Effective and Efficient Attribute-aware Open-set Face Verification

Thesis submitted in partial fulfillment
of the requirements for the degree of

*Master of Science in **Computer Science and Engineering** by Research*

by

Arun Kumar Subramanian
2018900014
arun.subramanian@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, INDIA
June 2023

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "Effective and Efficient Attribute-aware Open-set Face Verification" by Arun Kumar Subramanian, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Adviser: Dr. Anoop Namboodiri

To My Family

Acknowledgments

I am deeply grateful to my advisor, Dr. Anoop Namboodiri, for his unwavering support, invaluable guidance, and profound insights that have been instrumental in shaping my academic journey. From being captivated by his lucid lectures in the classroom to the privilege of working under his mentorship during my research, every step of this journey has been enriched by his profound wisdom and engaging one-on-one discussions. I am fortunate to have had the opportunity to learn from and be inspired by Dr. Namboodiri, and I am sincerely thankful for his mentorship and contributions to my academic growth.

Further through his association, I'd like to thank the BASIL lab, the infrastructure, and the projects, for giving me an opportunity and exposure to the Biometrics domain in a wholesome manner. The lab's facilitation of interaction with research students cutting universities and stalwarts in the domain such as Dr Anil Jain, rocketed the interest and practice in the field. In this context, I would like to thank Saketh Srinivas for all the help in facilitating the efforts of students pursuing research/projects in the lab. I would like to thank each of my lab mates, for their help, and support, and for being great friends on and off work!

The encouragement of compassionate individuals beyond the campus, as well, has been instrumental in making this incredible journey a reality. I express my heartfelt gratitude to Asokan Pichai for his inspiration and initiation. I cherish the association with my good friend Raghava Modhugu, who was a great sounding board and support through this journey. I would also like to thank all the professors and students of CVIT lab and other research labs of IIIT, who've all inspired me greatly.

Lastly, my heartfelt thanks go to my family for their unwavering support and significant role in my journey.

Abstract

While face recognition and verification in controlled settings is already a solved problem for machines, the uniqueness of face as a biometric is that the mode of capture is highly diverse. A face could be captured nearby or at distance, at different poses, with different lighting, and by different devices. Face recognition/verification has several challenges to overcome to effectively perform under these varying conditions. Most current methods, try to find salient features of an individual by ignoring these variations. This can be looked at from the paradigm of signal and noise. The signal here refers to that information that is unique to an individual, but not varying as per the condition. Noise represents those aspects that are not related to the identity itself and are influenced by the capture mechanism, physical setting, etc. This is usually done through metric learning approaches in addition to the use of loss functions such as cross-entropy (e.g., Siamese networks, angular loss, and other margin losses such as ArcFace). There are certain aspects that lie between signal and noise such as facial attributes (such as eyeglasses). These may or may not be unique to the individual subject, but introduces artifacts into the face image. The question then arises, why can't these variations be detected using learning methods, and the knowledge thus attained about the variations be put to good use during the matching process? It is this curiosity that has resulted in aggregation strategies for matching, which were previously implemented for aspects such as pose, age, etc.

However, in the wild, humans demonstrate significant variability in facial attributes such as facial hair, eyeglasses, hairstyles, and make-up. This is common as one of the primary mechanisms of face image acquisition is covert capture in public (with ethics of consent in place), where people usually display significant variability in facial attributes. Hence it is very important to address this variability during the matching process. This work attempts to do the same. The curious question that arises however is if indeed matching performance varies if the attribute prior is known. Even if it does, how does one conceptualize a system that exploits the same? It is here that this thesis proposed two frameworks. One of the configuration-specific operating points and the other involves suppression of attribute information in face embedding prior to matching. The attribute suppression is attempted both directly at the final embedding, and suppression of intermediary layers of a Vision Transformer Deep Neural network. Both of these require the facial attribute of each image to be detected prior to passing the images into the proposed framework for matching.

The above naturally adds another task to the face verification pipeline. It is therefore extremely necessary to find efficient and effective ways of performing face attribute detection (and face template

generation), since efficiently performing parts, mitigates the pipeline expansion overhead and makes this a viable pipeline to consider for face verification. We observe that face attribute detection usually employs end-to-end networks, which results in a lot of parameters for inference. A feasible alternative is to constantly leverage the SOTA (state-of-the-art) face recognition networks and use the earlier feature layers to perform the face attribute classification task. Since the highly accurate SOTA is currently DNNs (Deep Neural Networks) for face, the same is dealt with in this thesis. More narrowly, we focus on open-set face verification, where DNNs aim to find unique representation even for subjects not used for training the DNN.

Contents

Chapter Page

1 Introduction . . . 1
1.1 Face Biometrics . . . 1
1.2 Signal and noise . . . 2
1.3 Overview of Face Verification . . . 2
1.3.1 Face Biometrics system and Evaluation . . . 3
1.3.1.1 Evaluation setting of Face biometrics . . . 3
1.3.1.2 Face Biometric System . . . 4
1.3.2 DNN Face template generation with SOTA models . . . 4
1.3.2.1 FaceNet . . . 4
1.3.2.2 ArcFace . . . 5
1.3.2.3 MagFace . . . 6
1.3.2.4 Face Transformer . . . 6
1.3.3 Face Attribute detection . . . 6
1.4 Motivation and Contributions . . . 7
1.4.1 Attribute-Aware Face Verification . . . 7
1.4.2 Efficient Face Attribute-detection and Face Verification to Implement Attribute-Aware Face Verification . . . 9

2 Literature Review . . . 10
2.1 Attribute-Aware Face Verification . . . 10
2.1.1 Potential for identifying and isolating facial-attribute from face templates . . . 10
2.1.1.1 Identifying facial attributes from pre-trained embedding . . . 10
2.1.1.2 Isolating face template neurons correlating with facial attributes . . . 11
2.1.1.3 Vision transformers for face attribute aware recognition . . . 11
2.1.2 Leveraging Facial Attribute Covariates to Improve Matching . . . 11
2.1.3 Can Face Masks be considered as a facial attribute? . . . 12
2.2 Effective and Efficient attribute detection approaches . . . 13
2.2.1 Effective attribute detection approaches . . . 13
2.2.2 Efficient attribute detection approaches . . . 14

3 Approaches to Attribute-aware Face Verification . . . 16
3.1 Introduction . . . 16
3.1.1 Score level suppression: Configuration Specific Operating Threshold . . . 16
3.1.2 Feature level suppression . . . 17
3.1.2.1 Attention Based Suppression . . . 18

3.1.2.2 Attribute Aware Face Embedding and Suppression 19
3.1.3 Need for the Two Approaches 20
3.2 Problem Setting 21
3.2.1 Verification Configurations Used in Our Methodology 21
3.2.2 Template Matching and CelebA Dataset 22
3.2.3 Choice of Facial Attributes 22
3.2.3.1 Attributes Chosen for Experiments on CelebA Dataset 22
3.2.3.2 Attributes Chosen for Experiments on IJB-C Dataset 23
3.2.4 Deviation from "Subject-specific" Template Modeling in IJB-C Dataset 23
3.3 Proposed Approach 23
3.3.1 CSOT: Configuration-specific operating threshold 23
3.3.1.1 Embedding Used and Choice of Facial Attributes for this Methodology 26
3.3.1.2 Scaling the in-between Distribution Mean 27
3.3.1.3 Application of CSOT for Masked Face Verification 28
3.3.2 Feature level suppression: Attention Based Suppression 28
3.3.2.0.1 Network trained: 29
3.3.2.0.2 Mask Dataset creation: 30
3.3.2.0.3 Suppression mechanism: 30
3.3.2.0.4 Sanity check with Eyeglasses attribute 30
3.3.3 Feature level susppression: AAFES: Attribute Aware Face Embedding and Suppression 31
3.3.3.1 Motivation 31
3.3.3.2 Correlating Attribute Label with Embedding Neurons and Generating Suppression Vector 33
3.3.3.3 Network Used and Training 33
3.3.3.4 Choice of Attributes in CelebA for AAFES Method 35
3.3.3.5 Sanity Check of Attribute Learning and Suppression 36
3.4 Experiments and results 37
3.4.1 Evaluation Methodology 37
3.4.2 Results for Operating Point Adjustment by Mean Scaling 37
3.4.2.1 Results on CelebA Dataset 37
3.4.2.2 Results on IJB-C Dataset 39
3.4.2.3 Results on LFW Mask Dataset 40
3.4.3 Results for Feature level Supression 40
3.4.3.1 Results for Attention Based Suppression Method 40
3.4.3.1.1 Quantitative results 40
3.4.3.1.2 Qualitative Results 40
3.4.3.2 Results for AAFES Method 40
3.4.3.2.1 ROC Curve for the Dataset with the Attribute in the Wild . 40
3.4.3.2.2 Qualitative Results 40
3.5 Summary 42

4 Efficient Face Attribute-detection and Face Verification to Implement Attribute-Aware Face Verification 44
4.1 Introduction 44
4.2 Overview of existing compact model approaches 45

4.3 Proposed Approach 45
4.3.1 Efficient attribute-detection DNNs 45
4.3.1.1 Efficient DNN model architectures for end-to-end face attribute detection 45
4.3.1.2 Efficient attribute detection from trained Face Recognition DNNs . . 46
4.3.2 Efficient Face Embedding DNN 47
4.4 Experiments and Results 48
4.4.1 Results for efficient attribute-detection DNNs 48
4.4.1.1 End-to-end attribute training 48
4.4.1.2 Efficient attribute detection from trained Face Recognition DNNs . . 49
4.4.2 Results for Efficient Face Embedding DNN 49
4.5 Summary 50

5 Conclusions and Future Work 53

Bibliography 55

List of Figures

Figure Page

1.1 Intra-subject variations . . . 2
1.2 Intra-subject variations due to attributes. . . . 3
1.3 Face recognition processing flow. [35] . . . 4
1.4 InceptionV3 network. . . . 5
1.5 ArcFace Metric Loss. [13] . . . 5
1.6 MagFace leveraging both quality and class information of image. [43] . . . 6
1.7 Diagram of the VIT architecture and mechanism . . . 7
1.8 Atrribute suppression at three levels: Image level suppression involves masking the facial attribute at the image level before extracting the features. Feature level suppression involves masking the neurons in the DNN embedding that is most sensitive to a given facial attribute or masking spatial region corresponding to attribute of interest. Score level processing involves determining the unique threshold at the score level for a given facial attribute configuration of the matching pair. . . . 8

2.1 Effect on genuine impostor score, with varying yaw (left) and roll(right) . . . 12
2.2 Sample masks images for two subjects . . . 13

3.1 A plot for the 'Smiling' attribute, showing that matching operates differently depending on whether probe and gallery have the attribute in question or not. *att* in the plot above refers to probe and gallery having attributes. *att-noatt* is probe having the attribute and gallery without the attribute. *noatt* refers to probe and gallery not having the attribute. . 17
3.2 For both genuine-impostor pair, for eyeglass attribute, the first block is att-att, where probe and gallery both possess the attribute; The att-noatt where probe has the attribute but the gallery doesn't. And finally noatt-noatt is when both the probe and gallery do not possess the attribute. . . . 17
3.3 A typical face image has the following Attention map generated by the Vision transformer model trained on a large face training . . . 19
3.4 The three factors considered for the choice of attributes in experiments in Chapter 3 . . 22
3.5 The IJB-C Occlusion grid labels are leveraged to determine the eyeglass and forehead occlusion labels . . . 23
3.6 On the left, the regular DNN; Proposed method on the right, where we determine the config using attribute detector network, and use mapped scaling-factor(synonymous to unique threshold).RMSE block computes RMSE between two embeddings and multiplies with the scaling factor. . . . 24

3.7 On top is a schematic plot where the plot in Orange (both genuine and impostor), Green, and Blue each refer to att-att, att-noatt, and noatt-noatt distribution 26
3.8 The plot here shows the genuine impostor score distribution of FaceNet network on CelebA dataset's Eyeglass attribute.The highlighted marking in yellow shows the regions were each configuration distribution ends up different from the other configuration distribution 27
3.9 The first attention map (center image) constitutes the raw attention map of a VIT network when inferenced over a masked face image. The second attention map (rightmost image) constitutes the case where the face-mask region is 0 weighted in the attention map. 29
3.10 First two images from left to right represent mask overlay on LFW images. The last two represent eyeglass overlay used for sanity checking attribute suppression 30
3.11 On the left, the regular DNN; Proposed method on the right, where config is determined using attribute detector network, and use mapped scaling factor. 31
3.12 p is a subset of neurons most positively or negatively correlated with a given attribute . 34
3.13 The left half is frozen, while the right half of InceptionResnetV1 is trained. 35
3.14 Positive class rate of the smiling attribute is balanced and labeling robust as well. . . . 35
3.15 Pixel level occlusion patch to show the largest drop in accuracy. The same was performed for Smiling and bangs. 36
3.16 Top to bottom: Eyeglass, Heavymakeup, Goatee, Mustache. ROC plots on left are for individual configuration; And on the right on full data, with scaling in brown and; without-scaling in black. The labels on all the graphs are of the form Accuracy as a number; Intra/inter pair count and protocol. 38
3.17 Top: ROC plots for Facenet on IJB-C dataset. From the accuracy numbers, one can see the average of att/noatt/att-noatt protocol is better than combined (in black) accuracy. Bottom: The plot shows an improvement in accuracy i.e. 0.5 % increase when mean scaling is done. 39
3.18 This is the out of each head for the face image with mask in Figure 3.9 41
3.19 The top two rows are genuine pairs and the last row is the impostor pair matched correctly after the suppression of maximal activation. 43
3.20 The yellow line demonstrates the improvement in matching after suppression. 43

4.1 MobileVIT architecture from the original paper [42]. 46
4.2 Figure showing attribute detector in orange, branched from the second layer. 47
4.3 Schematic Diagram of Resnet34.Resnet50 however is in actual use as our model 48
4.4 A typical squeeze excitation layer 48
4.5 On the right is a fully connected final layer; On the left is our removal of the fully connected layer and introduction of average pooling. The blue font shows the extra parameters. The bold black numbers show the difference in size and params 52

List of Tables

Table Page

3.1 As a part of sanity check, the table demonstrates how LFW images overlayed with dark patches mimicking eyeglasses verification accuracy remains pretty much unchanged. . 31
3.2 Accuracy before and after suppression in percentage for available labels of high confidence from MAAD on VGGFace2 36
3.3 GMean is the Genuine mean and Gstd is Genuine Standard deviation. Likewise, Imean is Impostor mean 41
3.4 Verification accuracy 41
3.5 Accuracy post *att*, *noatt* and *att-nott* binning individually. *Without CSOT* refers to current SOTA; *CSOT* is our method that uses individual bin thresholds and aggregates the result as explained 42
3.6 Accuracy@EER, TAR@FAR 1e-4 and Oper Threshold (operating threshold) for Masked Faces on LFW dataset. NoAtt refers to probe-and-gallery having no attribute (face mask in this case). Similarly Att refers to probe-and-gallery having the attribute 42
3.7 The table demonstrates how LFW images overlayed with attribute i.e face-masks Verification accuracy drops for Eyeglasses after suppression, but increases for face-masks after suppression 42

4.1 Table showing achieving SOTA with half the number of params. The parameter Count is in Millions. 49
4.2 Table showing accuracy of face attributes varies, with taken representation at different layers.The configuration Layer2 implies activations are collected(by-passed) from this layer and passed to embedding layer. Whereas config OnlyFinalEmbedding refers to the final embedding of the network. Reduction_to means the reduction from Reduction_from number of channels down to a specified number. EmbeddingLength is the final embedding length (Embedding length is not applicable for Layer3 and Layer4 configuration because feature-map reduces to a very small size. Hence directly using the flattened layer) 50
4.3 Demonstrates that Magface embedding from early layers captures near SOTA accuracy over most attributes 51
4.4 Predictive power of three facial attributes at different layers between MagFace and ArcFace 51
4.5 Verification accuracy of Reduced Model vis-a-vis original 51

Chapter 1

Introduction

1.1 Face Biometrics

Face biometrics involves encapsulating the unique characteristics of an individual's face. Humans do this on a daily basis to recognize others and we are quite adept at it. During this process, we naturally eliminate details that are not inherent to the face, such as facial hair, spectacles, and expressions. Machines on the other hand require mimicking this ability, by creating templates or extracting key features from a facial image. In this aspect face biometrics intersects the field of computer vision and machine learning.

As quoted in [35], *Face recognition has several advantages over other biometric modalities such as fingerprint and iris: besides being natural and nonintrusive, the most important advantage of face is that it can be captured at a distance and in a covert manner. Of the 6 attributes considered by [28], facial features scored the highest compatibility in a Machine Readable Travel Documents (MRTD) (http www.icao.intmrtdoverviewoverview.cfm) system based on a number of evaluation factors, such as enrollment, renewal, machine requirements, and public perception.* Moreover, the internet is a huge repository of facial images, and many a datasets are crawled from the web after requisite permissions are obtained. These benign aspects of face modality lends itself to multiple applications and scenarios of verification/identification. From the learning perspective, face offers ample opportunities for data collection and modeling. This of course is done, while keeping the ethics of privacy after acquiring requisite permissions.

Face recognition, by machine learning leverages the advantages discussed above, and aims to create a template/embedding/features from the facial images gathered in various forms. Given the diversity of acquisition mentioned above, there is a natural but significant variation in illumination, facial pose, expression, age span, hair, facial wear, and motion. The goal of an ideal template for recognition would be to capture the identity information while being unaffected by these variations. Some of the earlier milestones in face template generation involves PCA based EigenFaces [65], Fischerface method [[18], [4]] , Gabor jets [[69],[36]]. Adaboost based face detection algorithm by Viola and Jones [66] became the defacto face detector in most applications. The emergence of deep-neural-networks imparted a

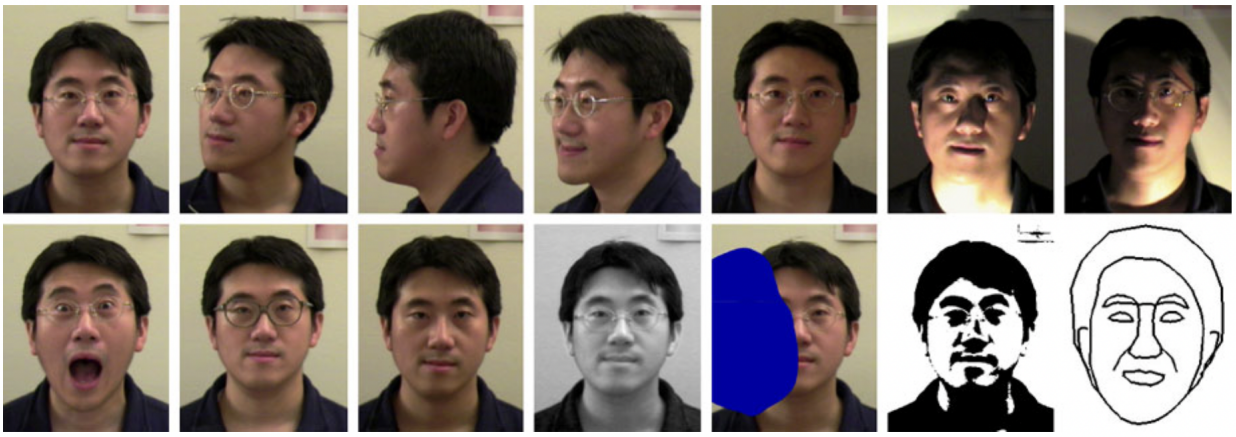


Figure 1.1: Intra-subject variations

[31]

significant boost to the field of face recognition. We will take a closer look at some of these recognition and detection models in the next few sections.

1.2 Signal and noise

To effectively learn a unique face embedding of a subject, it is necessary to separate core facial features of the subject from other variations such as pose, illumination, occlusion, accessories, brightness, and color (see Figure 1.1). This unique representation can be called a *Signal* that needs to be separated from the *noise*, which in this case can be thought of as the aforementioned variations.

However, in the context of creating a unique template of a subject's face, there are certain kinds of artifacts that are hard to classify as *noise* or *signal*. Facial accessories for instance fall into this category (see Figure 1.2). They can't be termed as *signal* because they aren't persistent enough to be integral to the unique subject identity, and neither can they be termed *noise* because these attributes are quite prevalent in the faces-in-the-wild scenarios. And since the DNN embedding could potentially capture these attributes in the representation, we could potentially leverage this information for better matching. It is towards this end, that the current thesis directs its efforts.

