2023

Deep Face Morph Detection Based on Wavelet Decomposition

Poorya Aghdaie
West Virginia University, pa00002@mix.wvu.edu

Deep Face Morph Detection Based on Wavelet Decomposition

Poorya Aghdaie

Dissertation submitted to the
Benjamin M. Statler College of Engineering and Mineral Resources at West Virginia University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Nasser Nasrabadi, Ph.D., Chair
Matthew Valenti, Ph.D.
Jeremy Dawson, Ph.D.
Omid Dehzangi, Ph.D.
Piyush M. Mehta, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2023

Keywords: Deep Learning, Wavelet Decomposition, Morph Detection, Group Sparsity, Attention Mechanism, Biometrics

Abstract

Deep Face Morph Detection Based on Wavelet Decomposition
Poorya Aghdaie

Morphed face images are maliciously used by criminals to circumvent the official process for receiving a passport where a look-alike accomplice embarks on requesting a passport. Morphed images are either synthesized by alpha-blending or generative networks such as Generative Adversarial Networks (GAN). Detecting morphed images is one of the fundamental problems associated with border control scenarios. Deep Neural Networks (DNN) have emerged as a promising solution for a myriad of applications such as face recognition, face verification, fake image detection, and so forth. The Biometrics communities have leveraged DNN to tackle fundamental problems such as morphed face detection. In this dissertation, we delve into data-driven morph detection which is of great significance in terms of national security.

We propose several wavelet-based face morph detection schemes which employ some of the computer vision algorithms such as image wavelet analysis, group sparsity, feature selection, and the visual attention mechanisms. Wavelet decomposition enables us to leverage the fine-grained frequency content of an image to boost localizing manipulated areas in an image. Our methodologies are as follows: (1) entropy-based single morph detection, (2) entropy-based differential morph detection, (3) morph detection using group sparsity, and (4) Attention aware morph detection. In the first methodology, we harness mismatches between the entropy distribution of wavelet subbands corresponding to a pair of real and morph images to find a subset of most discriminative wavelet subbands which leads to an increase of morph detection accuracy. As the second methodology, we adopt entropy-based subband selection to tackle differential morph detection. In the third methodology, group sparsity is leveraged for subband selection. In other words, adding a group sparsity constraint to the loss function of our DNN leads to an implicit subband selection. Our fourth methodology consists of different types of visual attention mechanisms such as convolutional block attention modules and self-attention resulting in boosting morph detection accuracy.

We demonstrate efficiency of our proposed algorithms through several morph datasets via extensive evaluations as well as visualization methodologies.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Problem and Motivation

This dissertation investigates the well-known problem of morphing attacks [9–17], which has draw considerable attention in biometric community in that morphed images have maliciously made the face recognition systems prone to false acceptance, having dire security consequences, especially for the national security. Morphed images have exploited loopholes in the face recognition checkpoints, e.g., Credential Authentication Technology (CAT), used by Transportation Security Administration (TSA), which is a non-trivial security concern. A synthetic morphed image can be verified against multiple subjects; thus, a blacklisted subject can circumvent official process of requesting a passport. Therefore, an innocent individual can request a passport using a morphed face image and a blacklisted subject can use the issued passport for cross-country trips. In Fig. 1.1, face images of two real subjects and their corresponding morphed image is displayed. Morphed images are mostly generated using two methodologies: (1) Landmark-based (2) Generative Networks such as Generative Adversarial Network (GAN). In the landmark-based face morphing [5,7,18], facial landmarks of two or more subjects are located and average of the landmarks of the subjects' faces are found to account for the landmarks of the morphed image. Once the averaged landmarks are found, subjects' faces are aligned to the common averaged landmarks and the aligned real images are alpha-blending in order to generate a morphed image (see Fig. 1.2). On the other hand, in the generative method for face morphing [1, 7, 19], a generative network

