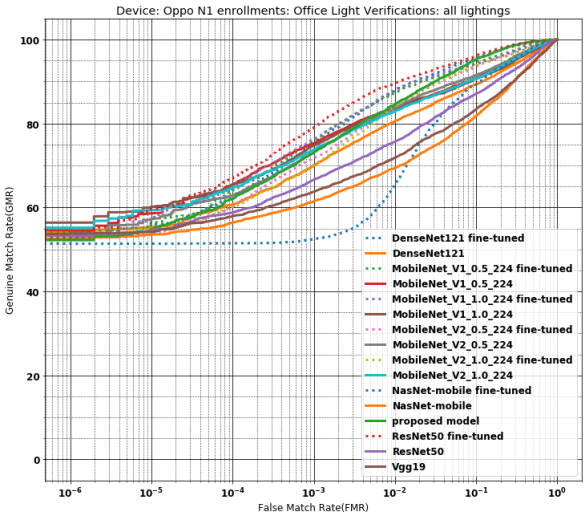


Figure 48: ROC curves for various deep learning models for enrollments in office lighting and verification samples from all the lighting conditions for samples in DATA-B set from Oppo N1 device.

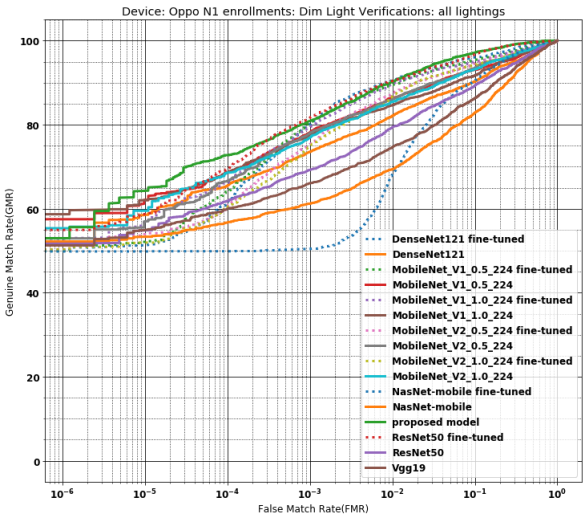


Figure 49: ROC curves for various deep learning models for enrollments in dim office lighting and verification samples from all the lighting conditions for samples in DATA-B set from Oppo N1 device.

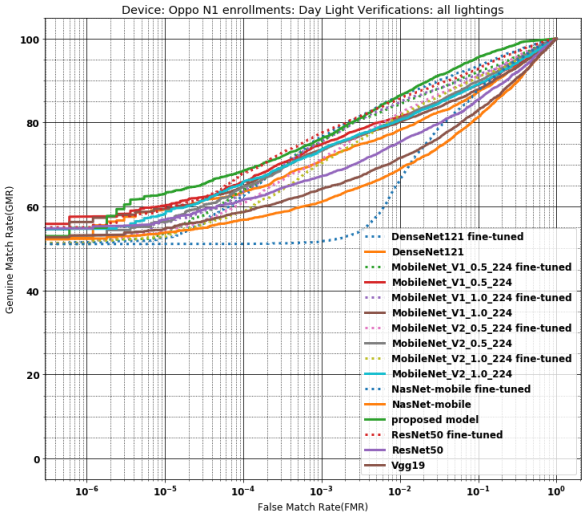


Figure 50: ROC curves for various deep learning models for enrollments in outdoor day lighting and verification samples from all the lighting conditions for samples in DATA-B set from Oppo N1 device.

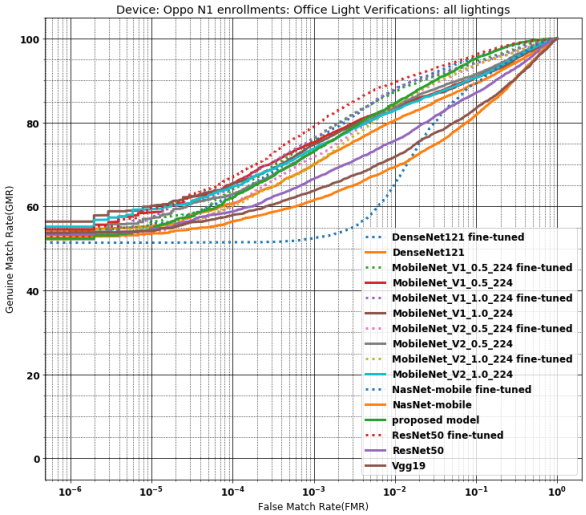


Figure 51: ROC curves for various deep learning models for enrollments in office lighting and verification samples from all the lighting conditions for samples in DATA-B set from Oppo N1 device.

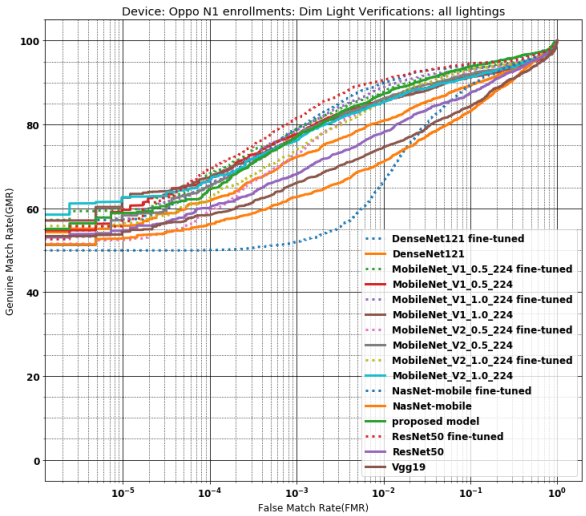


Figure 52: ROC curves for various deep learning models for enrollments in dim office lighting and verification samples from all the lighting conditions for samples in DATA-C set from Oppo N1 device.

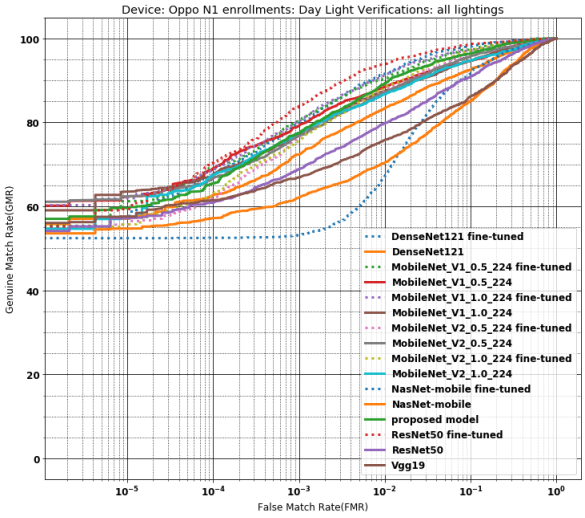


Figure 53: ROC curves for various deep learning models for enrollments in outdoor day lighting and verification samples from all the lighting conditions for samples in DATA-C set from Oppo N1 device.