1.3 Overview of Face Verification

The following three aspects of face verification are relevant to this thesis. The first is the settings in which a face biometric system is deployed and its evaluation setting. The second is deep-neural-network-based Face embedding generation and matching, as employed by state-of-the-art (SOTA) methods. As we want to use facial attribute information to improve face recognition systems, its efficient and effective detection from a given image is also important to our overall system.

Figure 1.2: Intra-subject variations due to attributes.

1.3.1 Face Biometrics system and Evaluation

Due to the nature of face capture, the setting under which face images are captured for recognition significantly differs from other biometric traits. We will look into the overall recognition pipeline and its evaluation.

1.3.1.1 Evaluation setting of Face biometrics

While the taxonomy of Face biometric evaluation delineates all aspects of Face biometrics, the following definitions are critical to our work.

We broadly have *Face Verification* and *Face Identifcation/Recognition*. *Face Verification* is a one-to-one matching between a query face image against an enrollment face image whose identity is being claimed. While *Face Recognition* is a one-to-many matching between query face image against all the face images in the enrollment database. Refer Book [35] for aforementioned definitions

Within *Face Identification* we further have *Open-Set Identification* and *Closed-set Identification*

- Closed-set Identification: Here the query image is known to be present in the enrollment database, and you determine if the matching algorithm is able to retrieve the correct identity within top n matches (n is determined by the performance needs of the user).

- Open-set Identification: Here the query image might *not* be present in the enrollment database, and you determine if the matching algorithm is able to retrieve the correct identity within top n matches. In case the identity is not present in the database, the matching algorithm should say so.

Within *Face Verification* since the face image presented already claims an identity, we do not need the classification approach we took above. However, from the perspective of training the models to learn

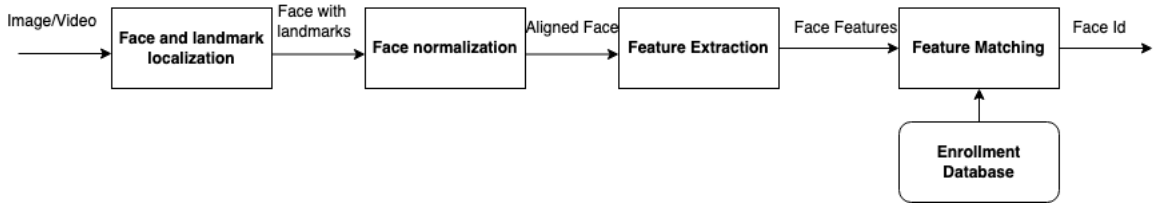


Figure 1.3: Face recognition processing flow. [35]

the face templates, *Open-set verification* involves not using *any* of the face images of the class/identity being presented for verification, during the training of the model. *Closed-set verification* on the other hand allows usage of the face images of the class/identity presented during verification when training a model that generates face templates. *It is this Open-set verification setting that is of relevance to our thesis*

1.3.1.2 Face Biometric System

A typical Face Biometric system has the following processing flow Figure 1.3.

Face detection step involves the first step of localizing the *landmarks* of the face: eyes, ears, nose, and mouth; and detecting the face region around it.

Face normalization involves taking care of those variations in the face due to geometry (due to various poses) and photometry (illumination and color etc). Normalization to reduce the effect of these variations significantly improves the effectiveness of the next steps.

Feature extraction involves drawing salient features, from the image passed on by the prior steps, to generate a discriminating template, unique to each identity.

Face matching involves matching the erstwhile generated salient feature vector against a similarly extracted template in the enrollment database.

1.3.2 DNN Face template generation with SOTA models

We now discuss three state-of-the-art (SOTA) deep-neural-networks that are used in the experiments of this work. All three SOTA networks demonstrate high performance on the *Labeled faces in the wild* (LFW [34]) dataset. However, with the emergence of more recent challenging datasets such as IJB-C [40] there is tighter benchmarking w.r.t SOTA models. These datasets will be elaborated upon in Chapter3

1.3.2.1 FaceNet

FaceNet [56] employed SOTA CNN architectures (of its time) and applied Triplet loss over a large amount of proprietary data to create a trained network. One of these networks is based on InceptionV3 as its backbone (see Figure 1.4).

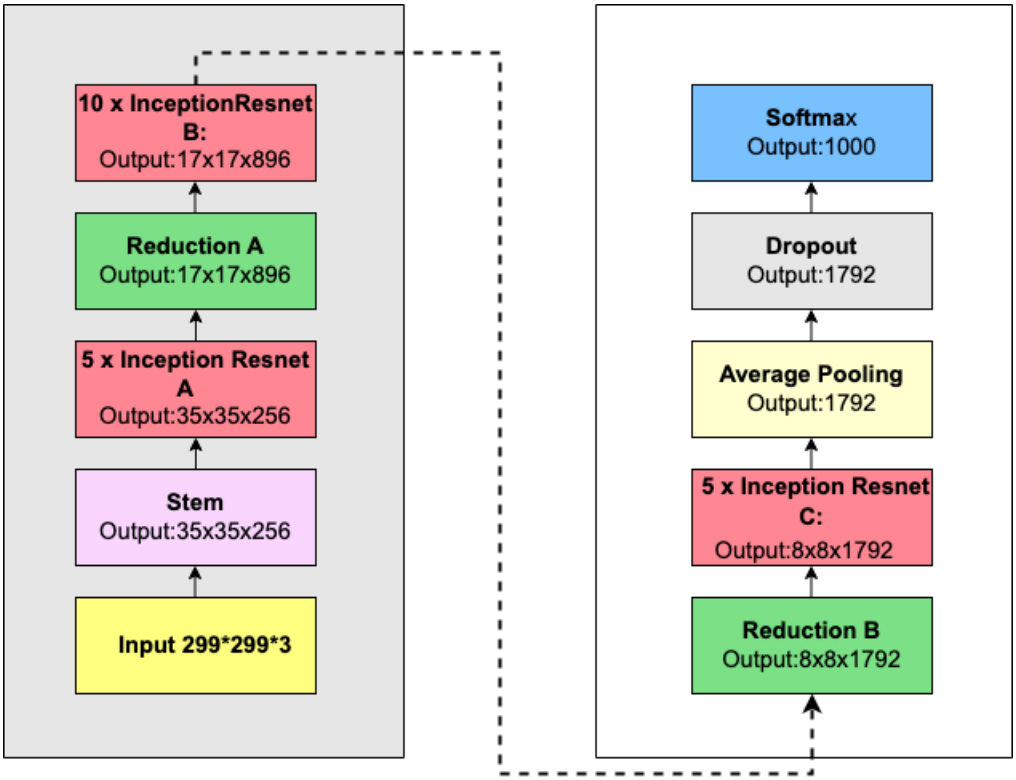


Figure 1.4: InceptionV3 network.

[59]

1.3.2.2 ArcFace

ArcFace [13] is a metric-learning approach to face template embedding. It improves upon the concept of center-loss, by leveraging the normalized weight vector as the center and adding an additive margin between the weight vector and the class face embedding as depicted in Figure 1.5. The network used here is ResNet50 and ResNet100, and the dataset used here was the largest publicly available dataset of the time called MS1M [25].

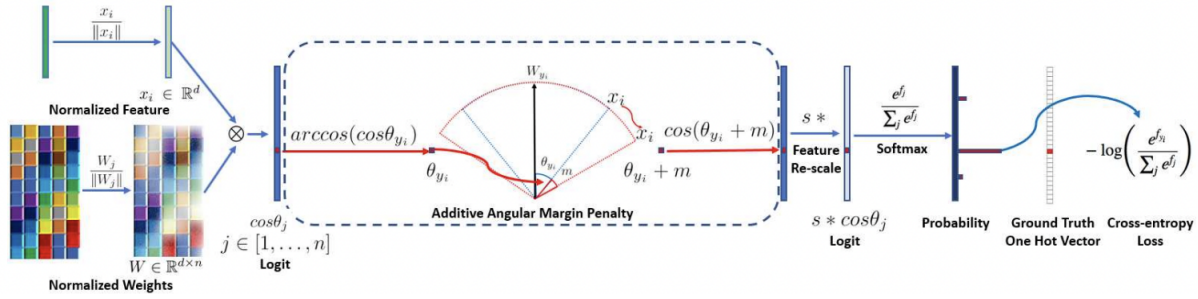


Figure 1.5: ArcFace Metric Loss. [13]

1.3.2.3 MagFace

MagFace [43] improves upon ArcFace by simultaneously enforcing direction and magnitude when optimizing, the learned face representation is more robust to the variability of faces in the wild. Thus while ArcFace normalizes the weight vector, MagFace leverages the magnitude of the weight vector by factoring it into the additive angular margin, thus getting an extremely interesting distribution of identity and quality as demonstrated in the image in their paper Figure 1.6

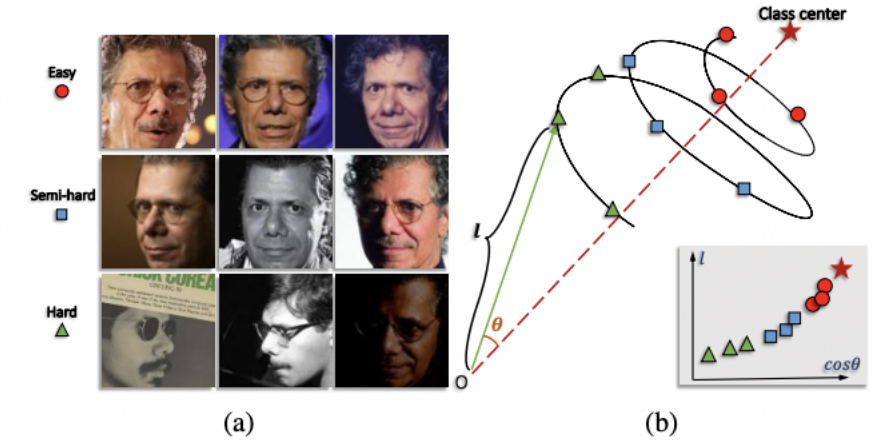


Figure 1.6: MagFace leveraging both quality and class information of image. [43]

1.3.2.4 Face Transformer

The paper [72] extrapolates the Vision Transfomer paper [17] to Faces with the additional step that they modify the tokens generation method of ViT, to generate tokens with sliding patches, i.e., to make the image patch overlaps, for a better description of the inter-patch information, as shown in Figure 1.7. Each such patch created runs through a linear projection layer and is then fed to the multi-head transformer encoder block, and finally, the MLP head takes in the output of the transformer encoder to identify the identity corresponding to the individual. The model achieved over 99.71% accuracy on the LFW dataset.

1.3.3 Face Attribute detection

Face attribute detection involves detecting attributes such as facial hair, expression, eyeglasses, age, and gender from a face image. CelebA [37] dataset has 40 facial attributes, whole presence/absence is marked for each image in the dataset. Deep-neural-network-based models have shown on an average of 92% accuracy over these 40 attributes of CelebA as demonstrated in [55]. MobileNetV2 with 2.2 million parameters has been trained to reproduce the same.

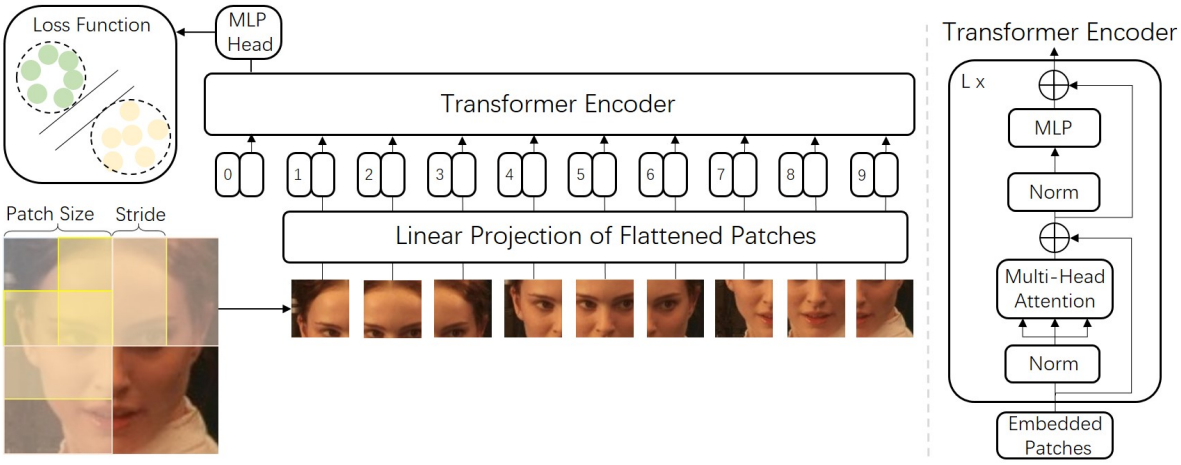


Figure 1.7: Diagram of the VIT architecture and mechanism

[72]

1.4 Motivation and Contributions

In the context of above discussions, we make two primary contributions in this thesis:

1.4.1 Attribute-Aware Face Verification

The SOTAs in facial recognition are Deep Neural Networks. As explained in the previous section, their fundamental goal is to create a unique representation for each individual. As discussed in the section 1.2 (*Siganal And Noise*), the approach of using attribute agnostic DNN embeddings does not consider the presence or absence of facial attributes (e.g., facial hair, eyeglasses) during matching. We are curious to see if the matching performance would improve, given that the probe image and the gallery image, both adorned the same facial attribute. This argument could be extended, even to the absence of the facial attribute in the probe-template pair, and finally to the presence in the probe template and absence in the gallery template. Alternatively, can we suppress the attribute information in the probe and the templates to improve matching performance? If so what approaches can be used for this suppression? We have considered the following (see image 1.8):

- **Image Level**: This involves first detecting the facial attribute ROI (Region of Interest) of each image in the matching pair of images and then masking the ROI, before sending the images to the DNN to generate face embedding, and further generate matching scores.

- **Score Level**: Here, based on the presence or absence of a facial attribute, we first determine the configuration and then apply a configuration-specific threshold to the matching score. The configuration-specific threshold referred to here is determined from a large number of matching pairs per bin, prior to the matching stage during inference. This score-level fine-tuning of the

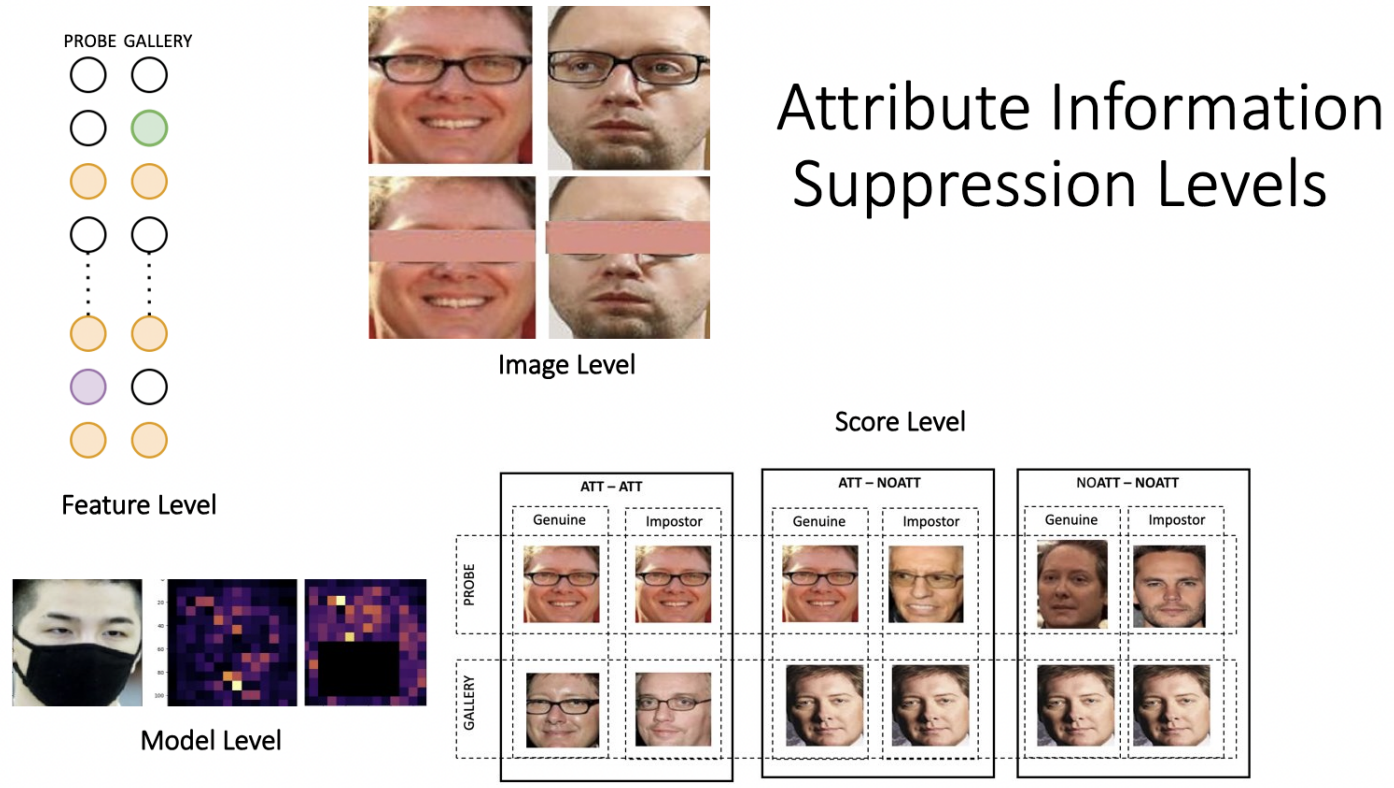


Figure 1.8: Atrribute suppression at three levels: Image level suppression involves masking the facial attribute at the image level before extracting the features. Feature level suppression involves masking the neurons in the DNN embedding that is most sensitive to a given facial attribute or masking spatial region corresponding to attribute of interest. Score level processing involves determining the unique threshold at the score level for a given facial attribute configuration of the matching pair.

algorithm is what translated to our first method: i.e., Configuration Specific Operating Threshold (CSOT)

- **Feature Level suppression**:

  Model Level Suppression: Here, for a given Vision Transformer model trained on face recognition, we zero-out the attention map corresponding to the spatial region of interest, and forward pass the image over the entire model; The resulting embedding we hypothesis would be attribute agnostic, and have indeed proven so through out experiments.

  Final Embedding Suppression: Here, we determine the neurons in the standard face embedding, that are most sensitive to a given facial attribute. We then zero-out (disregard) the output of these neurons during the matching process. This feature-level embedding suppression is what is translated to our second method Attribute-aware feature embedding and suppression (AAFES)

With the above fundamental motivation, two methods of performing such an attribute-aware match have been devised. On one hand, a unique operating point has been derived for each case where the

probe and template image exhibit different combinations of the presence or absence of an attribute. By using this unique operating threshold, when determining where the matched pair of images is a mate or non-mate, it is shown that the matching performance can be improved. On the other hand, a method of suppressing the attribute information in a given face template (DNN embedding) either by suppressing individual neurons of the embedding itself (final embedding level) or suppressing attention-map of the Vision transformer model (model level) has also been demonstrated. This act of suppressing the neurons in the template demonstrating the presence of the attribute helps to match in an attribute-agnostic manner.

1.4.2 Efficient Face Attribute-detection and Face Verification to Implement Attribute-Aware Face Verification

The approach described in 1.4.1 add additional processing (of attribute detection) in the pipeline. Efforts to minimize the processing cost of such a component is critical to reduce the matching time. It is, for this reason, an *efficient* method of processing facial attribute detection is desirable. Efficient here relates to enabling faster inferencing of facial-attribute detector. This efficiency goal can be achieved by ensuring the number of parameters of such a DNN is very small.

With the stated motivation, two approaches to seeking such efficient/compact networks have been proposed. One is to look at compact DNNs and demonstrate their ability to do facial attribute detection on par with SOTA methods. The second approach is to explore if current *face recognition DNNs* has layers that capture the facial attributes, and exploit those layers to build a head network on top, to classify the facial attributes. We present highly accurate solutions for attribute detection using both of these approaches.