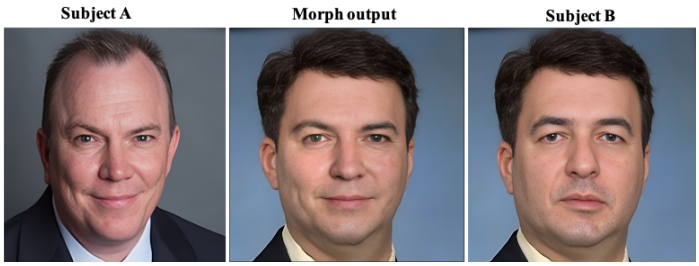


Figure 1.1: Examples of face images of two real subjects and their corresponding morphed image.

such as GAN which has an attached encoder network is trained to capture data distribution. In order to generate morphed images of two or more subjects' faces, the trained encoder network of the GAN converts RGB face images into latent vectors and the alpha-blending is achieved in the latent domain (see Fig. 1.3). Due to significance of morphed images in terms of national security, we have delved into detection of morphed face images and we have proposed several methodologies detailed in the following chapters. Please note that morph detection can be achieved in a single-image setting or a differential method. single-image face morph detection means labeling an image as either real or morphed without extra information. However, differential morph detection means labeling a face image as real or morphed using extra information, which is a live capture of a subject's face. Scope of this dissertation is the single-image face morph detection.

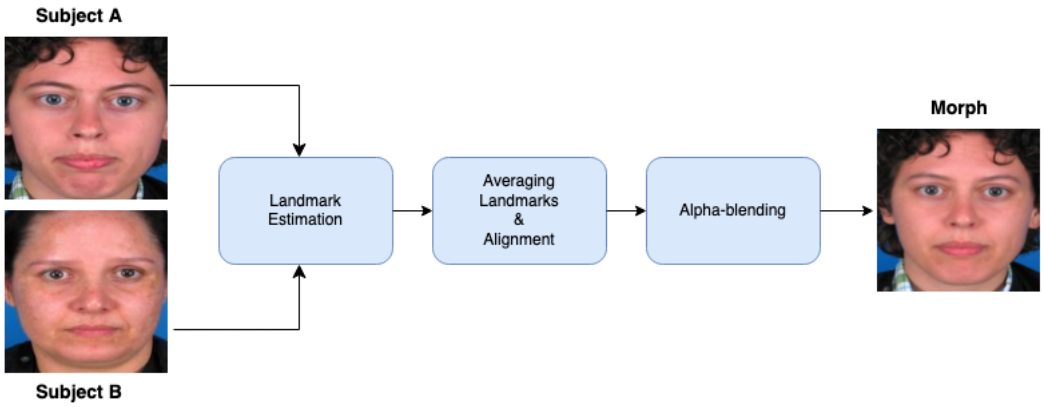


Figure 1.2: Landmark-based face morphing pipeline. The real face images are samples from the Utrecht ECVP face dataset [3] and the morph face image is from the VISAPP17 morph face dataset [4,5].