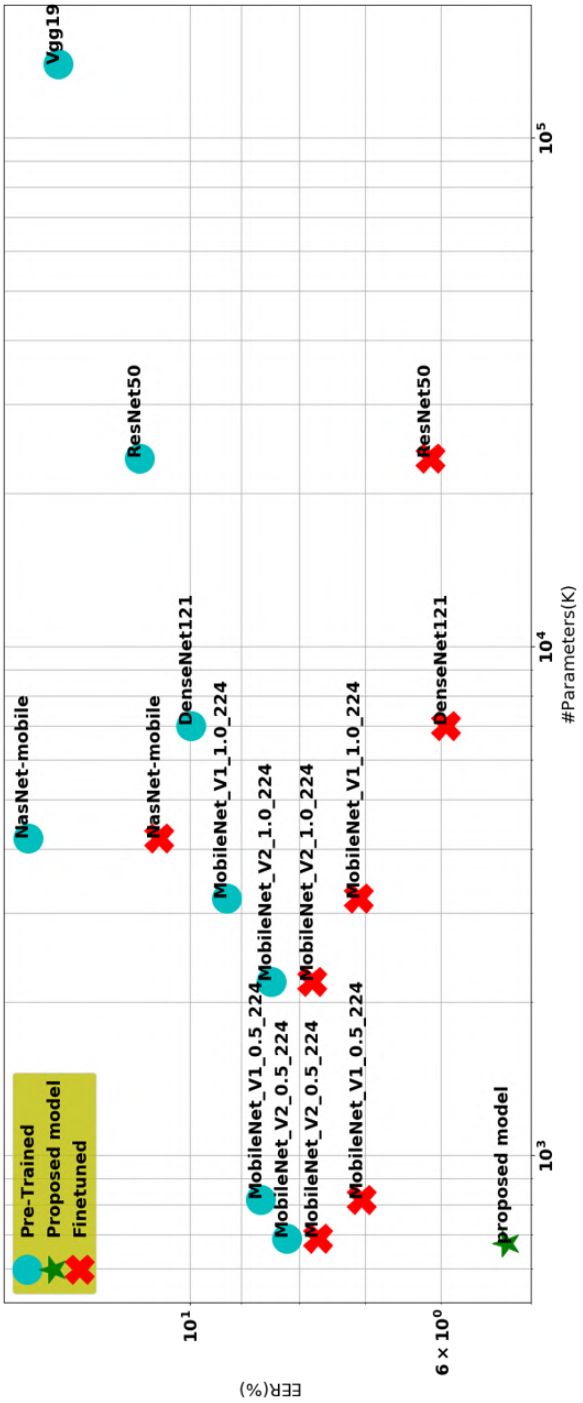


Figure 54: Comparing EER(%) with number of parameters for proposed model with state of the art deep learning models on DATA-B evaluation set. EER(%) is calculated by taking average of the lighting and device results.

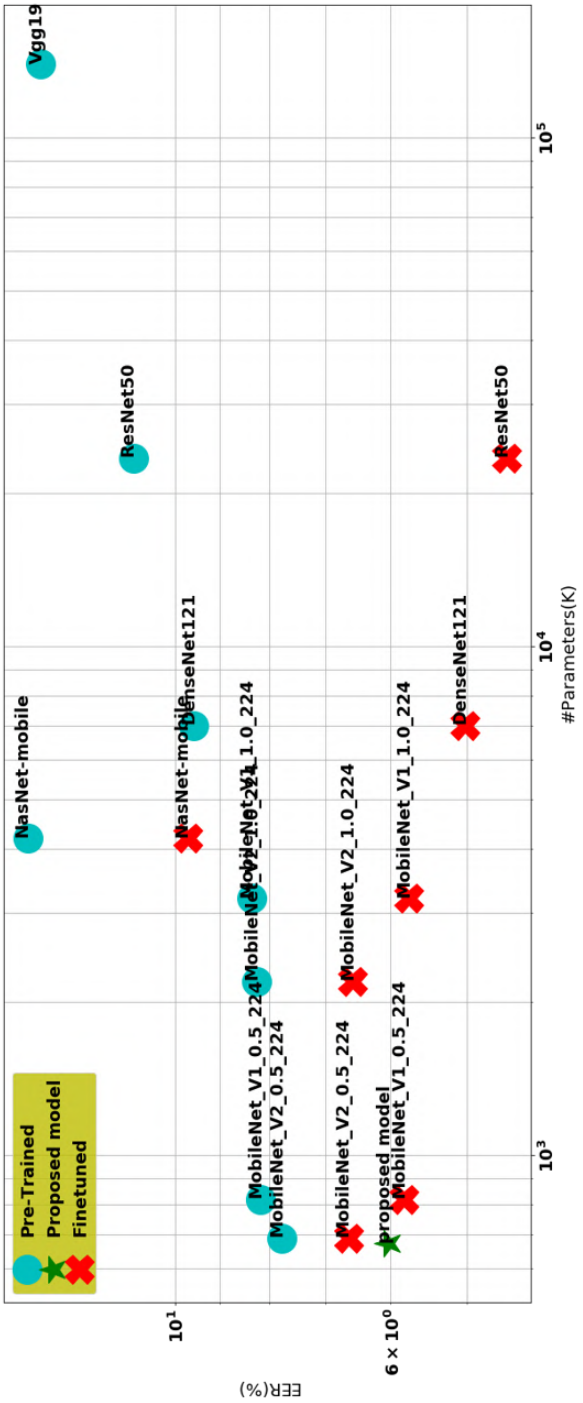


Figure 55: Comparing EER(%) with number of parameters for proposed model with state of the art deep learning models on DATA-C evaluation set. EER(%) is calculated by taking average of the lighting and device results.

APPENDIX B

SUPPLEMENTARY MATERIALS FOR CHAPTER 5

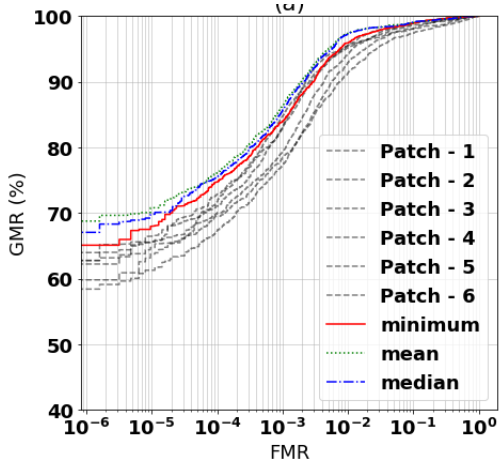


Figure 56: ROC curves for OcularNet model along with all the patches and fusing them using mean, median and max techniques. Enrollments in office lighting and verification samples from all the lighting conditions from VISOB Visit-I dataset iPhone device samples.

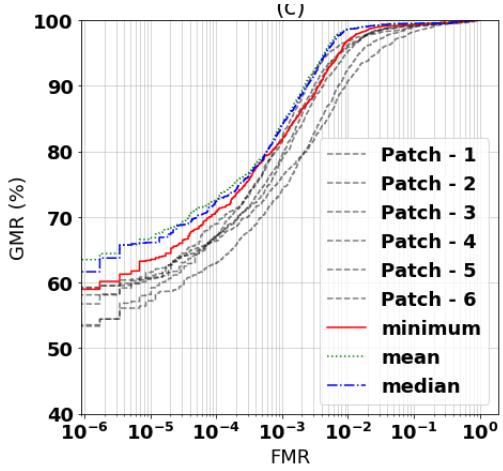


Figure 57: ROC curves for OcularNet model along with all the patches and fusing them using mean, median and max techniques. Enrollments in office lighting and verification samples from all the lighting conditions from VISOB Visit-I dataset Samsung Note-4 device samples.