Chapter 2

Literature Review

2.1 Attribute-Aware Face Verification

The papers reviewed in this section attempt at ascertaining that face templates indeed capture the facial attribute information and while the other subsection explores current approaches at using facial attribute covariates to improve matching accuracy.

2.1.1 Potential for identifying and isolating facial-attribute from face templates

2.1.1.1 Identifying facial attributes from pre-trained embedding

The goal of the literature survey in this section is to demonstrate the availability of finding and isolating face attribute information in face recognition templates. This is important for our work since it demonstrates that our effort of binning DNN embedding of templates having the same attribute or the other approach of suppressing attribute information in DNN embedding, both are indeed relevant. [62] shows that attribute-rich datasets such as CelebA (open-set verification), the resulting embedding are capable of capturing soft-biometrics such as age, demographics, ethnicity, and facial hair. Also, [46] shows that attributes clustered images are found at different layers of the face space.

Further, the work [15] first clearly demonstrates that the SOTA networks such as ArcFace reveal the gender and skin tone information. (They however further go on to try to mitigate this information in the embedding for other security purposes).

For instance, [48] attempts to create an embedding, that is capable of detection, landmark localization, pose estimation, and gender recognition. Embedding generated from attempts of this nature could be passed through the pipeline of our method, i.e. of using the attribute prior to finding a new operating threshold or suppressing the attribute prior(to be expanded in the next chapters), to get better verification accuracy. Face recognition tasks also have been shown to improve by leveraging attribute information [22]. However, there are approaches that aim for a joint representation of both identity and attributes as in [32] because as noted here Face Attribute Feature (FAF) are more robust though less discriminative, whereas Face Recognition Features (FRF) is less robust but more discriminative. Other approaches such

as [38] further analyze co-variation of attributes with generated embedding, and combined training used in [49] further denotes relevance of attribute aware embedding even if not captured in single embedding. In the work, [67] joint training in the multi-task setting of attributes and identity is performed, but for attributes that are invariant to the visual appearance of a person in a different situation (which is opposite to our goal). In the work [60] it is also attempted to create a joint representation of attribute and embedding (using a Kronecker product in the fusion layer).

2.1.1.2 Isolating face template neurons correlating with facial attributes

The literature that demonstrates the mechanism of isolating neural activation given a particular input is surveyed here. Since it is on this intuition, an attribute suppression network is built. [16] aims to find and isolate neurons that maximally activate for an attribute, however, the goal there is oriented more towards interpret-ability, whereas our work aims to find suppress able neurons in the embedding layer for a better match. Another work in a similar spirit is [19], but it differs in that it bins the average of the neurons of the embedding *after* averaging all the templates of a given identity (which too is a deviation, because this work focusses attributes).

2.1.1.3 Vision transformers for face attribute aware recognition

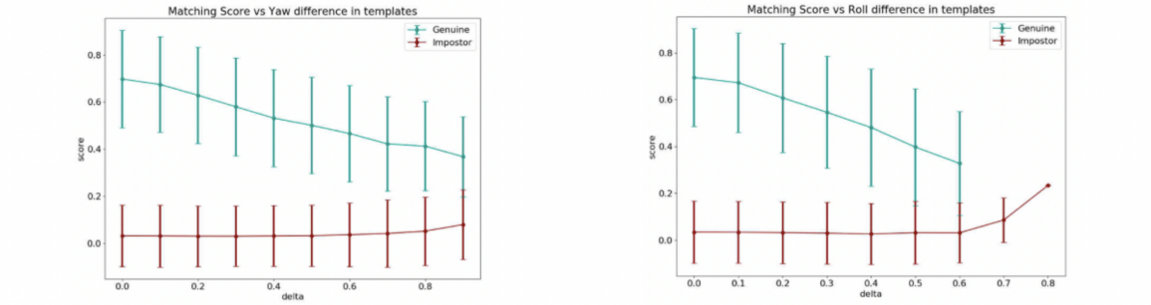
While face recognition using transformers was an extremely significant work [73], leveraging the attention maps we think could be crucial. One such work was done by [58] in leveraging vision transformer architecture to mimic the traditional part-based method of face recognition. Other works such as [2] leverage the intermediary attention maps for FPAD applications, while [72] explores self-attention for a visual linguistic description of face. But to the best of our knowledge, there has not been an exploration of face-attribute aware learning using ViTs.

2.1.2 Leveraging Facial Attribute Covariates to Improve Matching

The literature review in this section focuses on finding support for the fundamental idea of this thesis, that when a pair of face images are matched, the attribute covariate's presence or absence in the mated template, enables an effective match. For instance, [54] have shown that templates constructed for similar poses yielded better verification accuracy. Refer Figure 2.1 [7] relied on naive averaging of various facial and image attributes to form a unique subject-specific template.

[8] By creating one vs all classifier between a probe template and the rest of the gallery templates, the probe template "adapts" to mated-gallery. A similar classifier is also executed on the gallery.

[70] Assesses the importance of various attributes/features of a face image in a video frame and accordingly creates a representation.



(a) Matching Score vs Yaw Difference

(b) Matching Score vs Roll Difference

Fig. 2: Behavior of matching performance when using sub-templates constructed based on specific poses. The vertical limits of the error bar indicate the variance of scores observed for a specific pose difference in sub-templates.

Figure 2.1: Effect on genuine impostor score, with varying yaw (left) and roll(right) [54]

2.1.3 Can Face Masks be considered as a facial attribute?

This section performs a literature review of the impact of Face Verification performance due to Face Mask usage (primary as a result of COVID), and further justifies, why *face masks* are considered only with the Vision Transformers approach of model level suppression under the AAFES paradigm. Reference of persons with masks 2.2 [10] established negative impact especially of impostor distribution after the introduction of masks on human faces on SOTA network embeddings. This work is done on a dataset collected with real masks on the face of individuals. [9] also questions whether synthetic masks are effective in evaluating the performance of masked face verification. As studied in [45] the following conclusions are made w.r.t impact of mask-wearing on face verification:

- *False rejection performance: All algorithms submitted after the pandemic continue to give increased false non-match rates (FNMR) when the probes are masked. While a few pre-pandemic algorithms still remain within the most accurate on masked photos, some developers have submitted algorithms after the pandemic showing significantly improved accuracy and are now among the most accurate in our test. Using border crossing images, without masks, the most accurate algorithms will fail to authenticate about 0.3% of persons while falsely accepting no more than 1 in 100000 impostors (i.e. FNMR= 0.003 at FMR= 0.00001). With the highest coverage mask we tested and the most accurate algorithms, this failure rate rises to about 5% (FNMR = 0.05) among all algorithms. This is noteworthy given that around 70% of the face area is occluded by the mask. However, many algorithms submitted since mid-March 2020 remain much less tolerant: some algorithms that are quite competitive with unmasked faces (FNMR ¡ 0.01) still fail to authenticate between 10% to 40% of masked images (FNMR - 0.4).* - Quoted from the source.

- *Evolution of algorithms on face masks: We observe that a number of algorithms submitted since mid-March 2020 show notable reductions in error rates with face masks over their pre-pandemic predecessors. When comparing error rates for unmasked versus masked faces, the median FNMR*

across algorithms submitted since mid-March 2020 has been reduced by around 25% from the median pre-pandemic results. - Quoted from the source

- *False acceptance performance: As most systems are configured with a fixed threshold, it is necessary to report both false negative and false positive rates for each group at that threshold. When comparing a masked probe to an unmasked enrollment photo, in most cases, false match rates (FMR) are reduced by masks. The effect is generally modest with reductions in FMR usually being smaller than a factor of two. This property is valuable in that masked probes do not impart adverse false match security consequences for verification. However, when both the enrollment and verification images are masked, most algorithms give elevated false match rates, with FMR ranging from 10 to 100 times higher than when only the probe is masked or both images are unmasked, at the same threshold.* - Quoted from the source

Figure 2.2: Sample masks images for two subjects

[10]

One key observation made in the study above is *For the case where both the enrollment and verification images are masked, interestingly, many algorithms show a reduction in false non-match rates compared to when only the verification image is masked, at a fixed threshold. While the reduction in FNMR is favorable, we observe much larger false match rates when both images are masked. These findings are discussed in subsequent sections of this executive summary*. It is exactly this kind of observation, that bolsters the validity of my research, because, if the presence or absence of an attribute (in this case mask), in a probe or gallery or both impacts existing algorithms, thus setting off the alarm bell ringing for development of new algorithms, we need a way to be able to match, agnostic to the presence of the attribute. And our work provides such methods.

In our work, we do not however consider face masks as a face attribute, only under the AAFES method, under Model level suppression, because that is the only method that allows us to suppress the feature at the model level, thus propagating to the final feature.

2.2 Effective and Efficient attribute detection approaches

2.2.1 Effective attribute detection approaches

The goal of the literature review under this section is to emphasize that effective face attribute detection is a significant research problem, and progress in the same also circles back into the efficiency

of our attribute aware matching method, since, we can leverage this progress, by using the SOTA face attribute detector. There have been several works to enhance attribute recognition accuracy. For instance, [26] describes a method for estimating multiple attributes from a single image of a face. These attributes may include demographic characteristics such as age, gender, and ethnicity, as well as other facial features such as facial expression, eyeglasses, and facial hair.

MOON [51] is a method based on a type of neural network called a mixed objective optimization network (MOON), which is designed to optimize multiple objectives simultaneously.

In the context of facial attribute recognition, the MOON model is trained on a dataset of images and corresponding attribute labels. The model is able to recognize a wide range of facial attributes, such as age, gender, ethnicity, and facial expression, by optimizing multiple objectives simultaneously. This allows the model to achieve higher accuracy than a single-objective model, as it is able to take into account the complex relationships between different facial attributes. While [52] describes a method for using convolutional neural networks (CNNs) to perform attribute-based active authentication on mobile devices.

Attribute-based active authentication is a method of verifying the identity of a user based on certain characteristics or attributes. In the context of mobile devices, this could include things like the user's face, voice, or typing patterns. More recently even MobileNets and other Resnet back-boned architecture have also shown good accuracy in Deep multi-task Learning (DMTL) settings using CelebA dataset

2.2.2 Efficient attribute detection approaches

This section looks at the literature review of SOTA compact networks, with the goal of adopting these compact models for the task of face attribute detection because our method depends on an efficient face attribute classifier for faster inference. MobileNet [30] and MobileNetV2 [53] are both convolutional neural networks (CNNs) designed to run efficiently on mobile devices. These networks designed by Google was in the direction of compact mobilenetworks.

One of the main differences between MobileNet and MobileNetV2 is the way they handle the depth-wise convolutions that are used in their architecture. MobileNet uses depth-wise separable convolutions, which involve breaking down a standard convolution into a depth-wise convolution and a point-wise convolution. This allows MobileNet to perform the convolution operation with fewer calculations, making it more efficient. MobileNetV2, on the other hand, uses inverted residuals with linear bottlenecks, which involves adding a linear layer to the traditional residual block used in CNNs. This allows MobileNetV2 to improve the efficiency of the network while also increasing its representational power.

Another difference between MobileNet and MobileNetV2 is the number of parameters in the network. MobileNet has about 4.2 million parameters, while MobileNetV2 has about 3.4 million parameters. This means that MobileNetV2 is slightly smaller and faster than MobileNet, although the difference in performance may not be significant in some applications, it is relevant for the face attribute classifier tasks.

Further MobileVits [42] fuse the concept of applying vision transformers over CNN feature maps, to produce highly efficient and effective models. The number of parameters here goes down to 1.1 million.

Chapter 3

Approaches to Attribute-aware Face Verification

3.1 Introduction

Face images when trained on large-scale public databases, such as Vggface2 has the ability to create embedding that is capable of ensuring verification accuracy of over 99.5 on some public evaluation datasets. However, when these trained models are inferenced on various test-datasets unseen during train (open-set verification) the resulting embedding is known to capture variations such as soft-biometrics and facial attributes. For example, [62] shows that attribute-rich dataset such as CelebA (open-set verification), the resulting embedding are capable of capturing soft-biometrics such as age, demographics, ethnicity, and facial-hair. Also, [46] that attributes clustered images are found at different layers of the face-space. Also, [54] have shown that templates constructed for similar poses yielded better verification accuracy. Finally, we too experimented and observe as shown in Fig3.1 that the presence or absence of an attribute in probe and gallery influences the verification accuracy of the attribute computed from the same embedding. This finding of ours on the specified facial attribute, motivated us to devise methods for better verification/matching by exploiting the prior knowledge of the presence or absence of a specific facial attribute. This prior knowledge can be obtained by a trained attribute detector or human-labels if available. For demonstrating our idea in this paper, human-labels available in the datasets that are being tested are used. The two proposed methods to obtain better verification performance by exploiting the prior information are discussed in the next two sub-sections. While the third subsection discusses the need and relevance of having two such methods.

3.1.1 Score level suppression: Configuration Specific Operating Threshold

In the first method, henceforth referred to as, CSOT (Configuration Specific Operating Threshold) three bins consisting of matching template pairs are created where both the templates of the matching pair in the first bin, possess the attribute, and in the second bin one template does and other does not, and in the third bin, both do not possess facial attribute and use different matching thresholds for each of these bins. Please refer Figure 3.2 These three bins/configurations/protocols henceforth are referred to as

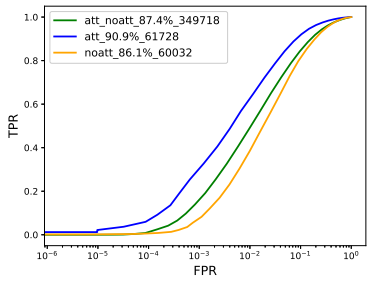


Figure 3.1: A plot for the 'Smiling' attribute, showing that matching operates differently depending on whether probe and gallery have the attribute in question or not. *att* in the plot above refers to probe and gallery having attributes. *att-noatt* is probe having the attribute and gallery without the attribute. *noatt* refers to probe and gallery not having the attribute.

att-att (short for attribute-attribute), att-noatt (short for attribute-no attribute) and noatt-noatt, our work leverages the facial attribute labels on challenging IJB-C dataset to create the three bins, each consisting of probe and gallery samples, conditioned on the bin type i.e att-att/att-noatt/noatt-noatt. An extensive set of 60000 pairs for each bin was created, and inference of SOTA networks was run on the same to determine the threshold. We have further applied this method over face-mask on the face images data we created over the LFW [34] dataset and for the Vision Transformer model trained on a large public dataset of faces, have shown an improvement in the matching Accuracy@EER and TAR@FAR1e-4 values.

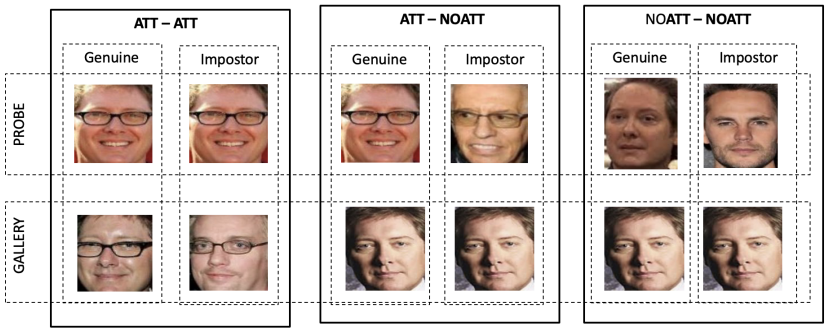


Figure 3.2: For both genuine-impostor pair, for eyeglass attribute, the first block is att-att, where probe and gallery both possess the attribute; The att-noatt where probe has the attribute but the gallery doesn't. And finally noatt-noatt is when both the probe and gallery do not possess the attribute.

3.1.2 Feature level suppression

Algorithm 2 demonstrates a generic Attribute aware feature suppression algorithm. The suppression vector denoted by *suppVector* is the only aspect that differs for *Attention Based Suppression* method

3.1.2.1 Attention Based Suppression

The Attention-based suppression falls under the paradigm of Model Level suppression since we suppress the attention-weights inside the model layers. One of the approaches that we employ with the goal of performing attribute agnostic matching is by turning to Vision transformers. A vision transformer model extends the Transformer architecture to process image data by treating the input image as a sequence of patches, and then each of these patches 'attends' to the other patches in a message-passing fashion to learn a comprehensive local and global representation of a class. Given that transformers have great explainability, the least inductive bias, and finally have achieved near SOTA accuracies on Face recognition as well, we consider attribute-aware face verification analysis on this model significant. For our purpose, the Vision Transformers, allow us to assign zero weights to attention maps at specific spatial regions where an attribute that we wish to be agnostic of is usually present. After preliminary examination of attention-maps in Figure 3.3 of a SOTA face recognition Vision Transformer model, we observe that vision transformers learn to weigh regions leading to least discriminability *less* (near the eye region) and weigh regions leading to most discriminability *more* (near the chin, cheek, forehead region). We therefore hypothesize that if a facial attribute found itself in the highly discriminable area that implies three things. One, the transformer model hasn't found enough samples of that attribute on the training set to ignore the specific attribute, because, if it had done so, the attention map wouldn't indicate it as a region of high discriminability. Two, if such an attribute does find itself appearing on the human face and we go through with matching faces adorning the attribute, the matching accuracy is bound to reduce. Three, if we zero-weight the attention weight of these highly significant spatial attention map regions, prior to matching the matching accuracy should improve. An attribute that exhibits these characteristics is a face mask. This is because face masks do not tend to be present on large open-source face datasets such MS1M RetinaFace available on InsightFace [50] [23] [21] [3] [11] [12] [24] [14] [13] or VggFace as it is a post-COVID-19 pandemic phenomenon. And hence, since the attention model is not trained to ignore this attribute during the matching process, the vision transformer attends to this feature, and when a probe image having the mask on and the gallery image not having the mask on, are matched, the excessive focus on the wrong region causes the drop in matching accuracy. Further, zero weighting these attention regions should improve face-mask matching performance. Conversely, if the vision transformer model has learned to ignore certain regions from the training sets, such as the eye region (due to the presence of the eyeglass attribute in the training data of these large datasets), zero weight this attention should cause a drop in matching accuracy since the attention maps are already fine-tuned to ignore these regions. We have experimentally verified this hypothesis, as, for the face mask attribute we have improved the accuracy@EER by almost 1% and TAR@FAR1e-4 by 0.6% where probe and gallery differ in their wearing of the face-mask. We have also sanity-checked that for low-attention regions such as eyeglasses, the accuracy@EER remains more or less constant. The methodology of how this is accomplished is detailed in the Methods section. Though improvement in face verification accuracy in the context of face-mask is an offshoot of our general analysis of attribute-aware matching, it is worthwhile to talk about the significance of the masked-face verification

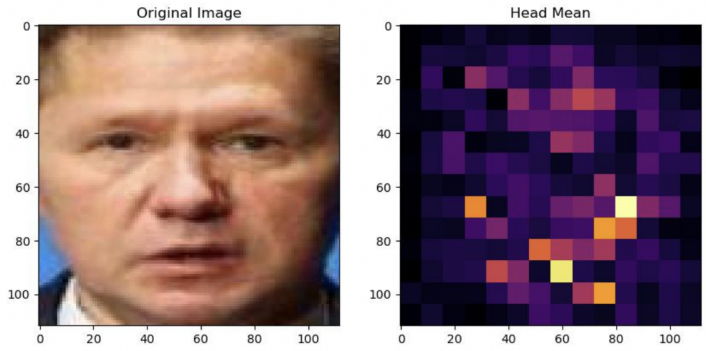


Figure 3.3: A typical face image has the following Attention map generated by the Vision transformer model trained on a large face training

domain.Masked face recognition is a challenging problem because wearing masks, particularly during times of public health concerns like the COVID-19 pandemic, has become a common practice. Masks cover a significant portion of the face, including the nose, mouth, and sometimes even the chin, which are crucial regions for facial recognition systems. Some reasons why masked face recognition poses difficulties are loss of facial features, limited visible information, altered face structure (due to stretch by tight masks), variability in mask types and training datasaet bias (since large training sets pre-COVID do not have masked subjects).