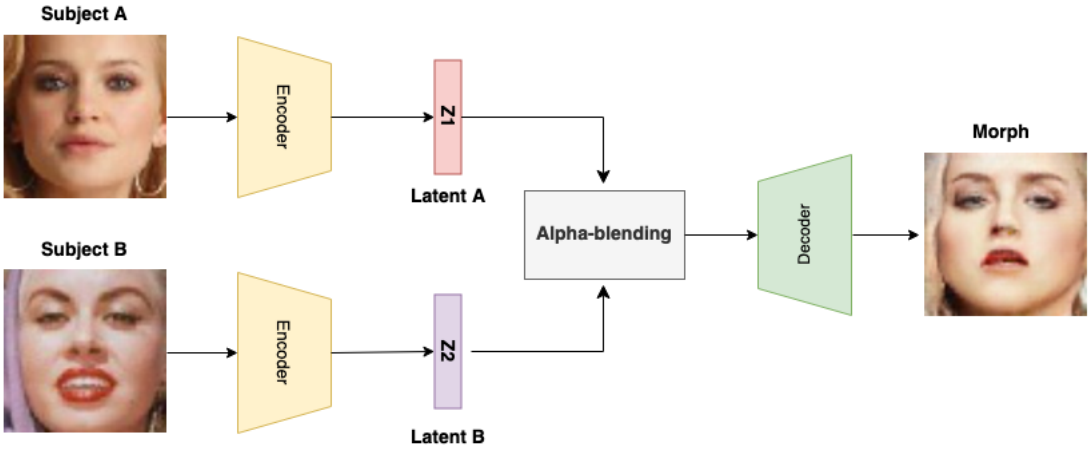


Figure 1.3: Face morphing pipeline based on a generative network. The real face images are samples from the CelebA dataset [6] and the morph face image is from the MorGAN face dataset [7].

1.2 Outline and Contributions

To detect morphing attacks, we propose several methods which are based on a discriminative 2D Discrete Wavelet Transform (2D-DWT). A discriminative wavelet sub-band is able to highlight inconsistencies between a real and a morphed image. In chapter 2, we observe that there is a salient discrepancy between the entropy of a given sub-band in a bona fide image, and the same sub-band's entropy in a morphed sample. Considering this dissimilarity between these two entropy values, and to generalize our method to all images in a dataset, we find the Kullback-Leibler divergence between two obtained distributions, namely entropy of the bona fide and the corresponding morphed images in the dataset. The most discriminative wavelet sub-bands with the highest corresponding KL divergence values are selected. Accordingly, 22 sub-bands are selected as the most discriminative ones in terms of morph detection. We show that a Deep Neural Network (DNN) trained on the 22 discriminative sub-bands can identify morphed samples accurately.

In chapter 3, we propose a wavelet-based morph detection methodology which adopts an end-to-end trainable soft attention mechanism [20]. Our attention-based DNN focuses on the salient Regions of Interest (ROI) which have the most spatial support for morph de-

tector decision function, i.e, morph class binary softmax output. A retrospective of morph synthesizing procedure aids us to speculate the ROI as regions around facial landmarks , particularly for the case of landmark-based morphing techniques. Moreover, our attention-based DNN is adapted to the wavelet space, where inputs of the network are coarse-to-fine spectral representations, 48 stacked wavelet sub-bands to be exact. In addition, as attention maps can be a robust indicator whether a probe image under investigation is genuine or counterfeit, we analyze the estimated attention maps for both a bona fide image and its corresponding morphed image.

In chapter 4, we decompose every image into its wavelet sub-bands using 2D wavelet decomposition and a deep supervised feature selection scheme is employed to find the most discriminative wavelet sub-bands of input images. To this end, we train a DNN morph detector using the decomposed wavelet sub-bands of the morphed and bona fide images. In the training phase, our structured group sparsity-constrained [21] DNN picks the most discriminative wavelet sub-bands out of all the sub-bands, with which we retrain our DNN, resulting in a precise detection of morphed images when inference is achieved on a probe image.

In chapter 5, we propose a morph detection framework to find the most discriminative information across frequency channels and spatial domain. To this end, we propose an end-to-end attention-based deep morph detector which assimilates the most discriminative wavelet sub-bands of a given image which are obtained by a group sparsity representation learning scheme. Specifically, our group sparsity-constrained DNN learns the most discriminative wavelet sub-bands (channels) of an input image while the attention mechanism captures the most discriminative spatial regions of input images for the downstream task of morph detection. To this end, we adopt three attention mechanisms to diversify our refined features for morph detection. As the first attention mechanism, we employ the Convolutional Block Attention Module (CBAM) [22] which provides us with refined feature maps. As the second attention mechanism, compatibility scores across spatial locations and output of our DNN highlights the most discriminative regions, and lastly, the multiheaded self-attention augmented convolutions [23, 24] account for our third attention mechanism.