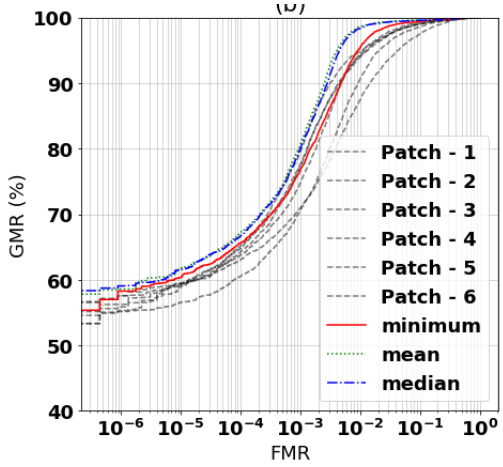


Figure 58: ROC curves for OcularNet model along with all the patches and fusing them using mean, median and max techniques. Enrollments in office lighting and verification samples from all the lighting conditions from VISOB Visit-I dataset Oppo N1 device samples.

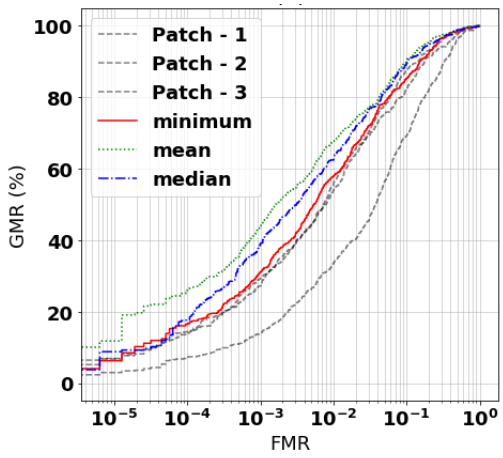


Figure 59: ROC curves for OcularNet model along with all the patches and fusing them using mean, median and max techniques on UBIRIS-V1 dataset.

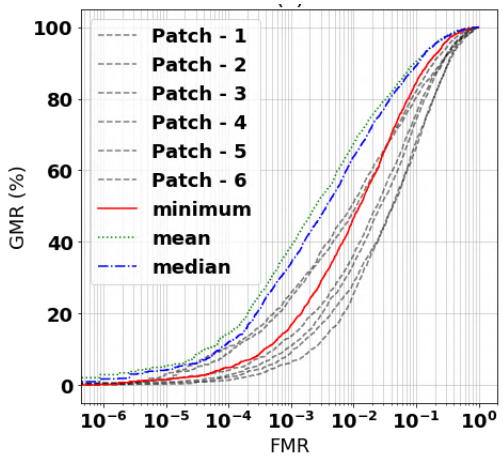


Figure 60: ROC curves for OcularNet model along with all the patches and fusing them using mean, median and max techniques on UBIRIS-V2 dataset for samples collected at 6 to 8 meters away from camera.

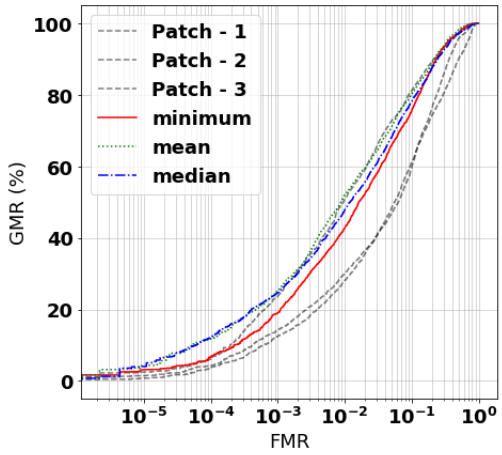


Figure 61: ROC curves for OcularNet model along with all the patches and fusing them using mean, median and max techniques on cross-eyed iris dataset.

APPENDIX C

SUPPLEMENTARY MATERIALS FOR CHAPTER 6

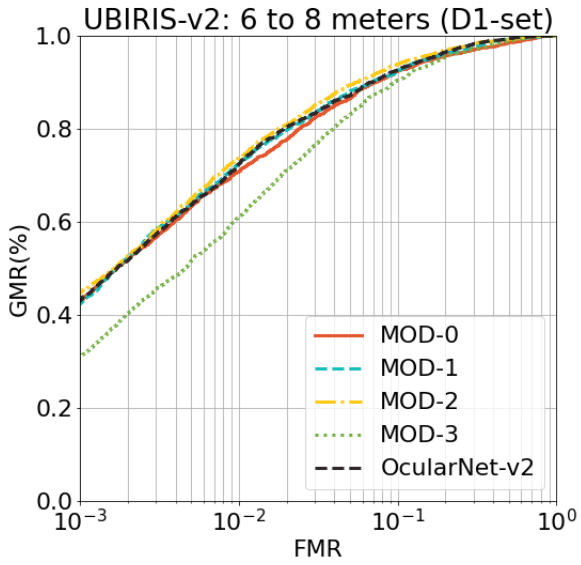


Figure 62: ROC curves for OcularNet-v2 model along with all the modifications of MobileNet-V2 model (MOD-0 to MOD-3) for UBIRIS-V2 at 6-8 meters distance (D1-set) dataset.

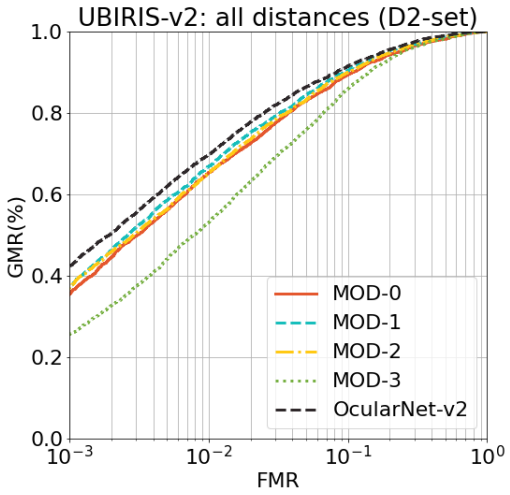


Figure 63: ROC curves for OcularNet-v2 model along with all the modifications of MobileNet-V2 model (MOD-0 to MOD-3) for UBIRIS-V2 at all distance (D2-set) dataset.

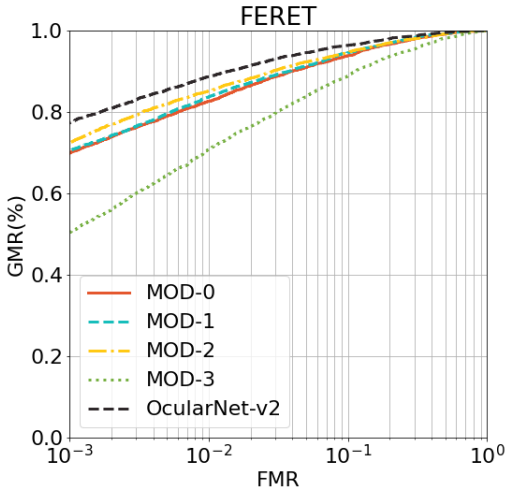


Figure 64: ROC curves for OcularNet-v2 model along with all the modifications of MobileNet-V2 model (MOD-0 to MOD-3) for FERET dataset.