3.1.2.2 Attribute Aware Face Embedding and Suppression

In the second method, hence referred to as, AAFES(Attribute Aware Face Embedding and Suppression) given the understanding that verification accuracy is influenced by the presence of an attribute, an attribute-aware embedding is created and then devised a method to isolate the neurons most sensitive to a given facial-attribute, and suppress it. The conception of this embedding is that it should leverage the learning from a pre-trained state-of-the-art network on a large data and to this effect pre-trained InceptionResnetV1 trained on VGGFace is used, and thus serve as a face-embedding, while later layers of the DNN are fine-tuned to serve as attribute classifier, hence making it more plausible to suppress the neurons. Such a network had to be trained as attribute-aware face embedding because existing attribute embedding isn't suited for face verification. For instance, while there have been efforts to learn the correlations between labels of CelebA data, and effort was made to take the low/mid-level representation in [6], it is still based on the limited data of CelebA which is high class imbalanced and hence doesn't suit our goal of having high identity learning in addition to the attributes. Even this work [6] wonders in the conclusion section if pre-training could have helped learn a more robust attribute classifier.

It was noticed that, with a drop of about 5 percent face verification accuracy after the training above, the attribute recognition accuracy remains intact at 93,99.6 and 96 percent for attributes Smiling, Eyeglasses, and Mustache respectively. The verification accuracy was assessed using probe/gallery template

match detailed in 3.4, while the attribute recognition accuracy was measured from the fully connected network output.

3.1.3 Need for the Two Approaches

In this section, the need and application areas for two approaches stated above i.e. 3.1.1 and 3.1.2 are discussed.

The Score level suppresion based CSOT approach is relevant when it is desired to directly use SOTA face verification models (both public and COTS), with no access or resources to train our own. The above models can directly be inferenced over a pool of attribute-labelled dataset, and determine the operating threshold for att-att, att-noatt, noatt-noatt configurations.

The Feature level Suppression based methods *Attention Based Suppression* and *AAFES* based methods are discussed below.

The *Attention Based Suppression* methods can be used when one has a Vision Transformer based faced recognition trained model, and one is aware of which spatial region within a cropped face it is expected to find a given facial attribute. It is preferable that this facial attribute is either a rare attribute or never seen by the training set. (Since, if this was not the case, the model would have learned during training itself to ignore the presence of the noisy facial attribute)

The *AAFES* approach is primarily relevant when we have access to both compute and data that need to be fine-tuned on. We can retrain using our DNN model architecture, generate attribute-aware embedding, and further suppress the attribute information before matching. In addition to this one can piggyback on the other research areas that take interest in attribute-aware embedding, and directly apply our method of isolating the most sensitive neurons in the embedding, on the embedding from those methods. For instance, [48] attempts to create an embedding, that is capable of detection, landmark localization, pose estimation, and gender recognition. Embedding generated from attempts of this nature could be passed through the pipeline of our method, to get better verification accuracy. Also, Attribute-aware embedding has a lot of potential applications. They could be used in language tasks, as we can rely on the embedding to perform visual Q and A and other such language tasks. There have been several works to enhance attribute recognition accuracy [26] [51] [52] using multi-task and other nuanced approaches. Face recognition tasks also have been shown to improve by leveraging attribute information [22]. However, there are approaches that aim for a joint representation of both identity and attributes as in [32] because as noted here Face Attribute Feature (FAF) are more robust though less discriminative, whereas Face Recognition Features (FRF) is less robust but more discriminative. Other approaches such as [38] further analyze co-variation of attributes with generated embedding, and combined training used in [49] further denotes relevance of attribute aware embedding even if not captured in single embedding. In the work, [67] joint training in the multi-task setting of attributes and identity is performed, but for attributes that are invariant to the visual appearance of a person in a different situation (which is opposite to our goal). In the work [60] it is also attempted to create a joint representation of attribute and embedding (using a Kronecker product in the fusion layer). All

methods listed above highlight two important factors. (a) There is a direction to look for the combined embedding of attributes and identity (b) Also, the approaches don't aim to create such an embedding to beat the state of art embedding generated by discriminative Deep DNNs trained on massive data (using metric learning or triplet loss schemes, etc). The former point helps us assert our current direction of work involving both attribute-aware embedding and suppression of attribute information, while the latter justifies our attribute-aware embedding's lower accuracy compared to SOTA open-set embedding, despite being very relevant to fine-tuning on a given dataset of concern.

3.2 Problem Setting

It is important that the key dataset, attribute choice, and configuration setup assumptions is delineated before the next sections because the configuration setup is unique to this work for the problem at hand. And since the evaluation is also based on the configuration setup, the *previous results* section is also reported on this configuration setup

3.2.1 Verification Configurations Used in Our Methodology

In usual face verification evaluation methodology involves having a probe and a gallery set of genuine and impostor identities. However, since our work looks at leveraging attribute information for face verification, the genuine-impostor probe and gallery is now *conditioned on* the attribute label i.e. we first choose an attribute, and then create a probe-gallery set of genuine impostors inside it. This leads to three configurations:

- Attribute-Attribute (att-att): Probe and the gallery contain genuine-impostor pairs of persons *possessing the attribute*
- Attribute-NoAttribute (att-noatt): Probe and gallery contain genuine-impostor pairs of persons with *probe possessing the attribute but not gallery*
- NoAttribute-NoAttribute (noatt-noatt): Probe and gallery contain genuine-impostor pairs of persons *not possessing the attribute*

Given the above setup it is bound to reporting *attribute specific face verification accuracy* either on CelebA dataset (since they have attribute label annotations and identity annotations in the training set), or for eyeglasses attribute within IJB-C [41] as explained in *Choice of Facial Attributes* section 3.2.3

Therefore other face verification datasets (such as AGEDB, LFW) is not being reported on, because they don't have attribute label annotations.

3.2.2 Template Matching and CelebA Dataset

To the best of my knowledge, face verification accuracy on CelebA is being reported for the first time here.(*CelebA dataset is publicly available as of date*) This is not surprising given that any attempt to enhance face verification accuracy report on LFW, CFG-FP, AGE-DB, etc, however, they don't serve our purpose, because they don't have attribute label annotations. As you'll see in the next section below, attribute information is critical for binning our data into different probe and gallery sets.

Implicit labeling of IJB-C occlusion grid labeling has been leveraged to help identification of eyeglasses attributes.

3.2.3 Choice of Facial Attributes

3.2.3.1 Attributes Chosen for Experiments on CelebA Dataset

The choice of the attributes used for both approaches involves *five* attributes *Smiling*, *Eyeglasses*, *HeavyMakeup*, *Goatee* and *Mustache* which display a variation of the same identity in different situations. Figure 3.4 demonstrates this under section *Need intra-class attribute variation*. It is this variation that is aimed to combat by suppression for better matching. For CSOT method, *Eyeglasses*, *HeavyMakeup*, *Goatee* and *Mustache* is used; while for AAFES method *Smiling* attribute is used. The rationale for choosing this is that, as shown in Figure 3.4 under section *class imbalance* the positive class rate for the Smiling attribute is quite high.

Finally, we ignore attributes, that get left out after an MTCNN crop. Since the MTCNN crop is a primary operation in the DNN network, it doesn't serve our experimental analysis to include these features. Please again refer Figure 3.4 under section *MTCNN crop unfavourable attributes*

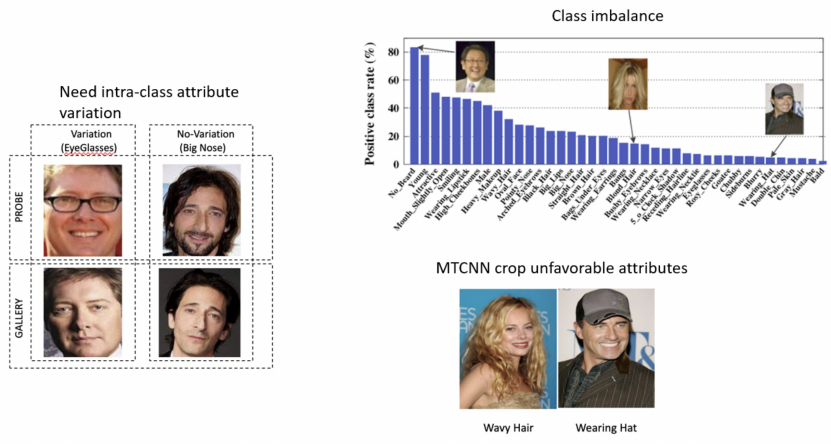


Figure 3.4: The three factors considered for the choice of attributes in experiments in Chapter 3

3.2.3.2 Attributes Chosen for Experiments on IJB-C Dataset

Eyeglasses attribute and *occluded forehead* are used in IJB-C dataset since that is an implicit label provided by IJB-C in their occlusion grid labeling on the eye region and forehead region respectively. This also satisfies the criteria of variation of the same identity in different situations as mentioned above. Refer to Figure 3.5 (Credit to https://www.nist.gov/system/files/documents/2017/12/26/readme.pdf for the figure) to see the occlusion grid, eyeglass label construction, and forehead occlusion label construction.

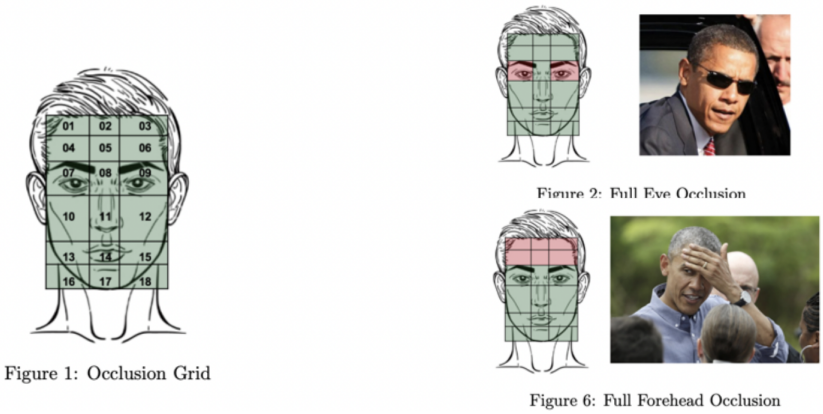


Figure 3.5: The IJB-C Occlusion grid labels are leveraged to determine the eyeglass and forehead occlusion labels

3.2.4 Deviation from "Subject-specific" Template Modeling in IJB-C Dataset

As defined in IJB-B paper [68] *...subject-specific modeling refers to a single template being generated using some or all pieces of media associated with a subject instead of the traditional approach of creating multiple templates per subject, one per piece of media.* However, this kind of modeling defeats our goal in this paper: to see the effect of attribute and probe in a given template. In our configuration setup (explained in 3.2.1), while using the IJB-C dataset (images and frames), instead of creating a subject-specific template in the gallery, image/frame from the gallery is itself used. I.e. Gallery contains images/frames with the attribute in question or without it.

3.3 Proposed Approach

3.3.1 CSOT: Configuration-specific operating threshold

As shown in Figure 3.6 the pipeline on the left of the figure shows two individual presented before the system to generate DNN embeddings, which is then matched to get the match score. The right side of the image shows the two individuals again presented to the system, but this time, in addition to the

DNN embeddings, the facial attribute is also detected as it of interest(in this paper however human-annotated attribute labels is used for experimental robustness), in each image presented, and depending on whether the pair of images have an attribute on, we determine the configuration/bin, and from the bin use a predetermined (by using a huge number of test pairs per bin) threshold value. Config-specific threshold value is now used to determine whether the pair is a match or a non-match The same is conveyed in Algorithm 1. Please note *FacialAttributeDetectorYesNo* method used in the algorithm is replaced by human annotated attributed labels in this paper.

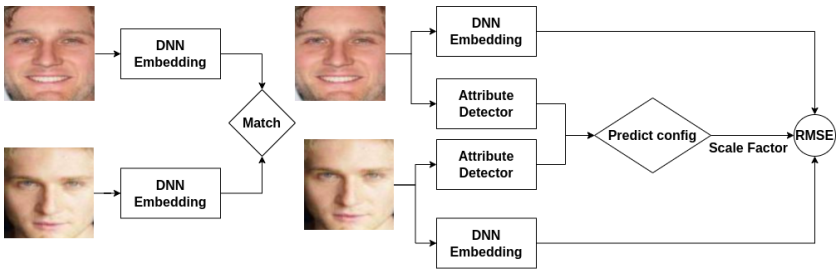


Figure 3.6: On the left, the regular DNN; Proposed method on the right, where we determine the config using attribute detector network, and use mapped scaling-factor(synonymous to unique threshold).RMSE block computes RMSE between two embeddings and multiplies with the scaling factor.

Instead of using a unique threshold for each bin, we can scale the distances of each bin to have a common threshold and derive a scaling factor instead to multiply the matching distance with. It is that *scaling-factor* that is being referred to in the figure 3.6. The reader can safely assume it is synonymous with a unique threshold per configuration/bin.

Here in Figure 3.7 is the intuition behind the mean shifting and scaling operation. The plot on top shows the three distributions i.e. att-att, att-noatt and noatt-noatt before the scaling operation, and thus we see three possible operating thresholds, i.e. one for each configuration. However, the plot below shows that after mean centering and scaling, we now have a unique threshold as opposed to three, and this is because the matching scores of each distribution is scaled accordingly. While this figure being discussed is schematic, a real distribution for near-SOTA network FaceNet on CelebA dataset's Eyeglass attribute is shown in the Figure 3.8. The highlighted marking in yellow shows the regions where each configuration distribution ends up different from the other configuration distribution.

On picking any of the left 4 figures in 3.16, for CelebA dataset, one can see the blue line with att-att config, the green with att-noatt config, and finally yellow with noatt-noatt config. The black line represents, the case where all three configurations co-exist in the data i.e. the data is now mixed. As it is observable, each configuration can best be operated upon, *with* the knowledge of the configuration. Refer to figure 3.6 that explains the same. But how do we determine if the difference in distribution is induced by the attribute and not a generic sampling distribution difference? For this, [62] is cited where it is shown that the state-of-the-art embedding *FaceNet* embedding, has tremendous attribute predictive power, and this evidence is used to back our experimental setup.

Algorithm 1 Config-specific operating point.

Require: N face image pairs to match
$binThresh \leftarrow$Threshold per config/bin inferenced from large test-set
$ta1 \leftarrow 0$ (Facial attribute yes/no for the first image)
$ta2 \leftarrow 0$ (Facial attribute yes/no for the second image)
$probeImage \leftarrow firstimagefrompair$
$galleryImage \leftarrow secondimagefrompair$
$config \leftarrow None$ (Placeholder for att-att, att-noatt, noatt-noatt)
$getThresh \leftarrow None$ (picks and returns appropriate threshold from binThresh for a given config)
$matchDist \leftarrow None$ (RMSE distance between image pair)
$t1 \leftarrow None$ (Face template generated by DNN for image 1)
$t2 \leftarrow None$ (Face template generated by DNN for image 2)
$thresh \leftarrow None$ (Threshold returned by getThresh for a given configuration)
$predict \leftarrow None$ (Final genuine impostor prediction by matching function)
Ensure: $i = 0, 1...Nmatchingpairs$
while $N \neq 0$ **do**
$t1 \leftarrow DNNEmbeddingGenerator(probe)$
$t2 \leftarrow DNNEmbeddingGenerator(gallery)$
$matchDist \leftarrow RMSE(t1, t2)$
$ta1 \leftarrow FacialAttributeDetectorYesNo(t1)$
$ta2 \leftarrow FacialAttributeDetectorYesNo(t2)$
if $ta1 = ta2 = 1$ **then**
$config \leftarrow att - att$
$thresh \leftarrow getThresh(config, binThresh)$
else if $ta1 = 0 and ta2 = 1$ **then**
$config \leftarrow noatt - noatt$
$thresh \leftarrow getThresh(config, binThresh)$
else$ta1 = ta2 = 0$
$config \leftarrow noatt - noatt$
$thresh \leftarrow getThresh(config, binThresh)$
end if
$predict \leftarrow getPredict(thresh, matchDist)$
end while

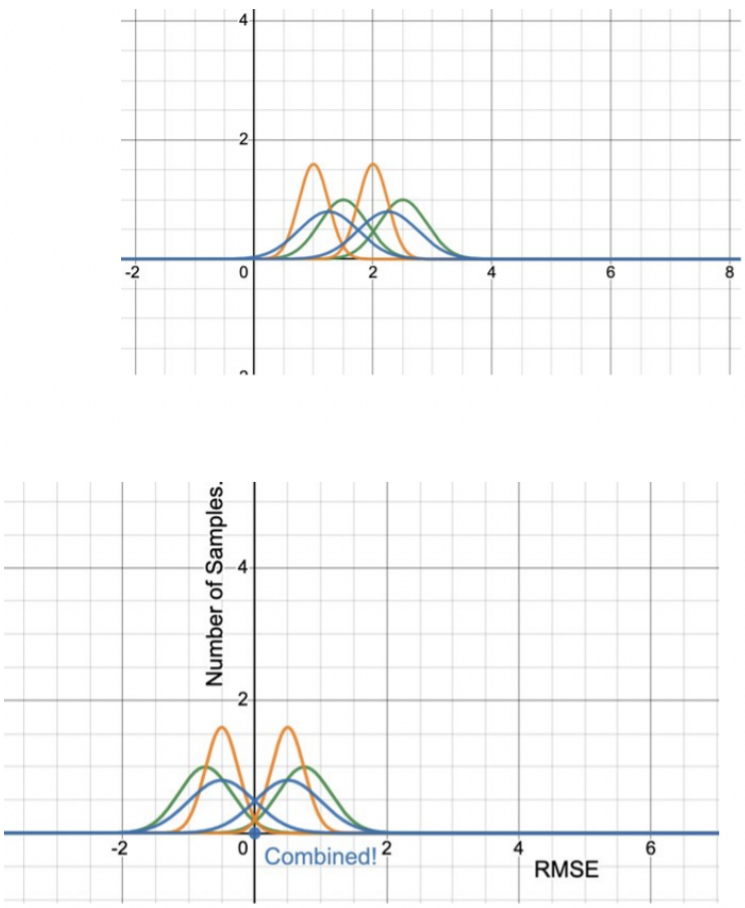


Figure 3.7: On top is a schematic plot where the plot in Orange (both genuine and impostor), Green, and Blue each refer to att-att, att-noatt, and noatt-noatt distribution

Please note that while preparing the graph 3.16 the following assumption has been made: Transparent eyeglasses attribute has been eliminated, and let only the dark glasses remain to avoid within-class variance).

The impact of the pose in the dataset to ensure there are no biased results were further analyzed. No impact of the pose.

3.3.1.1 Embedding Used and Choice of Facial Attributes for this Methodology

The embedding used to demonstrate this technique is InceptionResnetV1 pretrained on VGGFace2 (*the dataset has been removed from publicly available official page. Tested on licensed personal copy*) as made available by *FaceNet* [57], Arcface model pre-trained on MS1M [25], Magface [43] model pre-trained on MS1M dataset.

The choice of attributes of this methodology is the same as that discussed in the section 3.2.3.

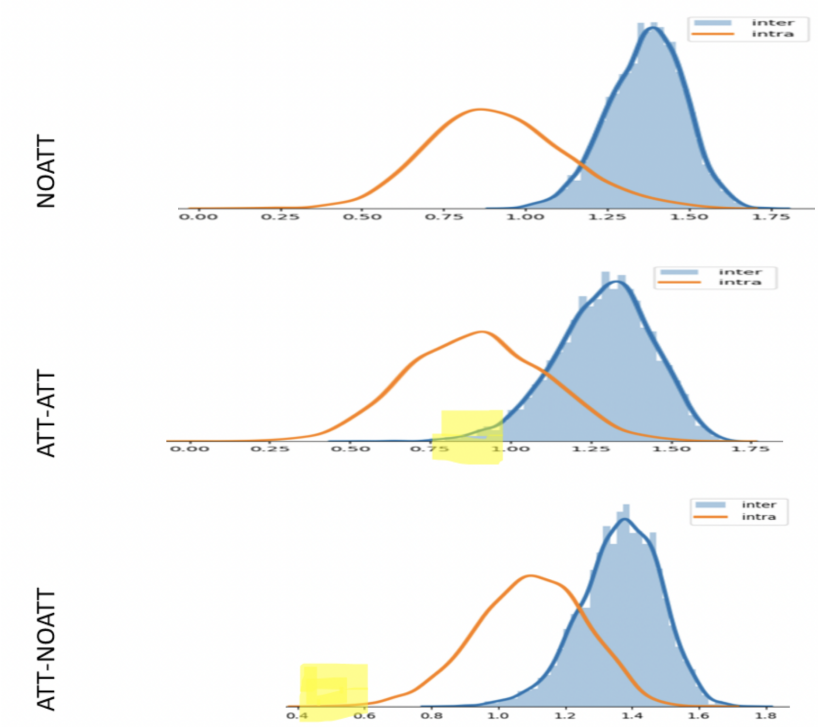


Figure 3.8: The plot here shows the genuine impostor score distribution of FaceNet network on CelebA dataset's Eyeglass attribute.The highlighted marking in yellow shows the regions were each configuration distribution ends up different from the other configuration distribution

3.3.1.2 Scaling the in-between Distribution Mean

While the above section offers an insight to operate individually at each scale, the mechanism to do the same is detailed below.

$$\forall x_i \in X_c$$

where c is configuration in question perform

$$\frac{x_i - \mu_{gc}}{\mu_{ic} - \mu_{gc}} \tag{3.1}$$

where $\mu_{gc}$is the Genuine mean, and μ_{ic} is the impostor mean defined for each of the configuration c i.e. att-att, noatt-att, noatt-noatt (for the rest of the paper, please assume att= attribute present. noatt = attribute absent i.e. no att) by passing a statistically relevant huge number of pairs through the trained network. Conceptually we are just zero-centering all the genuine mean and using the inter-class mean distance as the scaling factor. This operation helps us keep the threshold constant while scaling the match distance. The mean-shifting mentioned above as a conceptual operation lends itself to methods like parameter search of each of the configuration means using methods such as *Differential Evolution* to find configuration-specific mean. Scipy's implementation of the same has been used in graphs.

3.3.1.3 Application of CSOT for Masked Face Verification

We created a masked face dataset from the LFW dataset. The method is detailed in the section 3.3.2.0.2. We then run inference on the images by passing through two state-of-the-art networks (Magface, ArcFace), and finally, VIT trained by us detailed in section 3.3.2.0.1. We run each of these networks over the three-bin (refer 3.2.1 NoAtt-NoAtt, Att-Att, Att-NoAtt, and reported the Accuracy@EER and TAR@FAR1e-4. The findings of this experiment are reported in the result section 3.4.2.3 for CSOT

3.3.2 Feature level suppression: Attention Based Suppression

VIT, short for Vision Transformer, is a type of deep learning architecture specifically designed for computer vision tasks. It is based on the Transformer model, which was originally introduced for natural language processing tasks.

The Transformer model revolutionized the field of natural language processing by leveraging self-attention mechanisms to capture dependencies between words in a sequence. The success of Transformers in language processing tasks motivated researchers to explore their applicability in computer vision tasks as well.

The VIT model extends the Transformer architecture to process image data by treating the input image as a sequence of patches. These patches are linearly embedded and then processed by a series of Transformer layers. VIT removes the need for traditional convolutional layers that are commonly used in convolutional neural networks (CNNs) for computer vision tasks.

The key idea behind VIT is to leverage the self-attention mechanism to capture global and local relationships between image patches, enabling the model to learn representations and dependencies across the entire image.

Based on our preliminary observations, refer Figure-3.3 we speculate that the regions the transformer model turns its attention to are the regions where it can effectively address both inter-class and intra-class variability, and conversely, the regions it neglects are the ones where the variability doesn't contribute towards effective verification of either. Another thing to consider is that since we do not have explicit information on the facial attributes of each image in the huge training corpus, it is likely that for some attributes, such as (say) mask, the transformer is not capable of directing its attention to other relevant regions because it has been primarily trained on faces without masks. On the contrary, it is likely that for an attribute such as eyeglasses since its more likely for the subjects in the training data in both intra-class (as in the case of celebrities wearing dark glasses at different instances) and inter-class (as in the case of a subject who always wears eyeglass) the attention module "learns" to direct the attention away from such confusing regions. Thus the matching distance between a probe and gallery pair would be high for an intra-class pair, for a facial attribute such as a mask when the image in the probe wears the mask, but the image in the gallery does not wear a mask, primarily owing to the difference in the regions where the transformer attends to. However, for an attribute such as eyeglasses, since the transformer doesn't put emphasis on the eye region, the distance isn't huge.

With the above observation, we hypothesize that by suppressing the attention weightage on the region of the face where the transformer attends most, but is occluded by a facial attribute such as a face mask, we force the attention mechanism to emphasize the remaining regions of its attention. This particular excites us because we now have a novel method of achieving an increase in face-verification accuracy for attributes unseen during training by the Transformer, merely by zero weighting appropriate region-specific attention weights. While in this case, we have applied this for masks, we see it can be used for other applications such as bandannas, veils, etc, which too might be far few in number. Further, we can also apply this to classes that appear rarely as face attributes, such as bangs, chubby, etc as highlighted by CelebA dataset. To test the hypothesis we trained a VIT (Vision transformer) over the MS1M-RetinaFace dataset for over 30 epochs, and ran an inference of face with a mask over the model. As can be seen in the Figure 3.9 when the attention weights are zeroed out on all layers and heads of the transformer attention (to be discussed in the next section), we can see that the model turns to other regions of significance to make its prediction, whereas, if it was not zeroed out, it would still turn its attention to the mask region which constitutes a great source of noise especially when the probe image has a mask on, while the gallery image doesn't.

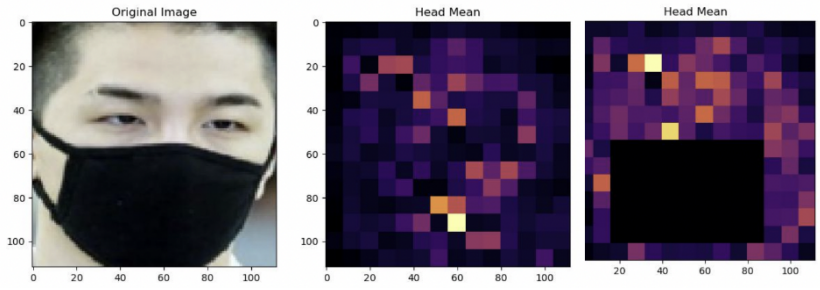


Figure 3.9: The first attention map (center image) constitutes the raw attention map of a VIT network when inferenced over a masked face image. The second attention map (rightmost image) constitutes the case where the face-mask region is 0 weighted in the attention map.

3.3.2.0.1 Network trained: The paper [72] extrapolates the Vision Transfomer paper [17] to Faces with the additional step that they modify the tokens generation method of ViT, to generate tokens with sliding patches, i.e., to make the image patch overlaps, for a better description of the inter-patch information, as shown in Figure 1.7. Each such patch created runs through a linear projection layer and is then fed to the multi-head transformer encoder block, and finally, the MLP head takes in the output of the transformer encoder to identify the identity corresponding to the individual. This network was trained over 30 epochs on the MS1M-RetinaFace made available by the InsightFace repository (cited previously). Learning rate of 1e-4 with 1 warmup epoch was performed for 20epochs and the final 10 epochs were training with the learning rate of 1e-5. The model achieved over 99.71% accuracy on the LFW dataset. We thus had a VIT face network ready for inference and for use in the next section.

Figure 3.10: First two images from left to right represent mask overlay on LFW images. The last two represent eyeglass overlay used for sanity checking attribute suppression

3.3.2.0.2 Mask Dataset creation: While several Mask Datasets are available, in our work we aim to perform an apple-to-apple comparison of the LFW dataset [34] with and without a mask. We do so by, pasting a mask image over the LFW face images by following the work of [5] who in their work pasted mask images of the child-face dataset. Please find first two images on left in Figure 3.10 that demonstrate the LFW face images before-and-after pasting with masks With the resulting image, both LFW images with masks pasted on them and the original LFW images, we then create three bins. Refer to details of the three fundamental bins we use in section 3.2.1 The first bin consists of the original LFW dataset, where the probe and gallery do not have masks. This is called the *NoAtt-NoAtt* configuration, with *NoAtt* being short for 'No-Attribute'. In the second bin, both probe and gallery images have masks. This bin is called the *Att-Att*, with *Att* being short for Attribute. The final bin is where the probe has the mask but the gallery doesn't and it's called *Att-NoAtt*.

3.3.2.0.3 Suppression mechanism: The Figure 3.9 consists of two attention maps; one, on the top right and the second on the bottom right. Each of these maps represents a grid of 14x14 numbers; This 14x14 number is nothing but the attention weights corresponding to 14x14 patches that the original 112x112 image was broken down into. Thus each number of the 14x14 grid when resized to the original 112x112 image represents how that region of the image attends to other regions of the image. We, therefore, set the values of attention to zero around the face-mask region, as can be seen in the bottom-right picture of Figure 3.9. This roughly corresponds to the pixel-region 73-110 along the height, and 20-90 along the width. Since the specific transformer blocks consist of 8 heads with 20 layers, after each layer, we set the mask region of the 14x14 attention weights to zero.

3.3.2.0.4 Sanity check with Eyeglasses attribute We have examined the effects of eyeglasses by introducing an artificial eye patch mimicking an eyeglass over LFW dataset. Please refer to the two rightmost images of Figure 3.10 to see a sample image. And then we zero weighted the attention map for the eye region (just as we did for mask in 3.9), and then performed the match Since the eyes region was least attended to as expected we didn't see an 'increase' in Accuracy@EER as was observed with the mask attribute. Please refer Table 3.1

Table 3.1: As a part of sanity check, the table demonstrates how LFW images overlayed with dark patches mimicking eyeglasses verification accuracy remains pretty much unchanged.

		Accuracy@EER	
	Configuration	VIT	VITSupp
Eyeglass	NoAtt-NoAtt	0.9968	0.9906
	Att-Att	0.9801	0.9745
	Att-NoAtt	0.9783	0.9768

3.3.3 Feature level susppression: AAFES: Attribute Aware Face Embedding and Suppression

The primary object as described in the pipeline 3.11 is to leverage identity-rich attribute-aware embedding, to first run an attribute detector over (in this paper however available human annotated attribute labels is used for experimental robustness). And once the attribute is known (say eyeglasses) suppression vector is applied, which is essentially a mask that has been created that masks out the most sensitive neurons to a given attribute, (details explained in this section) to zero out the neurons showing maximal correlation. The algorithms is given here 2. Note *FacialAttributeDetectorYesNo* in the algorithm, in our experiment is replaced with available attribute labels. It is to be noted that this work differs from the work [16], in that, correlation analysis is performed on the final embedding layer for a streaming validation data, as opposed to a lower dimensional representation of hidden layer analyzed through images in the cited work.

The details of how the suppression vector is created are the focus of the next two subsections

3.3.3.1 Motivation

A variation to the quantile streaming analysis is adopted, as was used in [20]. The cited work is deviated from in the aspect that, the activations of a given neuron (in our case, the embedding layer neurons) are gathered by passing the validation data into the model, and correlating it with the attribute label of the image, as opposed to performing quantile analysis on the same.

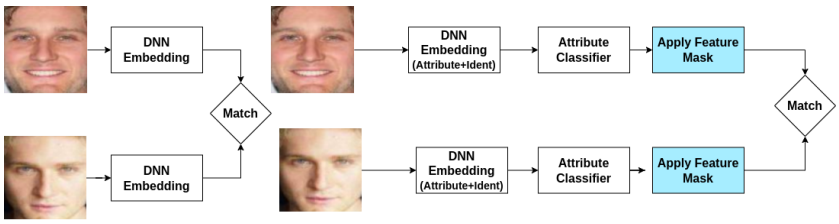


Figure 3.11: On the left, the regular DNN; Proposed method on the right, where config is determined using attribute detector network, and use mapped scaling factor.

Algorithm 2 Algorithm to execute suppression of attribute-aware embedding

Require: N image pairs to match
  $threshold \leftarrow$ Threshold determined by inferencing embedding over large test-set
  $suppressionVector \leftarrow$ Determined by our method for a given attribute
  $ta1 \leftarrow 0$ (Facial attribute yes/no for the first image)
  $ta2 \leftarrow 0$ (Facial attribute yes/no for the second image)
  $t1 \leftarrow None$ (Face template generated by DNN for image 1)
  $t2 \leftarrow None$ (Face template generated by DNN for image 2)
  $predict \leftarrow None$ (Final genuine impostor prediction by matching function)
Ensure: $i = 0, 1...N$ matching pairs
  while $N \neq 0$ **do**
    $t1 \leftarrow DNNEmbeddingGenerator(probe)$
    $t2 \leftarrow DNNEmbeddingGenerator(gallery)$
    $ta1 \leftarrow FacialAttributeDetectorYesNo(t1)$
    $ta2 \leftarrow FacialAttributeDetectorYesNo(t2)$
    if $ta1 = 1$ **then**
      $t1 \leftarrow t1 \odot suppVector$
    else
      $t1 \leftarrow t1$
    end if
    if $ta2 = 1$ **then**
      $t2 \leftarrow t2 \odot suppVector$
    else
      $t2 \leftarrow t2$
    end if
    $matchDist \leftarrow RMSE(t1, t2)$
    $predict \leftarrow getPredict(threshold, matchDist)$
  end while

3.3.3.2 Correlating Attribute Label with Embedding Neurons and Generating Suppression Vector

Let V be a $n1$ dimensional embedding, and L be a $n2$ dimensional attribute label vector (consisting of 0s and 1s). Let k be the number of samples in the validation dataset. For the k samples we now have a $n1 \times k$ matrix of embedding. We also have a $k \times n2$ labeling matrix for the k samples. Appending the V_i to the L_i, where i represents a particular sample we get a $n1 + n2$ dimensional vector P_i for each of k samples. Using the P matrix of P_i vectors we can now form a covariance matrix as follows:

$$C_{P,P^t} = \frac{\sum_{i=1}^{N}(P_i - \bar{P})(P_i - \bar{P})^t}{N-1} \tag{3.2}$$

Since the covariance matrix scales up the correlation as per the activation values it is dealing with, normalized correlation is performed to get the absolute value of correlation (independent of neuron activation) to determine which neuron relatively fires most. The relationship between the correlation coefficient matrix, R, and the covariance matrix, C, is

$$Rij = \frac{C_{ij}}{\sqrt{C_{ii} * C_{jj}}} \tag{3.3}$$

The matrix above can be decomposed as follows:

$$R = \begin{pmatrix} E & M \\ M^t & E^t \end{pmatrix}_{(n_1+n_2)\times(n_1+n_2)} \tag{3.4}$$

..where E and symmetric $E^T$is the normalized cross correlation between the embedding, and M and symmetric M^T are normalized cross-correlations between the embedding vector and the label vector for a given label. It is the M matrix of shape $n1 \times n2$ that is of interest to us in our suppression. Now for a given label $n2_i$, we have an embedding correlation vector $n1_i$ which is put into 10 bins in the histogram and index values corresponding to correlation value greater than the second topmost bin and less than bottom-most 2 bins are chosen. Refer Figure 3.12 for the same. The embedding size used is size 1792, penultimate to the fully-connected layer generated embedding of 512 on the InceptionResnetV1 network (while pre-trained on VGGFace2, trained on CelebaA by us). It is to be noted that performing correlation analysis on the final 512 embedding too works just as well. Interestingly while our trained network shows a high correlation for the discussed attributes, a similar attempt to check the correlation on the pre-trained embedding of 512 generated by InceptionResnetV1 on the same discussed attributes shows that all correlation values like just about 0.000. Thus showing no strong correlation of specific neurons with any attribute, while our embedding does.

3.3.3.3 Network Used and Training

The network used here is InceptionResnetV1 pretrained on VGGFaces2 as made available by *FaceNet* [57] as a starting point. The layers up to $ReductionB$ layer were frozen. Refer 3.13 for schematic diagram. This choice of using a pre-trained network and freezing initial layers was argued in [49] to be

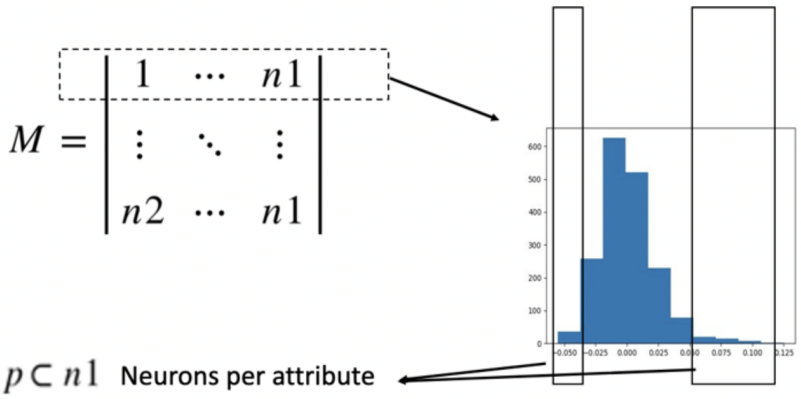


Figure 3.12: p is a subset of neurons most positively or negatively correlated with a given attribute

well suited for face analysis tasks (attribute detection in our case). The training was conducted on the Pytorch [47] platform.

Dropout from the penultimate layer was removed for ensuring that there is sparsity in the embedding generated for attribute learning. The remaining layers were trained on the $CelebA$ dataset with over 40 attributes and over 10,000 identities. Though the focus is only on 4 attributes outlined in the section 3.2.3, all the available attribute labels are leveraged, to exploit the attribute correlations in the multi-task setting. Pre-processing is limited to MTCNN [71] detection and RGB normalization. Attribute and identity accuracy on CelebA dataset are as follows. On attributes accuracies are Smiling - 93% , Goatee - 96 %, Heavy-Makeup -90.5% and Mustache - 96 % on Celeba. Since CelebA doesn't make an identity validation set available, the training set is split into 80-20 ratio, to determine a verification accuracy on a validation set of 91 %. These stats are just to show that our approach while doesn't claim a generic face embedding that can be SOTA (for instance the embedding generated by training above has 82% on LFWA dataset), finetunes to a specific dataset containing identity and attribute labeling, and thus enable both identity and attribute classification, and further using our proposed suppression method enhances the identity classification.

Our multi-task training architecture differs from [67] in that same final embedding for the classification of both tasks is used because it is desired that single embedding to encapsulate identity and attribute information exists, such that attribute neurons can later be suppressed. The network is denoted as $f(I;\theta)$, where θ is the parameter set of the deep architecture and I denotes the training images. Suppose we have M facial attributes and P face identities. Minimization of the expected loss is modeled as follws:

$$\Theta, W_a, W_p = argminL\left(\mathbf{I};\Theta, W_a, W_p\right) \tag{3.5}$$

where $\mathcal{L}\left(\mathbf{I};\Theta, W_a, W_p\right)$ is loss function defined of the task and defined as

$$\mathcal{L}\left(\mathbf{I};\Theta, W_a, W_p\right) = \mathcal{L}_a\left(W_a \cdot f(\mathbf{I};\Theta)\right) + \mathcal{L}_p\left(W_p \cdot f(\mathbf{I};\Theta)\right) \tag{3.6}$$

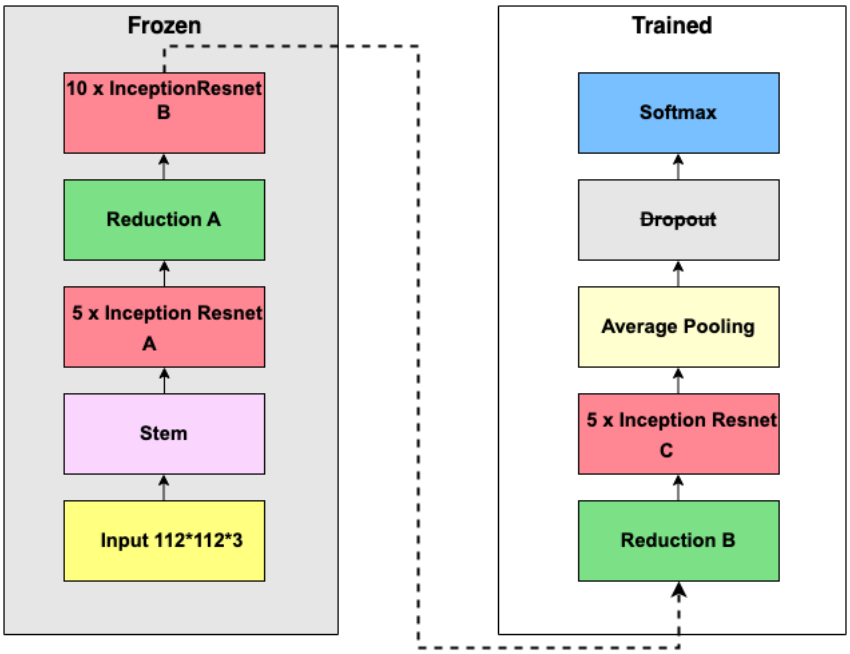


Figure 3.13: The left half is frozen, while the right half of InceptionResnetV1 is trained.

where $W_a \subseteq \mathbb{R}^{512\times 2\times M}$ (we have 512x2 here to accommodate a binary classification for each attribute, with CrossEntropy applied over it) and $W_p \subseteq \mathbb{R}^{512\times P}$ are the learned weights for facial attribute and face identification tasks.