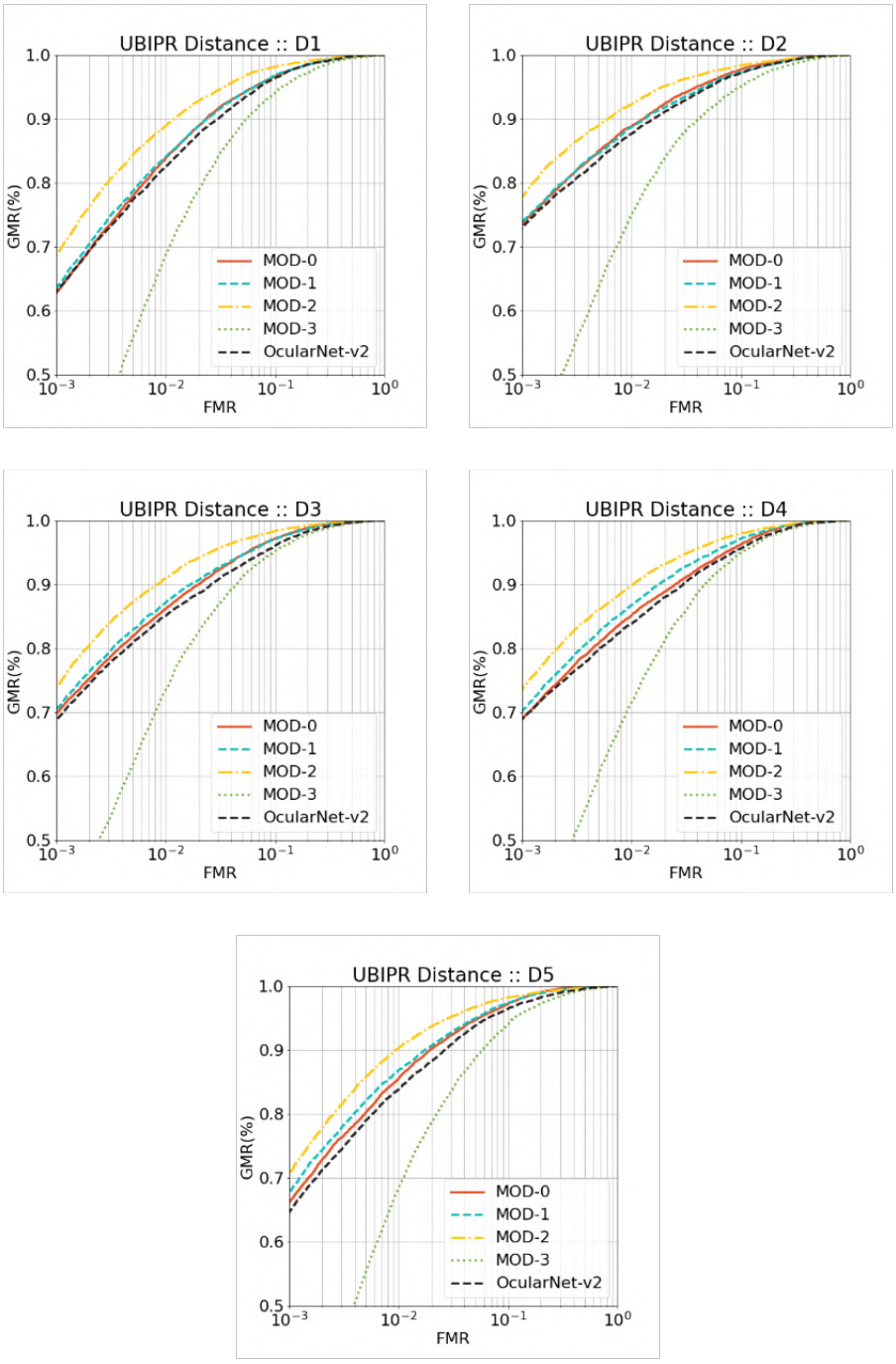


Figure 65: ROC curves for OcularNet-v2 model along with all the modifications of MobileNet-V2 model (MOD-0 to MOD-3) for UBIPR dataset with enrollments in a specific distance and verification samples in all remaining datasets.

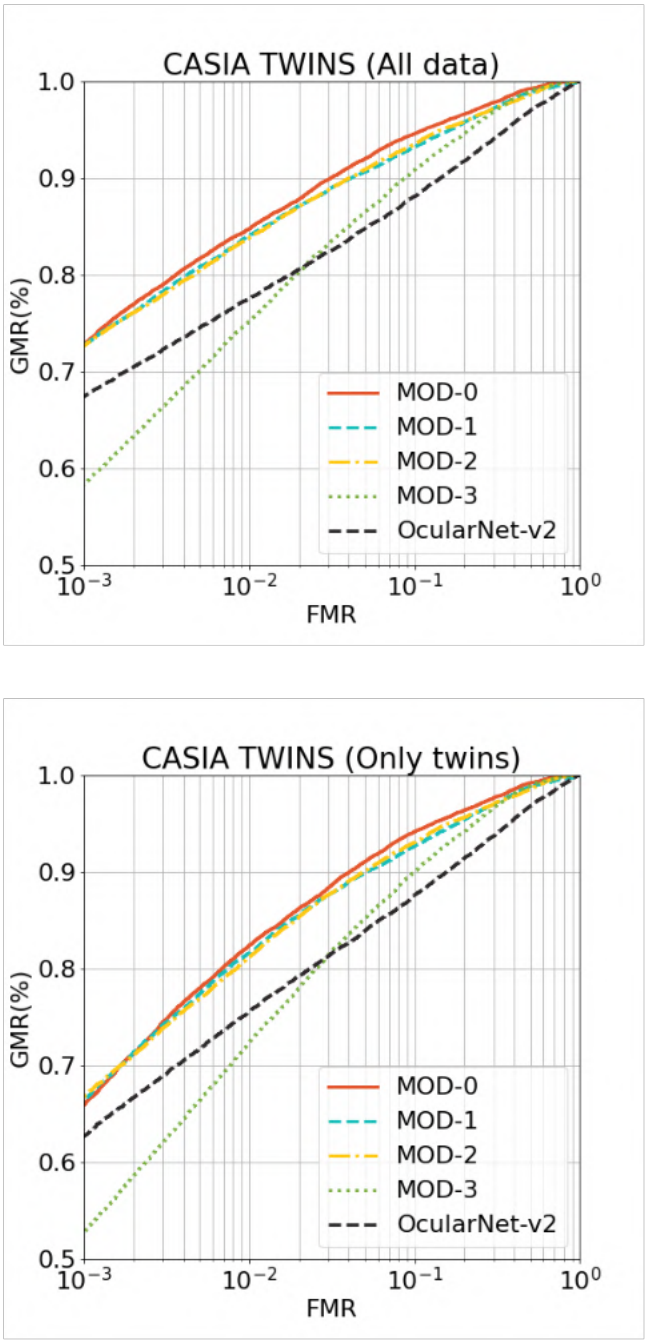


Figure 66: ROC curves for OcularNet-v2 model along with all the modifications of MobileNet-V2 model (MOD-0 to MOD-3) for CASIA TWINS dataset.