3.3.3.4 Choice of Attributes in CelebA for AAFES Method

In addition to 3.2.3, for this particular method, Smiling attribute is chosen because it has more class balance and hence trains better. The class imbalance and hence the balance shown by the smiling attribute is shown in the fig 3.14

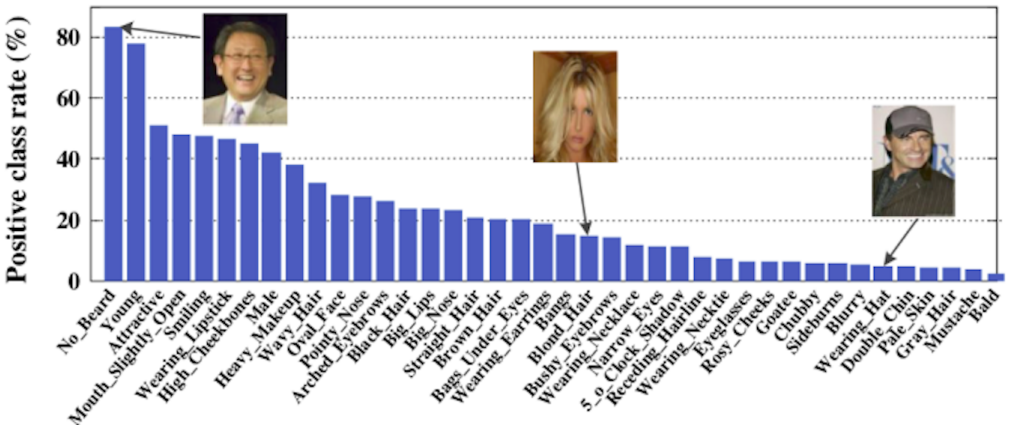


Figure 3.14: Positive class rate of the smiling attribute is balanced and labeling robust as well.

Table 3.2: Accuracy before and after suppression in percentage for available labels of high confidence from MAAD on VGGFace2

Attribute	Samples	Before (%)	After (%)
Smiling	3800	75	0.03
Eyeglasses	4100	98	0.08

3.3.3.5 Sanity Check of Attribute Learning and Suppression

It is critical that sanity checks are performed if indeed the attribute is learned by looking at the right regions of the image, and also if neurons that correlate most with a given face attribute are capable of being isolated. For the former, occlusion experiments were performed, while for the latter neuron suppression was done to see if face attribute predictions flip.

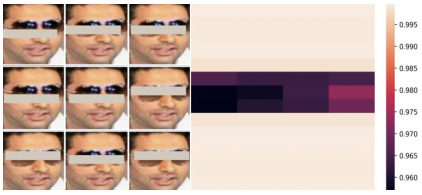


Figure 3.15: Pixel level occlusion patch to show the largest drop in accuracy. The same was performed for Smiling and bangs.

Occlusion Experiments: Since our methodology hinges on the activation of a neuron given a face attribute, in order to ensure that the model has learned the right regions, occlusion experiments were performed by patching various aspects of images and noticing drops in classification accuracy. In the fig 3.15 , you'll see the patched image on the left and the prediction confidence plotted on the right. The same was repeated for several other attributes such as *bangs*, and our subject attribute *smiling*.

Prediction flip with suppression: In order to check the effect of suppressing the neurons as deduced from the distribution of correlation value of embedding neurons, performed the sign flipping experiment, where I added to the activations a slightly positive value (about 1 or 1.5) for negatively correlated neurons, and subtracted the same value for activations with positive correlation, and check the effect of prediction on the accuracy of the attribute. Here is the accuracy for the attribute before and after the intervention, as applied on *VggFace2* dataset (with attribute labels picked up from *MAAD* annotations of VGGFace2 [63]. The attribute correlation values were derived from activations on a validation set of *CelebA*, and is being here as shown on another dataset i.e. *VggFace2*. Here is "Table 3.2" demonstrating the flip in attribute accuracy

3.4 Experiments and results

3.4.1 Evaluation Methodology

The evaluation method can be summarized as follows: A standard face verification evaluation involves generating genuine-impostor pairs from probe and gallery and then splitting the full list of pairs into train and eval in K-Fold manner. The training set here helps determine the optimum threshold for Equal Error Rate (EER), and the threshold is applied to the test, to get the test accuracy. The TPR/FPR is generated from the K-fold test set and averaged. Here the same is done, except that it is done for each of the configurations as detailed in 3.2.1 i.e. att-att, att-noatt and noatt-noatt by buckets the probe-gallery and generating genuine-imposter pair conditioned on the three configurations. Detailed steps below:

- Using MTCNN detector to get a clear region around the face.
- Splitting all images of CelebA or IJB-C (IJB-C relevant only to eyeglass attribute) dataset into two bins. The first bin has sub-bins, for each feature, and in turn, each of these bins contains all the identities who are identified with that feature. Similarly a second, has 40 sub-bin, for each feature, and in turn, each of these bins contains all identities who are *not* identified with that feature
- Half of all the images in the lowest bins are used for probe and the rest for a test.
- Creating pairs of images from the probe and gallery set above and iterating through them from disk with architected Pytorch DataLoader (including a change on their open source sampler program) to generate a maximum number of pairs, then generating their embedding, and further their RMSE distance
- For the generated RMSE and the Genuine/Impostor label assigned as 0/1, a validation split of 80-20 is done, with K-fold of 10.
- For each training set, a range of thresholds is evaluated and for the best threshold, the accuracy is computed on the validation (20 percent) set.
- This process is repeated for all 10 folds and average accuracy and TPR/FPR values are reported.
- Since there are a lot more impostor pairs at disposal compared to genuine pairs, the random genuine-pair-count number of images was sampled from impostor pairs, over 100 trials, and average accuracy was reported.

3.4.2 Results for Operating Point Adjustment by Mean Scaling

3.4.2.1 Results on CelebA Dataset

As can be seen in the graphs 3.16 plotted, where each row represents a particular attribute, the ROC graphs on the left, show the three individual configurations (att/att,att/noatt, noatt/noatt) in color, and

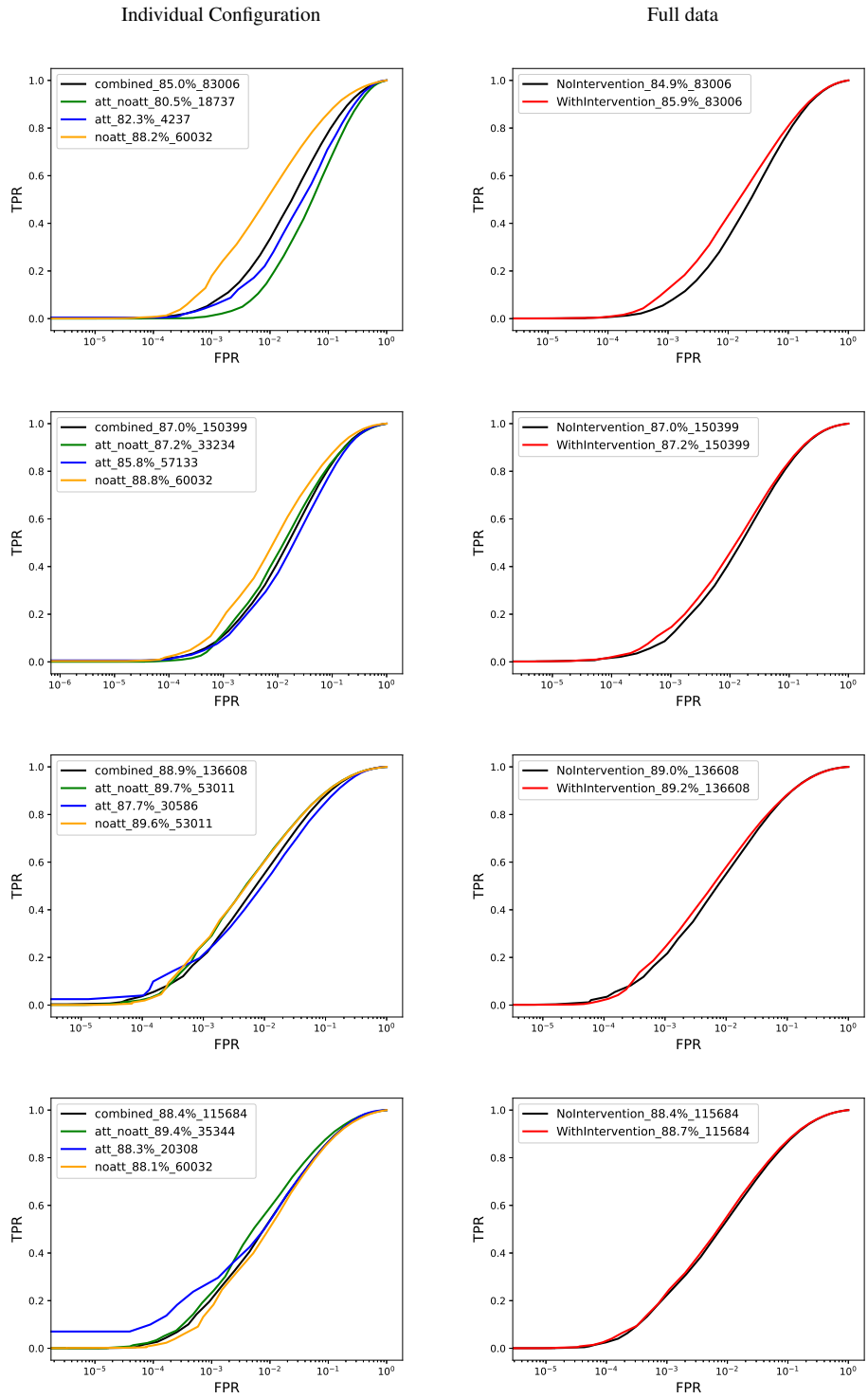


Figure 3.16: Top to bottom: Eyeglass, Heavymakeup, Goatee, Mustache. ROC plots on left are for individual configuration; And on the right on full data, with scaling in brown and; without-scaling in black. The labels on all the graphs are of the form Accuracy as a number; Intra/inter pair count and protocol.

the ROC of the configuration agnostic full pairs of images in black. The image on the right plots the configuration agnostic graph with and without scaling operation as described in 3.3.1.1. The result clearly shows that in most configurations our scaling approach beats the state-of-the-art at best by 1 % (Eyeglass and goatee). Tables showing accuracies 3.4 for InceptionResnetV1 pretrained on VggFace2 by FaceNet versus ours.

The table 3.3 lists the mean and variance before and after the scaling operation. This shows that once scaling and shifting are done, all three configurations end up with GMean (genuine mean) of 0, and IMean(impostor mean) of 1.

3.4.2.2 Results on IJB-C Dataset

The IJB-C dataset covers about 3,500 identities with a total of 31,334 images and 117,542 unconstrained video frames. Occlusion labeling was used(corresponding to occlusion grid numbers 07 and 09 of IJB-C https://www.nist.gov/system/files/documents/2017/12/26/readme.pdf) corresponding to the left eye region and the right eye region, to identify all the individuals wearing the eyeglass; Similarly, for attribute *occluded forehead* occlusion labels occ1,occ2,occ3,occ4,occ5 and occ6 was used. Further, the data was split into bins of att, noatt, att-noatt discussed in 3.2.1 and inferenced the two SOTA approaches, Facenet, ArcFace [13], Magface [43] over it. Our results in "Table 3.5" shows that our method 3.3.1.2 for *eyeglasses* attribute shows a significant improvement on earlier SOTAs such as Facenet and ArcFace by up to 1 % , while on the recent SOTA Magface, it equals it, showing that the SOTA Magface compared to other approaches, is much more robust in dealing with variation in attributes. For *occluded forehead*, an attribute which is more difficult compared to *eyeglass* (since a lot of eyeglasses in IJB-C dataset is the see-through eyeglass providing minimal but definite occlusion), our method improves by over 2 % over magface, while on Facenet it shows 1 % improvement, and ArcFace shows 0.5% improvement. 14900 template pairs of *occluded forehead* was used to report this, and 60000 template pairs of *eyeglass* attribute to report this.

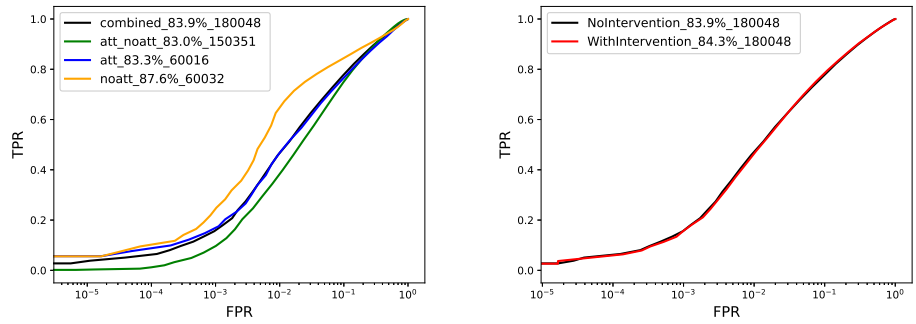


Figure 3.17: Top: ROC plots for Facenet on IJB-C dataset. From the accuracy numbers, one can see the average of att/noatt/att-noatt protocol is better than combined (in black) accuracy. Bottom: The plot shows an improvement in accuracy i.e. 0.5 % increase when mean scaling is done.

3.4.2.3 Results on LFW Mask Dataset

The results in Table 3.6 show that our method outperforms narrowly in the accuracy@EER metric for ArcFace and VIT i.e. 0.27% and 0.003% respectively, however on the TAR@FAR1e-4 metric we 1.4% improvement over SOTA Magface, and 2% improvement over ArcFace. We speculate that VIT's TAR@FAR1e-4 drops drastically because VIT pays strong attention to spatial image patches locally in addition to global representation, and given that masks were not a part of the training set the matching distance sees a spike. In order to explain this we tabulated the operating thresholds *Oper Threshold* in the table, and that shows that indeed VIT has a huge spike in the operating threshold, for the Att-NoAtt configuration, and that indeed contributes to a bad overall threshold and drop in TAR@FAR1e-4. And this is where the significance of our work shows itself, in that, since we are not having to deal with the idiosyncrasies of each model in how it deals with the matching configurations 3.2.1, we are consistently able to get a better TAR@FAR1e-4 metric.

3.4.3 Results for Feature level Supression

3.4.3.1 Results for Attention Based Suppression Method

3.4.3.1.1 Quantitative results The Table 3.7 demonstrates how for two configuration Att-Att and Att-NoAtt (refer 3.2.1) we were able to improve the matching accuracy@EER

3.4.3.1.2 Qualitative Results The figure Figure 3.18 represents the attention of all the 8 heads for the face image in Figure 3.9. The attention scores were computed by first gathering attention scores after each layer for each head, and then using attention rollout [1] across all layers for each head. This shows that several of the heads focuses attention regions primarily around chin, cheek, and forehead area, despite the mask being worn by the subject. Further, the bottom image of Figure 3.9 shows how by zeroing out attention near mask region (*observe the presence of a black patch in the attention map of the image located at the bottom right.*) the attention the overall attention weights now focuses on other regions of the face actively to help with better face matching.

3.4.3.2 Results for AAFES Method

3.4.3.2.1 ROC Curve for the Dataset with the Attribute in the Wild The ROC-curve 3.20, shows a 3 percentage improvement in the accuracy of verification after the suppression of the attribute.

3.4.3.2.2 Qualitative Results

- Figure 3.19 qualitative demonstrates our results.
- We also analyzed if a RMSE was to be taken only cosidering the suppressed neurons, the response was maximum when pair of images different in the presence of the attribute

Table 3.3: GMean is the Genuine mean and Gstd is Genuine Standard deviation. Likewise, Imean is Impostor mean

	Att-Att				Att-NoAtt				NoAtt-NoAtt				Full Data			
	GMean	GStd	IMean	IStd	GMean	GStd	IMean	IStd	GMean	GStd	IMean	IStd	GMean	GStd	IMean	IStd
Eyeglass Before Scale	32.34	7.49	46.5	7.74	42	6.54	53	6.46	38	7.79	55	6.55	38.69	7.87	53.74	6.92
Eyeglass After Scale	8e-05	0.53	0.99	0.54	0.0339	0.59	0.16	0.58	0.00	0.45	1.004	0.38	0.007	0.496	1.00	0.4901
Heavy Makeup Before Scale	33.32	7.30	52.90	7.18	40.16	7.18	55.13	6.25	38.06	7.75	55.46	6.55	38.24	7.53	54.38	6.82
Heavy Makeup After Scale	0.24	0.35	0.12	0.42	0.99	0.35	0.14	0.52	1.03	0.442	4.1	0.37	0.062	0.47	1.00	0.377
Goatee Before Scale	35.0015	7.66	51.98	6.83	38.9	7.54	55.88	5.95	38.97	7.54	55.88	5.95	38.086	7.75	55.25	6.26
Goatee After Scale	9e-05	0.45	1.00	0.40	0.994	0.445	1.002	0.35	0.994	0.44	1.99	0.35	0.003	0.446	1.0011	0.359
Mustache Before Scale	36.4	7.66	52.47	6.58	38.79	7.79	55.85	5.94	38.13	7.78	54.95	6.58	37.86	7.84	54.9	6.39
Mustache After Scale	-0.05	0.47	1.00	0.41	0.0004	0.46	1.00	0.39	0.00	0.45	0.99	0.34	-0.009	0.46	0.99	0.37

Table 3.4: Verification accuracy

Attribute	InceptionResnetV1	Ours
Eyeglasses	84.9	**85.9**
HeavyMakeup	87	**87.2**
Goatee	88.9	**89.3**
Mustache	88.5	**88.7**

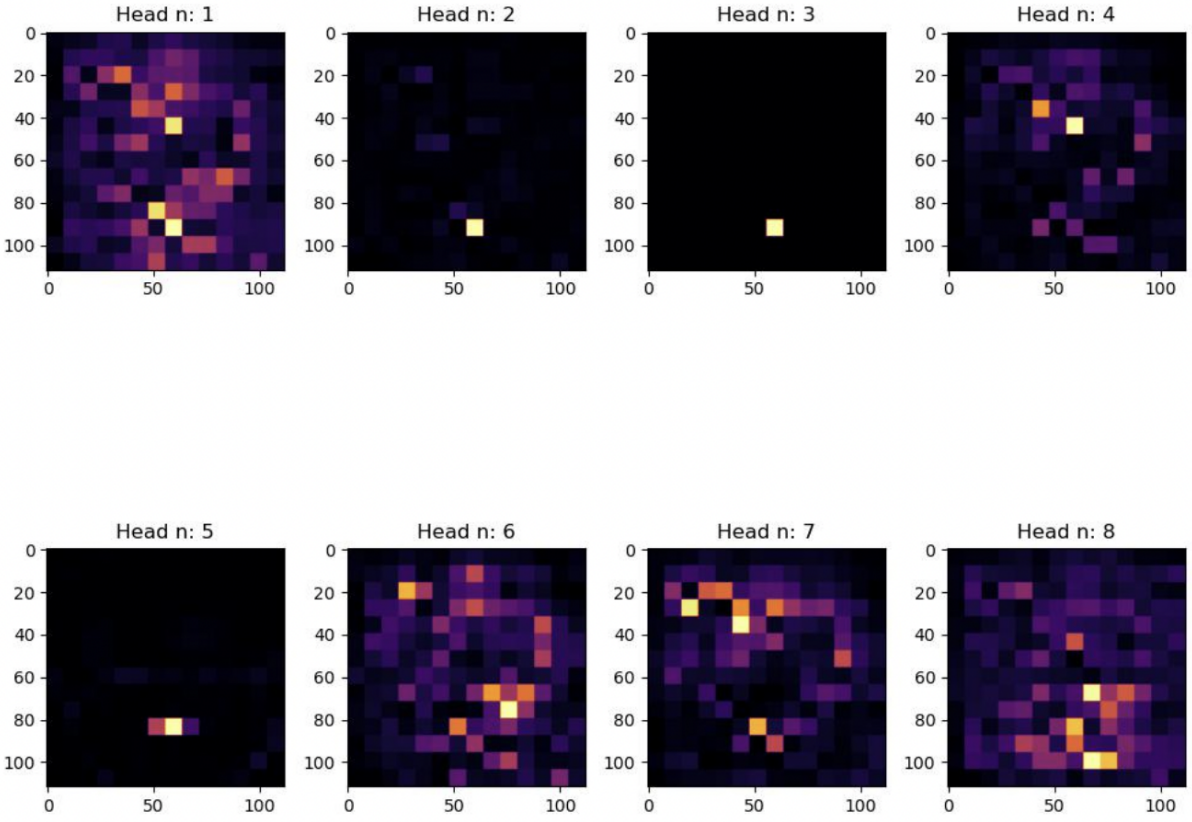


Figure 3.18: This is the out of each head for the face image with mask in Figure 3.9

Table 3.5: Accuracy post *att*, *noatt* and *att-nott* binning individually. *Without CSOT* refers to current SOTA; *CSOT*
is our method that uses individual bin thresholds and aggregates the result as explained

Attribute	Model	att	noatt	att-noatt	Without CSOT	CSOT (Ours)
Eyeglasses	Facenet	83.3	87.6	83.0	83.9	**84.6**
	Arcface	88.7	86.6	84.2	85.9	**86.4**
	Magface	95.9	92.3	90.8	93	**93.0**
Forehead Occlusion	Facenet	80.5	85.9	75.7	80.0	**80.7**
	Arcface	84.7	84.2	79.5	82.1	**82.7**
	Magface	93.3	93.0	85.8	88.7	**90.7**

Table 3.6: Accuracy@EER, TAR@FAR 1e-4 and Oper Threshold (operating threshold) for Masked Faces on LFW dataset. NoAtt refers to probe-and-gallery having no attribute (face mask in this case). Similarly Att refers to probe-and-gallery having the attribute

	Accuracy@EER					TAR@FAR1e-4					Oper Threshold			
	NoAtt	Att	Att-NoAtt	Combined	CSOT(Ours)	NoAtt	Att	Att-NoAtt	Combined	CSOT(Ours)	NoAtt	Att	Att-NoAtt	Combined
Magface	0.998	0.991	0.989	0.9935	0.9929	0.996	0.98	0.982	0.972	**0.986**	1.32	1.44	1.55	1.47
ArcFace	0.997	0.990	0.989	0.990	**0.9927**	0.996	0.973	0.974	0.961	**0.981**	1.49	1.63	1.51	1.53
VIT	0.996	0.976	0.970	0.978	**0.981**	0.995	0.975	0.978	**0.868**	**0.982**	1.65	**1.652**	**1.77**	1.72

3.5 Summary

In this chapter, we proposed two methods of exploiting attribute information available before matching. In the first case, we determined an ideal operating point for each configuration (att-att, att-noatt, noatt- noatt) separately, and used these operating points to match the pairs at test time (after determining whether each image in the matching pair has the attribute or not using attribute detector). To prove the validity of the same, we used a shift-scale method or parameter search using the Differential-evolution method over learned configuration-specific genuine-impostor mean values from training data and used the plots showed it beats state-of-the-art verification accuracy on CelebA dataset (for 4 listed attributes) and in case of IJB-C dataset beats SOTA for a tougher occluded- forehead attribute while equaling accuracy for eyeglass attribute. The second approach consists of two sub-approaches, in which one way

Table 3.7: The table demonstrates how LFW images overlayed with attribute i.e face-masks Verification accuracy drops for Eyeglasses after suppression, but increases for face-masks after suppression

		Accuracy@EER		TAR@FAR1e-4	
	Configuration	VIT	VITSupp	VIT	VITSupp
Face Mask	NoAtt-NoAtt	0.9968	0.9838	0.9955	0.9863
	Att-Att	0.97618	**0.9772**	0.9752	**0.9796**
	Att-NoAtt	0.9702	**0.9795**	0.9785	**0.9843**

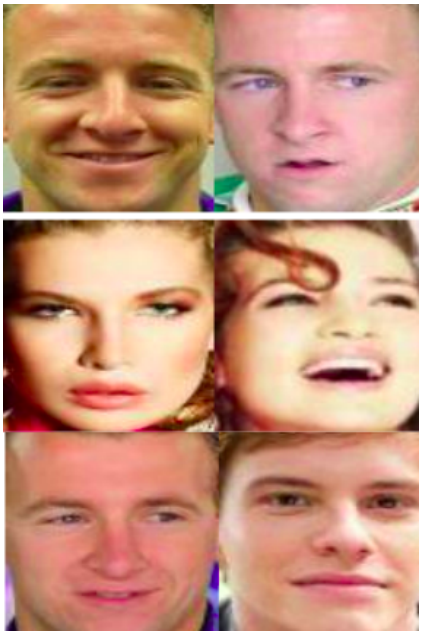


Figure 3.19: The top two rows are genuine pairs and the last row is the impostor pair matched correctly after the suppression of maximal activation.

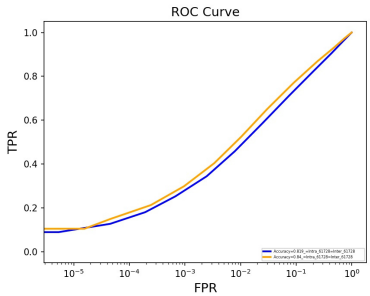


Figure 3.20: The yellow line demonstrates the improvement in matching after suppression.

is to use train a vision transformer for face recognition and suppress regions of the image that spatially corresponds to a given face attribute (face-mask in our research) and demonstrate an improvement in the Accuracy@EER and TAR@FAR1e-4. The other way to create attribute-aware embedding was demonstrated and it was shown that the verification accuracy can be increased by suppressing the neurons in the embedding correlating highly with a given attribute, thus showing a method to suppress the attribute information arguing that several applications and methodologies which generate such embeddings will benefit with the suppression to increase verification accuracy.

Chapter 4

Efficient Face Attribute-detection and Face Verification to Implement Attribute-Aware Face Verification

4.1 Introduction

While conventional methods have focused on quantization and pruning for compact models, Google has explored innovative approaches through EfficientNets and MobileNets. These approaches involve reevaluating the optimization of traditional convolution operations, considering depth-width-resolution trade-offs, and achieving a delicate balance in scaling. These advancements are crucial for our attribute-aware face verification, as the pipeline now includes detecting attributes for effective matching. Thus, both the DNN embedding model and face-attribute detection model must be highly compact.

In the models presented and analyzed in this chapter, the focus is on efficient training, and while FLOPs are not reported, it is guaranteed that all the models trained are highly compact. The rationale for this approach is explained in detail in the accompanying paper.[64] the following passage: *Note that floating-point operations (FLOPs) are not sufficient for low latency on mobile devices because FLOPs ignore several important inference-related factors such as memory access, degree of parallelism, and platform characteristics [39]. For example, the ViT-based method of [29], PiT, has 3x fewer FLOPs than DeIT [64] but has a similar inference speed on a mobile device (DeIT vs. PiT on iPhone-12: 10.99 ms vs. 10.56 ms). Therefore, instead of optimizing for FLOPs, this paper focuses on designing a lightweight, general-purpose, and low-latency network for mobile vision tasks.*

Also, the goal of this section is not to beat the state-of-art but to equal it at the least by using much lesser parameters. This goal emanates to enable our goal of attribute-aware face match, where face attribute detection must the robust and quick.

The facial attributes chosen for this chapter are *Eyeglasses*, *Mustache*, and *Goatee*. These attributes were chosen primarily because our work relating to attribute-aware face verification in chapter 3 3 uses all these attributes. Though the stated chapter uses other facial attributes in some sections, we are considering only the three here without any loss of generality.

After a brief introduction to existing model approaches, a few compact models that have been trained with the goal of balancing both effectiveness and efficiency, are discussed. In the case of face-verification, the DNN model is typically deeper, due to the challenging scenario of determining a unique template for the face image. Our challenge in this scenario is to find an effective experiment that can reduce the model size. While for attribute-detection the most compact model is attempted for use on CelebA dataset to classify facial attributes.

4.2 Overview of existing compact model approaches

MobileNets and EfficientNet are elaborated upon to provide a glimpse into how the stated compactness objectives are met. MobileNets(specifically MobileNet [30] and MobileNetV2[53]) are a family of convolutional neural networks (CNNs) that were developed specifically for mobile and embedded devices. They use a technique called depthwise separable convolutions to reduce the number of calculations required to perform the convolution operation, which makes them more efficient and suitable for model compression. MobileNets can be further compressed using techniques such as quantization, pruning, and distillation.

EfficientNets[61] are a family of CNNs that were developed to be both efficient and effective. They use several techniques to achieve this, including using a novel scaling method to automatically adjust the model architecture and using compound scaling to improve efficiency. EfficientNets can also be compressed using techniques such as quantization, pruning, and distillation.

4.3 Proposed Approach

4.3.1 Efficient attribute-detection DNNs

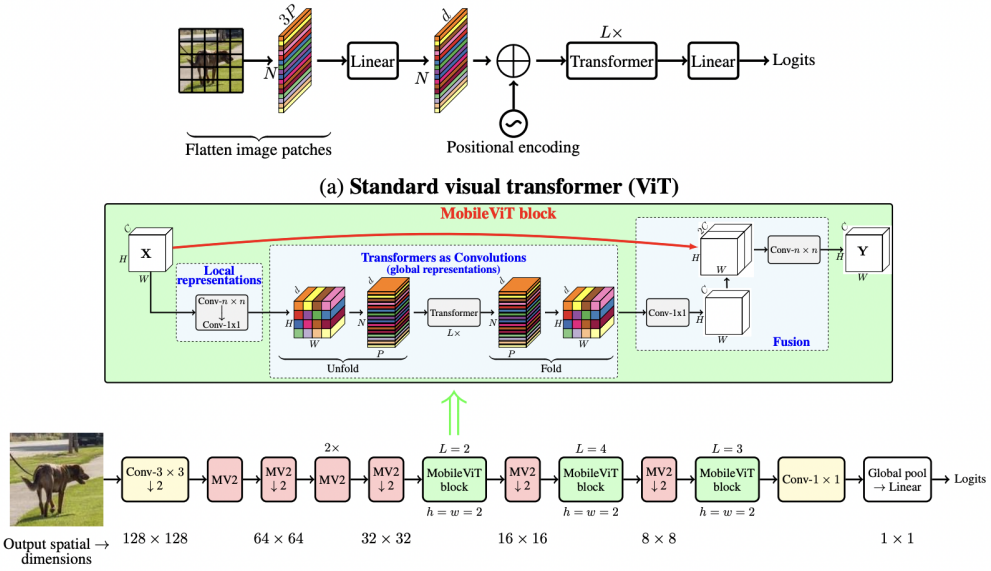
In the following subsections, the means of efficiently and effectively detecting attributes in two cases are discussed. One, training efficient SOTA architectures for face attribute detection purposes, and two, where we have a readily available fully trained DNN for face recognition (such as for the one used in CSOT 3.3.1).

For benchmarking [55] is referred. This uses two new DNN architectures (at the time), and they have matched this SOTA face-attribute detection accuracies.

4.3.1.1 Efficient DNN model architectures for end-to-end face attribute detection

Firstly, current attempts to leverage an efficient network for face attribute detection are delved into. One such effort is [55] where attribute detection is trained with about 2.4 million parameters. In addition, several more have been analyzed in the literature survey section 2.2. Our approach involves the usage of MobileViT architecture 4.1. The key idea used here is to apply a VIT block over the CNN layers; With

1.1 million parameters (almost half the 2.2 million parameters of MobilenetV2) same SOTA numbers are achieved. Please refer to 4.4.1.1 for the results.



(a) **Standard visual transformer (ViT)**

(b) **MobileViT**. Here, Conv-$n \times n$ in the MobileViT block represents a standard $n \times n$ convolution and MV2 refers to MobileNetv2 block. Blocks that perform down-sampling are marked with $\downarrow 2$.

Figure 4.1: MobileVIT architecture from the original paper [42].

4.3.1.2 Efficient attribute detection from trained Face Recognition DNNs

For CSOT method 3.3.1 intermediary layers (layer2 out of 4 available layers) of MagFace 1.3.2.3 are taken and create a separate attribute detection head. This choice is made keeping in mind, that intermediary layers are better suited for lower-level feature representation (Face Attribute) as opposed to final layers which contain face recognition-worthy templates to distinguish individuals. Refer Figure 4.2 Further average-pooling layer2 convolution feature maps, would be detrimental to our face-attribute classification task because we seek to preserve spatial features. Therefore channel compression a.k.a 1x1 convolution is more relevant. We, therefore, apply 1x1 convolution to 128 channels of layer2 output, to reduce (indicated by column-name *Layer2_Reduction_to*) it to as less as 10 channels, while preserving the feature-map size of 28x28. A great reduction in model size was achieved while nearing the SOTA accuracies. An ablation study was performed to see the impact of various 1x1 convolution and embedding layer size combinations. Please refer to 4.4.1.2 for the results.

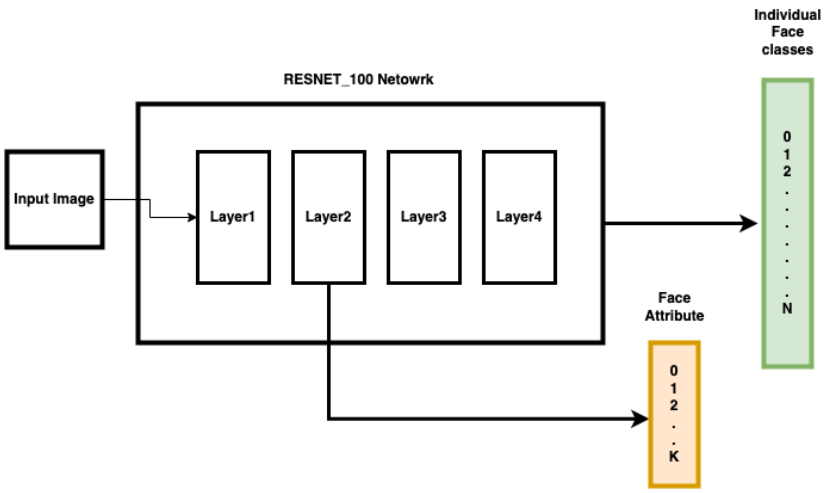


Figure 4.2: Figure showing attribute detector in orange, branched from the second layer.

4.3.2 Efficient Face Embedding DNN

The base network used for this approach is Resnet50 4.3 ResNet50 is a convolutional neural network (CNN) architecture that was developed by researchers at Microsoft and published in a 2015 paper. It is a variant of the ResNet architecture, which stands for ”Residual Network.”

ResNet50 is a 50-layer deep CNN that is trained on the ImageNet dataset, which is a large dataset of labeled images used for image classification tasks. It has been widely used as a base network for many computer vision tasks and has achieved state-of-the-art performance on a number of benchmarks.

Squeeeze-excitation layer 4.4 is however used over resnet50. Squeeze-and-Excitation Networks (SENet) are a type of convolutional neural network (CNN) architecture that was designed to improve the performance of CNNs on image classification tasks by enabling the network to adaptively re-calibrate the channel-wise feature responses in order to improve the representation power of the network.

SENets are based on the idea of ”squeezing” the global spatial information of the feature maps in a CNN and using that information to ”excite” the channels of the feature maps, which means scaling the channel-wise feature responses based on the global spatial information. This is done using a Squeeze-and-Excitation (SE) block, which consists of two main components: a Squeeze operation and an Excitation operation.

The Squeeze operation is a global average pooling layer that is used to reduce the spatial dimensions of the feature maps and extract the channel-wise statistics. The Excitation operation is a fully-connected layer that takes the output of the Squeeze operation as input and learns a set of linear transformations to scale the channel-wise feature responses.

In the stated network the fully connected layer contributes to almost 18 Million parameters (disk size of almost 70mb in addition). Of several attempts to reduce the model, mean pooling across 7x7 dimension of the 512x7x7 convolution output was performed to result in 512-sized embedding, achieving the same accuracy.

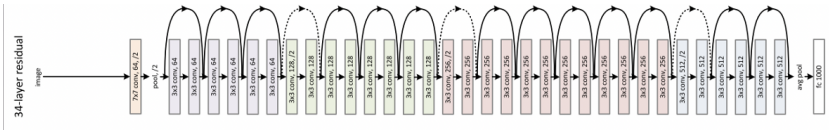


Figure 4.3: Schematic Diagram of Resnet34.Resnet50 however is in actual use as our model

[27]

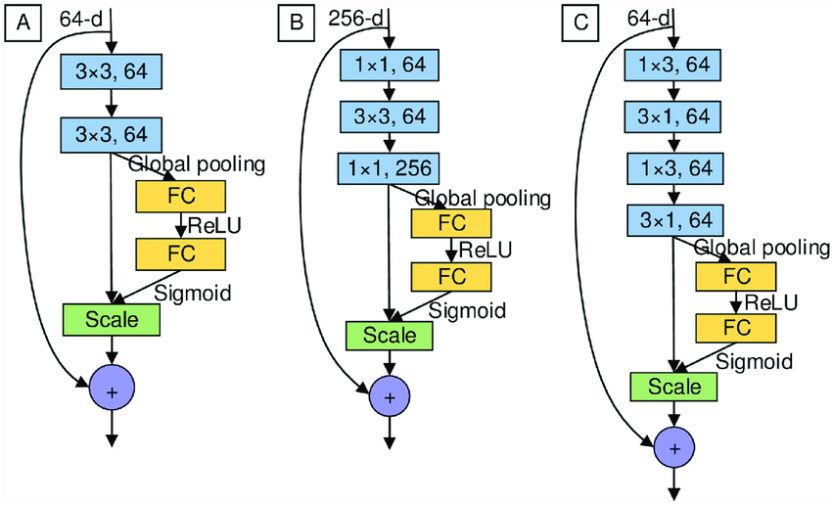


Figure 4.4: A typical squeeze excitation layer

[33]

4.4 Experiments and Results

- All the networks were trained using Pytorch, with optimizer being SGD with momentum value of 0.9 and weight decay of 5e-4. Networks are trained for around 100 epochs using distribute GPU training.

- For attribute detection, the dataset used for training the attribute detector is CelebA dataset with 162770 training images and 19962 test images. For face verification the Resnet50 based network was trained with MS1M datastet [25] of 7 million images, while evaluation was done on LFW [34], Age-DB[44]

4.4.1 Results for efficient attribute-detection DNNs

4.4.1.1 End-to-end attribute training

This section discusses the result of end-to-end attribute training referred to in 4.3.1.1. The key takeaway here is the half-the-model size reduction at keeping the face attribute accuracy nearly the same.

Table 4.1: Table showing achieving SOTA with half the number of params. The parameter Count is in Millions.

Network	Parameter Count	Eyeglasses	Goatee	Mustache
MobileNetV2	2.2M	99.3	96.3	96.5
MobileVIT	**1.1M**	**99.15**	**95.6**	**95.4**

4.4.1.2 Efficient attribute detection from trained Face Recognition DNNs

Table 4.2 demonstrated the face attribute detection for Eyeglasses, Goatee, and Mustache features, for various combinations of 1x1 convolution, layer, and final embedding size for the MagFace model (with Resnet100 backbone). As can be seen, the embedding length plateaus, around 128 to give slightly lower accuracies over embedding length 512 which gives the highest face attribute verification accuracy. *This table, therefore, shows, that if we are willing to compromise on a slight lack of accuracy we can reduce the train parameters and hence the model size to as less as 125K* It's worth that in comparison to SOTA methods of training an efficient attribute detection network end-to-end, took at least 2.2 million parameters. To reiterate the goal, the SOTA attribute detection is not attempted to be beaten here, but to find ways of using the highly compact model, with minimal loss of SOTA accuracy, to that extent, the ablation study below shows one can reduce the model size to upto 125K.

For extremely fast inference, (just 4.1K parameters) for 1.5 to 2 % drop in attribute detection accuracy, in the 4.2 configuration *OnlyFinalEmbedding* shows that that indeed is possible.