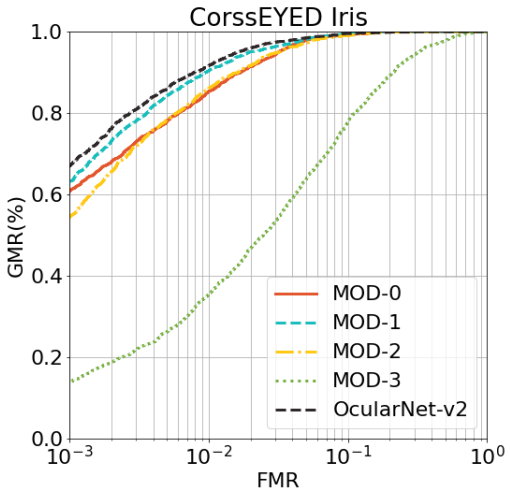


Figure 67: ROC curves for OcularNet-v2 model along with all the modifications of MobileNet-V2 model (MOD-0 to MOD-3) for CrossEYED iris only dataset.

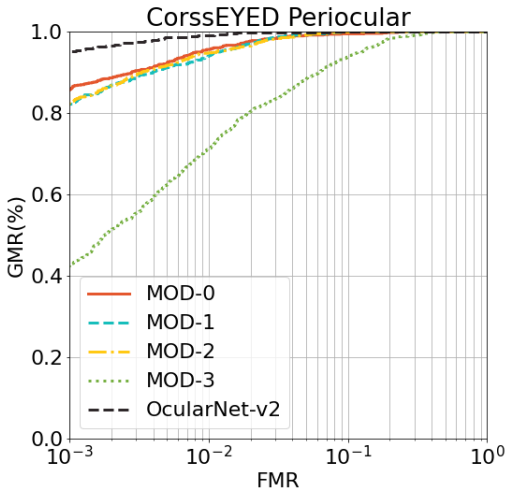


Figure 68: ROC curves for OcularNet-v2 model along with all the modifications of MobileNet-V2 model (MOD-0 to MOD-3) for CrossEYED periocular only dataset.

APPENDIX D

SUPPLEMENTARY MATERIALS FOR CHAPTER 7

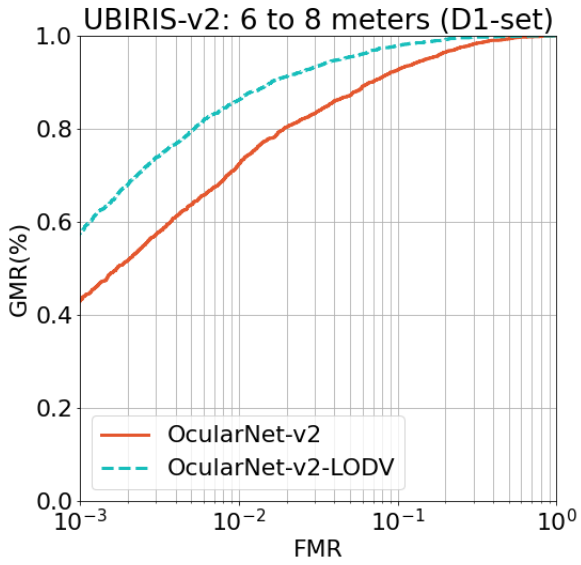


Figure 69: ROC curve of OcularNet-v2 model compared to OcularNet-v2 trained with LOD-V for UBIRIS-V2 at 6-8 meters distance (D1-set) dataset.

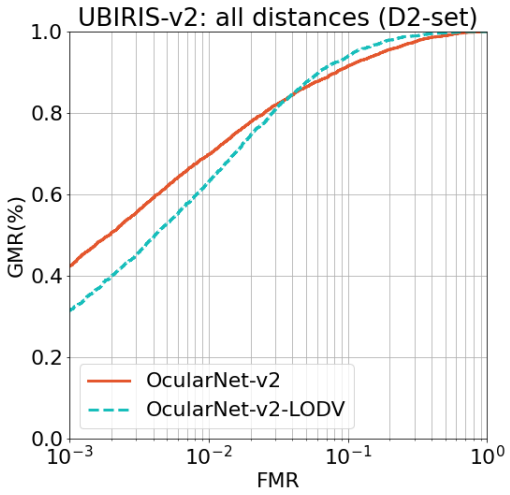


Figure 70: ROC curve of OcularNet-v2 model compared to OcularNet-v2 trained with LOD-V for UBIRIS-V2 at all distance (D2-set) dataset.

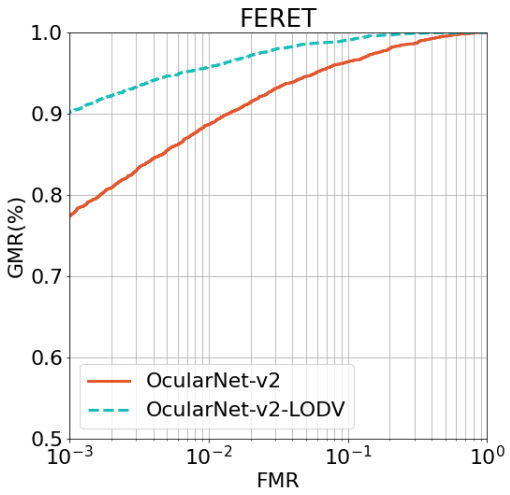


Figure 71: ROC curve of OcularNet-v2 model compared to OcularNet-v2 trained with LOD-V for FERET dataset.