We further show that over all the attributes, using the magface layer2 outputs, and using a 1x1 convolution over it, to reduce to 10 channel, and then further using 512 embeddings, resulting in 4.1M params(column 1), and then 256 size embedding resulting in 2M param (column 2) channel sized embedding we are able to achieve near SOTA (refer [55]) attribute recognition accuracy 4.3

Another question to ask is, how ArcFace [13] (another SOTA network) which uses a metric learning approach to create an additional angular margin between the normalized class weight vector and face embedding of the same class, compare with MagFace which uses a similar metric learning approach but does not normalize thus not ignoring the magnitude component of the weight vector and embedding. How does ArcFace compare with MagFace when similar layers are chopped for final attribute classification? For this, an apple-to-apple comparison is made between similar layers' predictive power of an attribute. However, the early layers show a similar predictive power of the attributes. Showing that for robust prediction across several DNN architectures early layers could be considered. So both for initial and final layers ArFace and MagFace DNN embeddings behave similarly for final classification. Results in the table 4.4

4.4.2 Results for Efficient Face Embedding DNN

The results here below show the matching of state-of-the-art network accuracy while reducing the model size by almost 18 million parameters 4.5. A demonstration of how model size reduction such as this on both face recognition and face attribute detection, can together benefit the full pipeline.

Table 4.2: Table showing accuracy of face attributes varies, with taken representation at different layers.The configuration Layer2 implies activations are collected(by-passed) from this layer and passed to embedding layer. Whereas config OnlyFinalEmbedding refers to the final embedding of the network. Reduction_to means the reduction from Reduction_from number of channels down to a specified number. EmbeddingLength is the final embedding length (Embedding length is not applicable for Layer3 and Layer4 configuration because feature-map reduces to a very small size. Hence directly using the flattened layer)

Configuartion	Reduction_from	Reduction_to	EmbeddingLength	TrainParam	Eyeglasses	Goatee	Mustache
Layer2	512	20	512	8M	99.144	95.523	95.618
	512	20	256	4M	99.206	95.595	95.515
	512	20	128	2M	99.158	95.579	95.645
	512	20	64	1M	99.191	95.726	95.541
	512	20	32	503K	99.081	95.746	95.547
	512	10	512	4M	99.016	95.412	95.425
	512	10	256	2M	99.121	95.462	95.488
	512	10	128	1M	98.92	95.459	95.573
	512	10	64	500K	98.974	95.411	95.432
	512	10	32	250K	98.853	95.294	95.521
	512	10	16	125K	98.9	95.594	95.421
	512	5	512	2M	98.766	95.087	95.486
	512	5	256	1M	98.82	95.184	95.415
	512	5	128	500K	98.857	95.472	95.434
	512	5	64	250K	98.785	95.337	95.45
	512	5	32	125K	98.955	95.186	95.512
Layer3	256	80	n/a	146K	99.249	95.825	95.58
	256	20	n/a	36.5K	98.715	95.699	95.561
Layer4	128	80	n/a	72.5K	98.118	95.3	95.051
Layer4	128	20	n/a	18.1K	97.11	94.524	94.932
OnlyFinalEmbedding	n/a	n/a	512	4.1K	97.047	94.238	94.888

4.5 Summary

In this chapter, we have made a case for the necessity of effective and efficient face attribute detection and demonstrated ways of identifying the same. This was done by first surveying the existing methods of highly compressed models. (Mobilenets, Mobilevits, Efficientnets pioneered by Google research) and identifying the most current compressed model MobileVITs that are half the parameter size of mobilenetv2 and still perform with similar accuracy. We then asked the question of why attributes cannot be detected from deeply trained face recognition DNNs, and demonstrated that indeed can be done. While doing so, it is demonstrated that the final embedding of MagFace, ArcFace captures face attribute information (thus enabling a very small-sized face attribute detector at a cost of 1.5-2% accuracy points). It is also shown that early layers are more robust in face attribute prediction, however at a cost of slightly more parameters when compared to final layer models, however still substantially lesser in size compared to end-to-end face attribute detectors.

Table 4.3: Demonstrates that Magface embedding from early layers captures near SOTA accuracy over most attributes

Attribute	**4.1M Param**	**2.0M Param**	**SOTA**
5 o Clock Shadow	91.755	91.178	94.9
Arched Eyebrows	84.592	84.531	84.2
Attractive	80.266	79.996	82.7
Bags Under Eyes	82.723	82.657	85.6
Bald	98.555	98.519	99
Bangs	94.986	94.723	96.2
Big Lips	84.245	84.426	72.3
Big Nose	82.092	82.309	84.6
Black Hair	86.468	86.336	89.9
Blond Hair	92.975	93.107	96.0
Blurry	95.666	95.681	96.3
Brown Hair	77.546	77.478	88.8
Bushy Eyebrows	91.787	91.573	92.8
Chubby	95.277	95.09	95.8
Double Chin	96.29	96.273	96.5
Eyeglasses	99.141	99.116	99.6
Goatee	95.695	95.536	97.5
Gray Hair	97.33	97.327	98.3
Heavy Makeup	90.492	90.429	91.8
High Cheekbones	86.898	87.076	87.7
Male	97.609	97.493	98.4
Mouth Slightly Open	92.101	92.269	94.1
Mustache	95.609	95.665	97.1
Narrow Eyes	93.386	93.595	87.8
No Beard	93.953	94.004	96.5
Oval Face	74.828	75.027	76.0
Pale Skin	95.942	95.905	96.8
Pointy Nose	76.09	76.103	77.4
Receding Hairline	93.522	93.183	93.6
Rosy Cheeks	94.488	94.617	95.1
Sideburns	96.258	96.183	97.9
Smiling	92.343	92.344	93.1
Straight Hair	82.591	82.718	84.6
Wavy Hair	82.318	82.424	85.0
Wearing Earrings	84.648	84.8	90.8
Wearing Hat	98.514	98.568	99.1
Wearing Lipstick	91.052	91.141	93.9
Wearing Necklace	91.141	87.668	87.4
Wearing Necktie	95.675	95.679	96.8
Young	86.361	86.733	88.4
Average	**90.3302**	**90.237**	**91.5075**

Table 4.4: Predictive power of three facial attributes at different layers between MagFace and ArcFace

Configuration	Model	Param size	EyeGlasses	Goatee	Mustache
Final Layer Embedding	ArcFace	3.1K	96.526	93.38	94.86
	Magface	4.1K	97.047	94.238	94.88
1x1 Conv on early layers	Arcface	2.6M	99.042	95.704	95.439
	Magface	4M	99.206	95.412	95.425

Table 4.5: Verification accuracy of Reduced Model vis-a-vis original

Model	Param Size	LFW dataset Acc	AGE-DB Acc
ArcFace	61.9M	99.7	97.5
ArcFace WithoutFC	**43.7M**	**99.7**	**97.6**

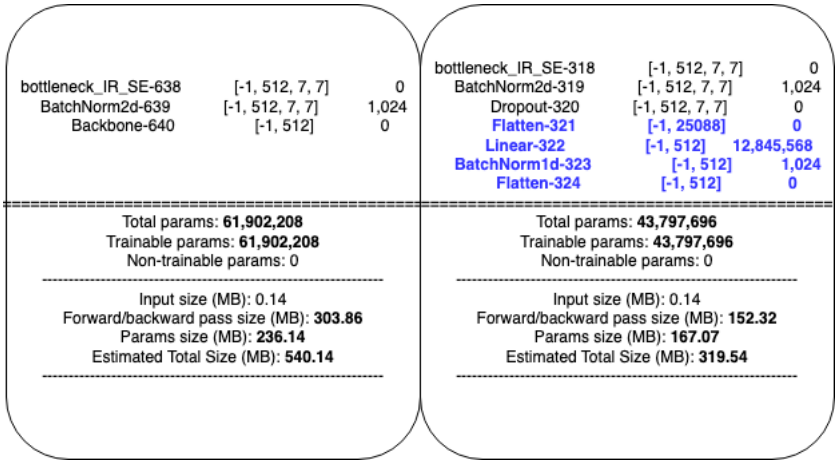


Figure 4.5: On the right is a fully connected final layer; On the left is our removal of the fully connected layer and introduction of average pooling. The blue font shows the extra parameters. The bold black numbers show the difference in size and params

Chapter 5

Conclusions and Future Work

This thesis looked at the fundamental question of leveraging facial attributes as prior information for matching. This was a natural question given that the face space has a high degree of variation, most of the common facial attributes are relatively low entropy and can be detected easily.

We first establish that when face verification is done within images with the same value for an attribute, it has a significant shift in matching scores. This was tested and demonstrated over a very large number of samples. Having established that, two methods are proposed to utilize this information to improve face verification. The first one was to find a unique operating threshold for each of the three possibilities of attribute values: the matching pair both possess the attribute, both do not possess the attribute and finally one possesses the attribute while the other does not. It is thus shown that operating on a unique threshold for each case improves matching performance. The other method devised consists of two sub-methods one of which was to train a Vision Transformer for face-recognition and demonstrate that for classes unseen by the training set or classes little seen (class imbalance) by the training set we are able to improve the matching accuracy by suppressing the spatial regions of the image that most correspond to an attribute in question, and the other was to suppress the attribute information in the template, thus neutralizing the variation caused by matching due to the presence of a facial attribute. To this end, correlation analysis was performed on the appropriate layer at the end of the DNN to suppress the maximally firing neurons given a facial attribute. When using embeddings that contain both attribute and identity information, we demonstrate that suppressing attribute information improves matching performance.

Since the method above, now adds a new aspect to the face verification pipeline i.e., attribute detection, chapter4 addresses ways to efficiently and effectively perform face attribute detection. Two methods of doing so were devised: compact end-to-end network and piggy-backing on learned attribute representations within deeply trained DNNs on a massive publicly available dataset for the purpose of face recognition to look for the presence of an attribute. The possibility of highly efficient and effective attribute detectors is established in this context.

Related Publications

- **Arun Kumar Subramanian** and Anoop Namboodiri, "On Attribute Aware Open-set Face Verification", *18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. (VISAPP)*, 19-21 February 2023, Lisbon, Portugal.
- **Arun Kumar Subramanian** and Anoop Namboodiri, "Face Verification through Attribute-based Attention with Transformers and Other Approaches", *Springer - CCIS Series*, (Under Submission).

Bibliography

[1] S. Abnar and W. Zuidema. Quantifying attention flow in transformers. *arXiv preprint arXiv:2005.00928*, 2020.

[2] R. Al-Refai and K. Nandakumar. A unified model for face matching and presentation attack detection using an ensemble of vision transformer features. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 662–671, 2023.

[3] X. An, X. Zhu, Y. Xiao, L. Wu, M. Zhang, Y. Gao, B. Qin, D. Zhang, and F. Ying. Partial fc: Training 10 million identities on a single machine. In *Arxiv 2010.05222*, 2020.

[4] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):711–720, 1997.

[5] P. K. Chandaliya, Z. Akhtar, and N. Nain. Longitudinal analysis of mask and no-mask on child face recognition. In *Proceedings of the Thirteenth Indian Conference on Computer Vision, Graphics and Image Processing*, pages 1–7, 2022.

[6] Z. Chen, F. Liu, and Z. Zhao. Let them choose what they want: A multi-task cnn architecture leveraging mid-level deep representations for face attribute classification. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 879–883, 2021.

[7] A. R. Chowdhury, T.-Y. Lin, S. Maji, and E. Learned-Miller. One-to-many face recognition with bilinear cnns. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9. IEEE, 2016.

[8] N. Crosswhite, J. Byrne, C. Stauffer, O. Parkhi, Q. Cao, and A. Zisserman. Template adaptation for face verification and identification. *Image and Vision Computing*, 79:35–48, 2018.

[9] N. Damer, F. Boutros, M. Süßmilch, F. Kirchbuchner, and A. Kuijper. Extended evaluation of the effect of real and simulated masks on face recognition performance. *Iet Biometrics*, 10(5):548–561, 2021.

[10] N. Damer, J. H. Grebe, C. Chen, F. Boutros, F. Kirchbuchner, and A. Kuijper. The effect of wearing a mask on face recognition performance: an exploratory study. In *2020 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–6. IEEE, 2020.

[11] J. Deng, J. Guo, T. Liu, M. Gong, and S. Zafeiriou. Sub-center arcface: Boosting face recognition by large-scale noisy web faces. In *Proceedings of the IEEE Conference on European Conference on Computer Vision*, 2020.

[12] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S. Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *CVPR*, 2020.

[13] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019.

[14] J. Deng, A. Roussos, G. Chrysos, E. Ververas, I. Kotsia, J. Shen, and S. Zafeiriou. The menpo benchmark for multi-pose 2d and 3d facial landmark localisation and tracking. *IJCV*, 2018.

[15] P. Dhar, J. Gleason, A. Roy, C. D. Castillo, and R. Chellappa. Pass: protected attribute suppression system for mitigating bias in face recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15087–15096, 2021.

[16] M. A. Diniz and W. R. Schwartz. Face attributes as cues for deep face recognition understanding. *CoRR*, abs/2105.07054, 2021.

[17] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[18] K. Etemad and R. Chellappa. Face recognition using discriminant eigenvectors. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 4, pages 2148–2151. IEEE, 1996.

[19] C. Ferrari, S. Berretti, and A. D. Bimbo. Discovering identity specific activation patterns in deep descriptors for template based face recognition. In *2019 14th IEEE International Conference on Automatic Face Gesture Recognition (FG 2019)*, pages 1–5, 2019.

[20] R. Fong and A. Vedaldi. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. *CoRR*, abs/1801.03454, 2018.

[21] B. Gecer, J. Deng, and S. Zafeiriou. Ostec: One-shot texture completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[22] E. Gonzalez-Sosa, J. Fierrez, R. Vera-Rodriguez, and F. Alonso-Fernandez. Facial soft biometrics for recognition in the wild: Recent works, annotation and cots evaluation. *IEEE Transactions on Information Forensics and Security*, PP:1–1, 02 2018.

[23] J. Guo, J. Deng, A. Lattas, and S. Zafeiriou. Sample and computation redistribution for efficient face detection. *arXiv preprint arXiv:2105.04714*, 2021.

[24] J. Guo, J. Deng, N. Xue, and S. Zafeiriou. Stacked dense u-nets with dual transformers for robust face alignment. In *BMVC*, 2018.

[25] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European conference on computer vision*, pages 87–102. Springer, 2016.

[26] H. Han, A. K. Jain, S. Shan, and X. Chen. Heterogeneous face attribute estimation: A deep multi-task learning approach. *CoRR*, abs/1706.00906, 2017.

[27] 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*, pages 770–778, 2016.

[28] R. Heitmeyer. Biometric identification promises fast and secure processing of airline passengers. *ICAO journal*, 55(9):10–11, 2000.

[29] B. Heo, S. Yun, D. Han, S. Chun, J. Choe, and S. J. Oh. Rethinking spatial dimensions of vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11936–11945, 2021.

[30] 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.

[31] R.-L. Hsu. *Face detection and modeling for recognition*. Michigan State University, 2002.

[32] G. Hu, Y. Hua, Y. Yuan, Z. Zhang, Z. Lu, S. S. Mukherjee, T. M. Hospedales, N. M. Robertson, and Y. Yang. Attribute-enhanced face recognition with neural tensor fusion networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3764–3773, 2017.

[33] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[34] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[35] A. K. Jain and S. Z. Li. *Handbook of face recognition*, volume 1. Springer, 2011.

[36] M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. Von Der Malsburg, R. P. Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on computers*, 42(3):300–311, 1993.

[37] Z. Liu, P. Luo, X. Wang, and X. Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018):11, 2018.

[38] B. Lu, J. Chen, C. D. Castillo, and R. Chellappa. An experimental evaluation of covariates effects on unconstrained face verification. *CoRR*, abs/1808.05508, 2018.

[39] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, pages 116–131, 2018.

[40] B. Maze, J. Adams, J. A. Duncan, N. Kalka, T. Miller, C. Otto, A. K. Jain, W. T. Niggel, J. Anderson, J. Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In *2018 international conference on biometrics (ICB)*, pages 158–165. IEEE, 2018.

[41] B. Maze, J. Adams, J. A. Duncan, N. Kalka, T. Miller, C. Otto, A. K. Jain, W. T. Niggel, J. Anderson, J. Cheney, and P. Grother. Iarpa janus benchmark - c: Face dataset and protocol. In *2018 International Conference on Biometrics (ICB)*, pages 158–165, 2018.

[42] S. Mehta and M. Rastegari. Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer. *arXiv preprint arXiv:2110.02178*, 2021.

[43] Q. Meng, S. Zhao, Z. Huang, and F. Zhou. Magface: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14225–14234, 2021.

[44] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou. Agedb: the first manually collected, in-the-wild age database. In *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 51–59, 2017.

[45] M. Ngan, P. Grother, and K. Hanaoka. Face recognition accuracy with masks using pre-covid-19 algorithms. In *NISTIR 8311*, 2020.

[46] A. J. O'Toole, C. D. Castillo, C. J. Parde, M. Q. Hill, and R. Chellappa. Face space representations in deep convolutional neural networks. *Trends in Cognitive Sciences*, 22(9):794–809, Sept. 2018.

[47] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[48] R. Ranjan, V. M. Patel, and R. Chellappa. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1):121–135, 2019.

[49] R. Ranjan, S. Sankaranarayanan, C. D. Castillo, and R. Chellappa. An all-in-one convolutional neural network for face analysis. *CoRR*, abs/1611.00851, 2016.

[50] X. Ren, A. Lattas, B. Gecer, J. Deng, C. Ma, and X. Yang. Facial geometric detail recovery via implicit representation. In *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*, 2023.

[51] E. M. Rudd, M. Günther, and T. E. Boult. MOON: A mixed objective optimization network for the recognition of facial attributes. *CoRR*, abs/1603.07027, 2016.

[52] P. Samangouei and R. Chellappa. Convolutional neural networks for attribute-based active authentication on mobile devices. In *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–8, 2016.

[53] 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*, pages 4510–4520, 2018.

[54] N. Sankaran, D. D. Mohan, S. Tulyakov, S. Setlur, and V. Govindaraju. Tadpool: Target adaptive pooling for set based face recognition. In *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, pages 1–8, 2021.

[55] F. Saxen, P. Werner, S. Handrich, E. Othman, L. Dinges, and A. Al-Hamadi. Face attribute detection with mobilenetv2 and nasnet-mobile. In *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pages 176–180. IEEE, 2019.

[56] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[57] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[58] Z. Sun and G. Tzimiropoulos. Part-based face recognition with vision transformers. *arXiv preprint arXiv:2212.00057*, 2022.

[59] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567, 2015.

[60] F. Taherkhani, N. M. Nasrabadi, and J. M. Dawson. A deep face identification network enhanced by facial attributes prediction. *CoRR*, abs/1805.00324, 2018.

[61] M. Tan and Q. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[62] P. Terhörst, D. Fährmann, N. Damer, F. Kirchbuchner, and A. Kuijper. Beyond identity: What information is stored in biometric face templates? *CoRR*, abs/2009.09918, 2020.

[63] P. Terhörst, D. Fährmann, J. N. Kolf, N. Damer, F. Kirchbuchner, and A. Kuijper. Maad-face: A massively annotated attribute dataset for face images. *CoRR*, abs/2012.01030, 2020.

[64] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021.

[65] M. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 01 1991.

[66] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. Ieee, 2001.

[67] Z. Wang, K. He, Y. Fu, R. Feng, Y.-G. Jiang, and X. Xue. Multi-task deep neural network for joint face recognition and facial attribute prediction. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, ICMR '17, page 365–374, New York, NY, USA, 2017. Association for Computing Machinery.

[68] C. Whitelam, E. Taborsky, A. Blanton, B. Maze, J. Adams, T. Miller, N. Kalka, A. K. Jain, J. A. Duncan, K. Allen, et al. Iarpa janus benchmark-b face dataset. In *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 90–98, 2017.

[69] L. Wiskott, N. Krüger, N. Kuiger, and C. Von Der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):775–779, 1997.

[70] J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li, and G. Hua. Neural aggregation network for video face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4362–4371, 2017.

[71] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10):1499–1503, 2016.

[72] Y. Zheng, H. Yang, T. Zhang, J. Bao, D. Chen, Y. Huang, L. Yuan, D. Chen, M. Zeng, and F. Wen. General facial representation learning in a visual-linguistic manner. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18697–18709, 2022.

[73] Y. Zhong and W. Deng. Face transformer for recognition. *arXiv preprint arXiv:2103.14803*, 2021.