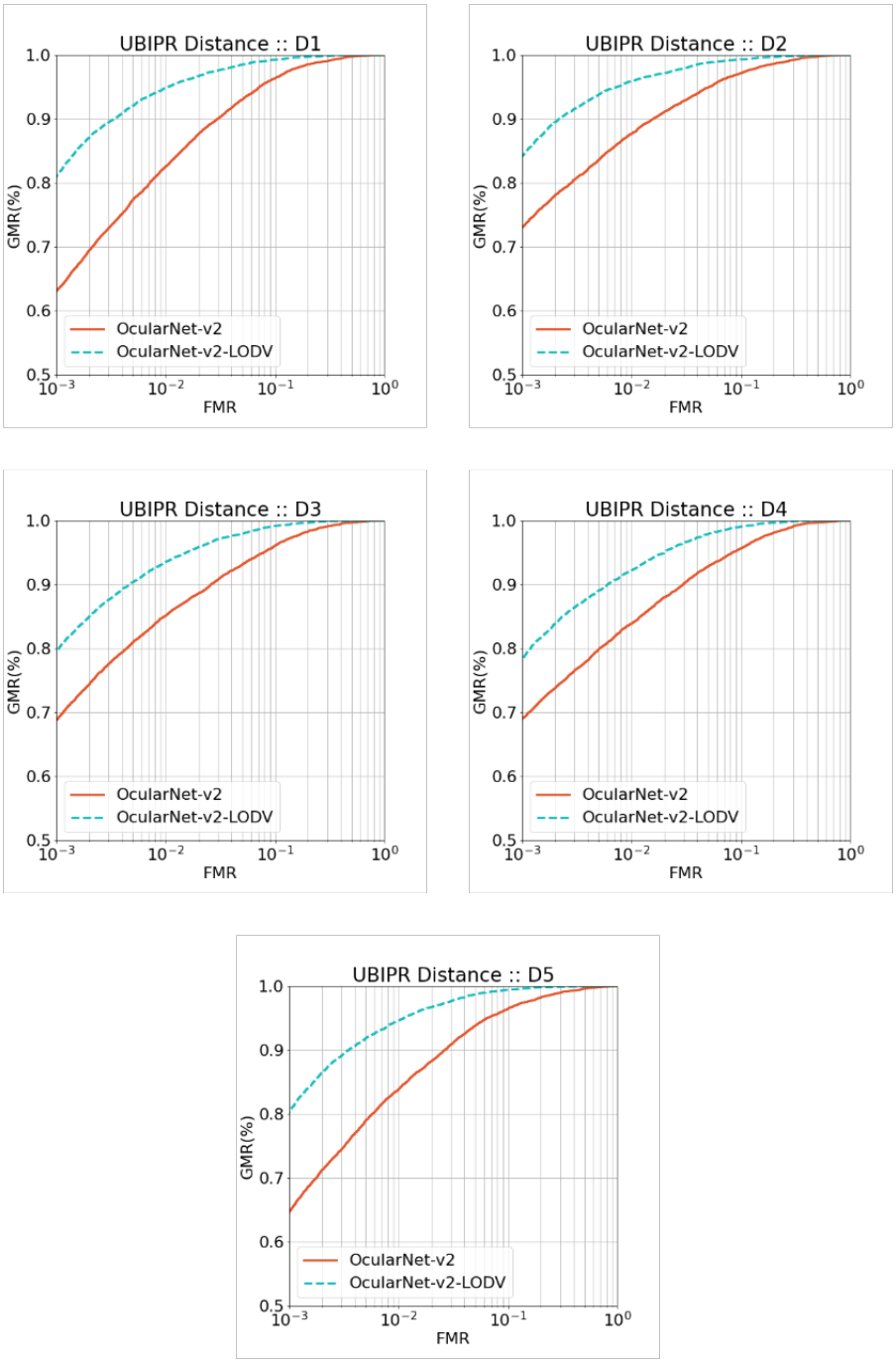


Figure 72: ROC curve of OcularNet-v2 model compared to OcularNet-v2 trained with LOD-V for UBIPR dataset with enrollments in a specific distance and verification samples in all remaining datasets.

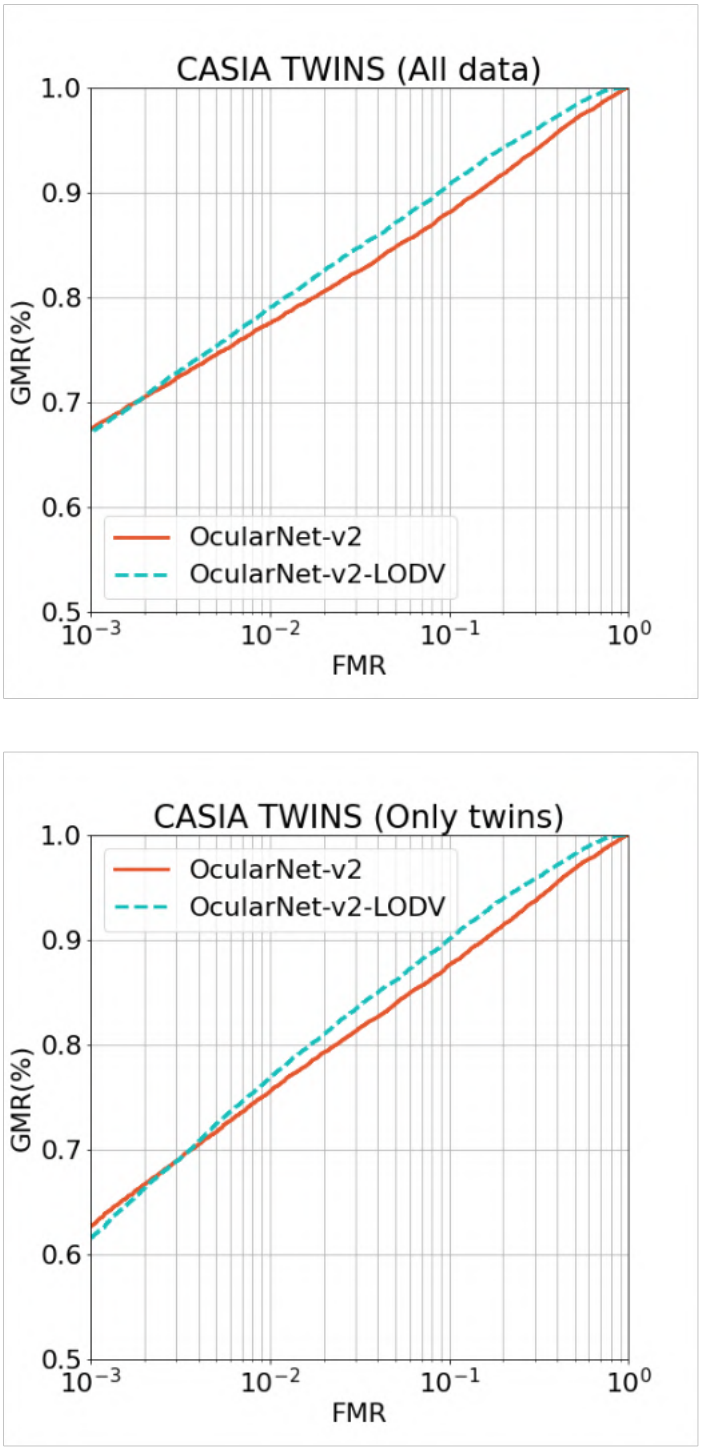


Figure 73: ROC curve of OcularNet-v2 model compared to OcularNet-v2 trained with LOD-V for CASIA TWINS dataset.

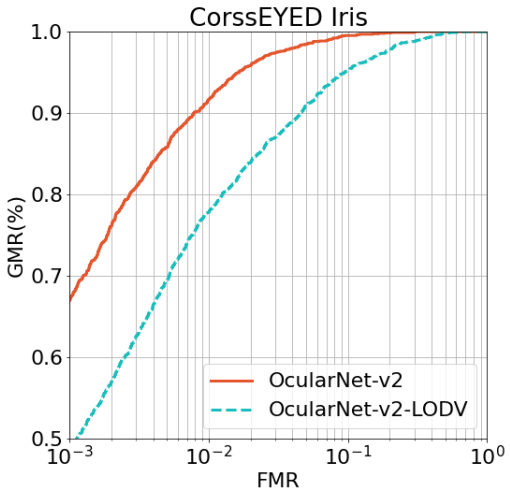


Figure 74: ROC curve of OcularNet-v2 model compared to OcularNet-v2 trained with LOD-V for CrossEYED iris only dataset.

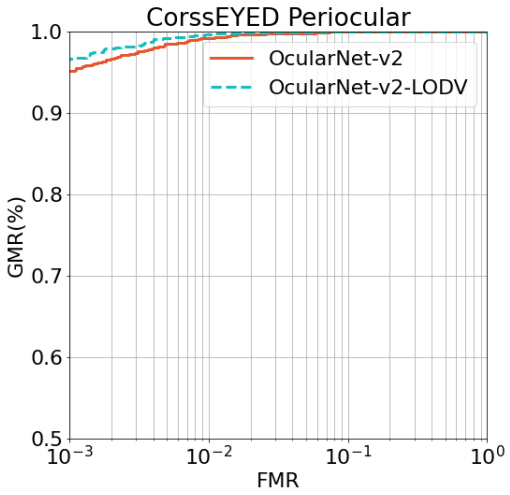


Figure 75: ROC curve of OcularNet-v2 model compared to OcularNet-v2 trained with LOD-V for CrossEYED periocular only dataset.

REFERENCE LIST

[1] Arunava, "An introduction to convolutional neural networks," Dec. 2018. [Online]. Available: https://towardsdatascience.com/convolutional-neural-network-17fb77e76c05

[2] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[3] A. Rattani, R. Derakhshani, S. K. Saripalle, and V. Gottemukkula, "ICIP 2016 competition on mobile ocular biometric recognition," in *Proceedings - International Conference on Image Processing, ICIP*, vol. 2016-August. IEEE Computer Society, aug 2016, pp. 320–324.

[4] Z. Zhao and A. S. Kumar, "Improving periocular recognition by explicit attention to critical regions in deep neural network," *IEEE Transactions on Information Forensics and Security*, vol. 13, pp. 2937–2952, 2018.

[5] A. F. Sequeira, L. Chen, J. Ferryman, P. Wild, F. Alonso-Fernandez, J. Bigun, K. B. Raja, R. Raghavendra, C. Busch, T. de Freitas Pereira *et al.*, "Cross-eyed 2017: Cross-spectral iris/periocular recognition competition," in *2017 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2017, pp. 725–732.

[6] A. Rattani, R. Derakhshani, and A. Ross, *Selfie Biometrics Advances and Challenges*, ser. Advances in Computer Vision and Pattern Recognition. Springer, 2019.

[7] A. Das, C. Galdi, H. Han, R. Ramachandra, J. Dugelay, and A. Dantcheva, "Recent advances in biometric technology for mobile devices," in *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, 2018, pp. 1–11.

[8] F. Alonso-Fernandez and J. Bigun, "A survey on periocular biometrics research," *Pattern Recognition Letters*, vol. 82, pp. 92–105, 2016.

[9] A. Rattani and R. Derakhshani, "Ocular biometrics in the visible spectrum: A survey," *Image and Vision Computing*, vol. 59, pp. 1 – 16, 2017.

[10] P. Kumari and K. Seeja, "Periocular biometrics: A survey," *Journal of King Saud University - Computer and Information Sciences*, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1319157818313302

[11] L. Nie, A. Kumar, and S. Zhan, "Periocular recognition using unsupervised convolutional rbm feature learning," in *2014 22nd International Conference on Pattern Recognition*. IEEE, 2014, pp. 399–404.

[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[13] Z. Zhao and A. Kumar, "Accurate periocular recognition under less constrained environment using semantics-assisted convolutional neural network," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 5, pp. 1017–1030, 2017.

[14] L. C. O. Tiong, Y. Lee, and A. B. J. Teoh, "Periocular recognition in the wild: Implementation of rgb-oclbcp dual-stream cnn," *Applied Sciences*, vol. 9, no. 13, p. 2709, 2019.

[15] A. Rattani and R. Derakhshani, "On fine-tuning convolutional neural networks for smartphone based ocular recognition," in *2017 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2017, pp. 762–767.

[16] K. B. Raja, R. Raghavendra, and C. Busch, "Collaborative representation of deep sparse filtered features for robust verification of smartphone periocular images," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sept 2016, pp. 330–334.

[17] A. Alahmadi, M. Hussain, H. Aboalsamh, and M. Zuair, "Convsrc: Smartphone based periocular recognition using deep convolutional neural network and sparsity augmented collaborative representation," *Journal of Intelligent Fuzzy Systems*, vol. 38, pp. 3041–3057, 2020.

[18] H. Proença and J. C. Neves, "Deep-prwis: Periocular recognition without the iris and sclera using deep learning frameworks," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 4, pp. 888–896, 2017.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[21] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[22] A. Canziani, A. Paszke, and E. Culurciello, "An analysis of deep neural network models for practical applications," *arXiv preprint arXiv:1605.07678*, 2016.

[23] A. Ignatov, R. Timofte, A. Kulik, S. Yang, K. Wang, F. Baum, M. Wu, L. Xu, and L. Van Gool, "Ai benchmark: All about deep learning on smartphones in 2019," in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, 2019, pp. 3617–3635.

[24] S. Bakshi, P. K. Sa, and B. Majhi, "A novel phase-intensive local pattern for periocular recognition under visible spectrum," *Biocybernetics and Biomedical Engineering*, vol. 35, no. 1, pp. 30–44, 2015.

[25] D. L. Woodard, S. J. Pundlik, J. R. Lyle, and P. E. Miller, "Periocular region appearance cues for biometric identification," in *2010 IEEE Computer Society Conference*

on Computer Vision and Pattern Recognition-Workshops. IEEE, 2010, pp. 162–169.

[26] Z. X. Cao and N. A. Schmid, "Matching heterogeneous periocular regions: Short and long standoff distances," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 4967–4971.

[27] A. Gangwar and A. Joshi, "Robust periocular biometrics based on local phase quantisation and gabor transform," in *2014 7th International Congress on Image and Signal Processing*. IEEE, 2014, pp. 714–720.

[28] K. B. Raja, R. Raghavendra, M. Stokkenes, and C. Busch, "Smartphone authentication system using periocular biometrics," in *2014 International Conference of the Biometrics Special Interest Group (BIOSIG)*. IEEE, 2014, pp. 1–8. [Online]. Available: https://ieeexplore.ieee.org/document/7029409

[29] L. R. Marval-Pérez, K. Ito, and T. Aoki, "Phase-Based Periocular Recognition with Texture Enhancement," *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, vol. 102, no. 10, pp. 1351–1363, Oct 2019.

[30] R. Raghavendra and C. Busch, "Learning deeply coupled autoencoders for smartphone based robust periocular verification," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 325–329.

[31] K. Ahuja, R. Islam, F. A. Barbhuiya, and K. Dey, "Convolutional neural networks for ocular smartphone-based biometrics," *Pattern Recognition Letters*, vol. 91, pp. 17–26, 2017.

[32] M. De Marsico, M. Nappi, D. Riccio, and H. Wechsler, "Mobile iris challenge evaluation (miche)-i, biometric iris dataset and protocols," *Pattern Recognition Letters*, vol. 57, pp. 17–23, 2015.

[33] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2911–2918.

[34] C. N. Padole and H. Proenca, "Periocular recognition: Analysis of performance degradation factors," in *Proceedings - 2012 5th IAPR International Conference on Biometrics, ICB 2012*, 2012, pp. 439–445.

[35] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun, "Bayesian face revisited: A joint formulation," in *European conference on computer vision*. Springer, 2012, pp. 566–579.

[36] National Institute of Standards and Technology, "Face Recognition Grand Challenge (FRGC)," 2005. [Online]. Available: https://www.nist.gov/programs-projects/face-recognition-grand-challenge-frgc

[37] National Institute of Standards and Technology, "Face and Ocular Challenge Series (FOCS)," 2010. [Online]. Available: https://www.nist.gov/programs-projects/face-and-ocular-challenge-series-focs

[38] Center for Biometrics and Security Research, "CASIA Iris Image Database, http://biometrics.idealtest.org/," 2010. [Online]. Available: http://www.cbsr.ia.ac.cn/china/IrisDatabasesCH.asp

[39] R. Garg, Y. Baweja, S. Ghosh, R. Singh, M. Vatsa, and N. Ratha, "Heterogeneity aware deep embedding for mobile periocular recognition," in *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. IEEE, 2018, pp. 1–7.

[40] G. Santos, E. Grancho, M. V. Bernardo, and P. T. Fiadeiro, "Fusing iris and periocular information for cross-sensor recognition," *Pattern Recognition Letters*, vol. 57, pp. 52–59, 2015.

[41] A. Sharma, S. Verma, M. Vatsa, and R. Singh, "On cross spectral periocular recognition," in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 5007–5011.

[42] N. Reddy, A. Rattani, and R. Derakhshani, "Robust subject-invariant feature learning for ocular biometrics in visible spectrum," in *2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. IEEE, 2019, pp. 1–6.

[43] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[44] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.

[45] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 525–542.

[46] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 1800–1807.

[47] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation," *arXiv preprint arXiv:1801.04381*, 2018.

[48] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[49] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, and W. Heinzelman, "Cloud-vision: Real-time face recognition using a mobile-cloudlet-cloud acceleration architecture," in *2012 IEEE Symposium on Computers and Communications (ISCC)*, July 2012, pp. 000 059–000 066.

[50] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 2261–2269.

[51] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," *arXiv preprint arXiv:1611.01578*, 2016.

[52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[53] H. ProenÃ§a and L. Alexandre, "UBIRIS: A noisy iris image database," in *13th International Conference on Image Analysis and Processing - ICIAP 2005*, vol. LNCS 3617. Cagliari, Italy: Springer, September 2005, pp. 970–977.

[54] H. Proenca, S. Filipe, R. Santos, J. Oliveira, and L. Alexandre, "The UBIRIS.v2: A database of visible wavelength images captured on-the-move and at-a-distance," *IEEE Trans. PAMI*, vol. 32, no. 8, pp. 1529–1535, August 2010.

[55] A. Sequeira, L. Chen, P. Wild, J. Ferryman, F. Alonso-Fernandez, K. B. Raja, R. Raghavendra, C. Busch, and J. Bigun, "Cross-eyed-cross-spectral iris/periocular recognition database and competition," in *Biometrics Special Interest Group (BIOSIG), 2016 International Conference of the*. IEEE, 2016, pp. 1–5.

[56] Elucideye, "drishti: Real time eye tracking for embedded and mobile devices," https://github.com/elucideye/drishti, 2000–2004.

[57] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *ECCV 2016*. Springer, 2016, pp. 499–515.

[58] S. P. Tankasala, P. Doynov, and R. Derakhshani, "Visible spectrum, bi-modal ocular biometrics," *Procedia Technology*, vol. 6, pp. 564–573, 2012.

[59] N. Reddy, A. Rattani, and R. Derakhshani, "OcularNet: Deep Patch-based Ocular Biometric Recognition," in *2018 IEEE International Symposium on Technologies for Homeland Security, HST 2018*. Institute of Electrical and Electronics Engineers Inc., Dec 2018.

[60] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Advances in neural information processing systems*, 2015, pp. 2017–2025. [Online]. Available: http://papers.nips.cc/paper/5854-spatial-transformer-networks.pdf

[61] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[62] H. Wang, S. Z. Li, Y. Wang, and J. Zhang, “Self quotient image for face recognition,” in *2004 International Conference on Image Processing, 2004. ICIP’04.*, vol. 2. IEEE, 2004, pp. 1397–1400.

[63] D. E. King, “Dlib-ml: A machine learning toolkit,” *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009. [Online]. Available: http://jmlr.org/papers/v10/king09a.html

[64] P. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, “The feret database and evaluation procedure for face-recognition algorithms,” *Image and Vision Computing*, vol. 16, no. 5, pp. 295–306, 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S026288569700070X

[65] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, “Vggface2: A dataset for recognising faces across pose and age,” *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, pp. 67–74, 2018.

[66] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, “Sphereface: Deep hypersphere embedding for face recognition,” in *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition. CVPR 2017*, 2017, pp. 212–220.

[67] ISO/IEC 19795-1:2006, “Information technology â biometric performance testing and reporting â part 1: Principles and framework,” International Organization for Standardization, Geneva, CH, Standard, 2016. [Online]. Available: https://www.iso.org/standard/41447.html

[68] S. Cass, “Nvidia makes it easy to embed ai: The jetson nano packs a lot of machine-learning power into diy projects - [hands on],” *IEEE Spectrum*, vol. 57, no. 7, pp. 14–16, 2020.

[69] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li, “A face antispoofing database with diverse attacks,” in *2012 5th IAPR International Conference on Biometrics (ICB)*, 2012, pp. 26–31.

[70] D. S. Ma, J. Correll, and B. Wittenbrink, “The chicago face database: A free stimulus set of faces and norming data,” *Behavior Research Methods*, vol. 47, no. 4, pp. 1122–1135, 2015.

[71] N. C. Ebner, M. Riediger, and U. Lindenberger, “FACES-A database of facial expressions in young, middle-aged, and older women and men: Development and validation,” *Behavior Research Methods*, vol. 42, no. 1, pp. 351–362, 2010.

[72] D. Lundqvist, A. Flykt, and Ã. Arne, “The Karolinska Directed Emotional Faces - KDEF,” *CD ROM from Department of Clinical Neuroscience*, 1998. [Online]. Available: https://kdef.se/home/aboutKDEF.html

[73] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, and A. Hadid, “Oulu-npu: A mobile face presentation attack database with real-world variations,” in *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, 2017, pp. 612–618.

[74] O. Langner, R. Dotsch, G. Bijlstra, D. H. Wigboldus, S. T. Hawk, and A. Van Knippenberg, "Presentation and validation of the radboud faces database," *Cognition And Emotion*, vol. 24, no. 8, pp. 1377–1388, 2010.

[75] A. Costa-Pazo, S. Bhattacharjee, E. Vazquez-Fernandez, and S. Marcel, "The replay-mobile face presentation-attack database," in *2016 International Conference of the Biometrics Special Interest Group (BIOSIG)*. IEEE, 2016, pp. 1–7.

[76] Y. Liu, A. Jourabloo, and X. Liu, "Learning deep models for face anti-spoofing: Binary or auxiliary supervision," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 389–398.

[77] H. M. Nguyen, N. Reddy, A. Rattani, and R. Derakhshani, "Visob 2.0 -second international competition on mobile ocular biometric recognition," in *Conference: International Conference on Pattern Recognition 2020*, 2020.

VITA

Sai **Narsi Reddy**, Donthi Reddy, is a Ph. D. candidate at UMKC currently working on Deep Learning applications in mobile ocular biometrics. His research also includes multi and single frame image enhancement and generalizable feature extraction methods to improve ocular biometrics. He received the Biometric and Forensics Best Paper Award at IEEE Homeland Security Technologies Symposium in 2017.