Finally in chapter 6, we conclude the dissertation through summarizing methodologies proposed for single-image morph detection and future works are explained.

1.3 Literature Review

1.3.1 Morph Generation

Facial morph generation techniques are categorized into two types, i.e., landmark-based morphing [5,7,13,18,25], and morphing using a generative network [1,7,26]. In the landmark-based morphing attack, appearance of a resulting morphed image is associated with that of two underlying subject's bona fide face images, while geometric locations of its landmarks are the average of the corresponding landmarks in the two bona fide images [27]. Every pixel location in both bona fide images is warped to preserve the correspondence in the resulting morphed sample, and a convex combination of the warped pixels cross-dissolve the warped pixels in the bona fide images to synthesize that pixel location in the final morphed sample. By applying Delaunay triangulation on the two bona fide images, corresponding regions on the two facial images are further warped and mixed through alpha blending to synthesize the morphed image.

Generative methods have shifted the photo-realistic image synthesis paradigm considerably [1, 28–30]. Generative Adversarial Networks (GANs) are also employed for synthesizing morphed images. GAN-based image morphing techniques focus on the distribution of bona fide images. A trained GAN is able to find the distribution where data samples are drawn from. The point here is that a convex combination is generated in the latent domain. Since GANs do not map an image into a latent vector, an encoder attached to the generator of a GAN can achieve this mapping. A trained GAN maps two bona fide images into a latent domain for interpolation, and a decoder maps the interpolated vector into the image domain to realize morphed samples. In other words, if Z_1 represents the latent vector corresponding

to the first image, and Z_2 delineates the latent of the second image, the resulting morphed image in the latent domain can be formulated as the convex combination (alpha-blending) $Z_{morph} = \alpha Z_1 + (1-\alpha)Z_2$, where α and $1-\alpha$ are the coefficients delineating the contribution of the first and the second latent vector, respectively, to the final latent vector. Finally, a decoder maps the morphed latent vector into the spatial domain as the final morphed image.

In [7], morphed images are generated using a GAN which incorporates an encoder in its generator to model latent space. In addition, morphed images can be generated using StyleGAN [28,31,32] which adopts a style transfer method to synthesize an image given a reference style. In brief, the principal premise behind the StyleGAN is to make the statistics of deep feature maps consistent for both the image and the reference style. In [19, 33], a StyleGAN architecture is utilized to generate morphing attacks which are deemed highly photo realistic.

1.3.2 Morph Detection

Morph detection has been addressed under two scenarios. In the first scenario, called single image morph detection, a single image is classified as either bona fide or morphed [34, 35]. In the second scenario, called differential morph detection, auxiliary information which is a live version of a subject, is used to label an image as either bona fide or morphed [27,36–39]. The scope of this work is to design a single image morph detector. State-of-the-art methods on single image morph detection are summarized in Table 1.1.

Different methods have been proposed for morph detection [34, 35, 40–44], some of which are discussed here. Some morph detection techniques use hand-crafted features for training a classifier to identify morphed samples [7, 45–50]. In addition, deep embedding features extracted using off-the-shelf DNNs can be utilized for training a morph detector [7, 18, 35, 38, 43, 46, 51–54]. In [17], spectral behaviour of Photo Response Non-Uniformity (PRNU) is studied to detect morphed images. One of the research efforts [14] adopts Photo Response Non-Uniformity (PRNU) to distinguish between real and morphed images. Ferrara et al. [55] introduced face demorphing to reverse the morphing process to detect altered images.

Fusion of hand-crafted features extracted from color channels of HSV and YCbCr color spaces are studied in [56] as a method for detecting morphed images. One of the most important aspects of the morphing attacks is carefully selecting two bona fide subjects's face images such that the morphing attack looks highly photo realistic. In [57], the morph detection is investigated when the morphed images are generated using three different pairing protocols: (1) two similar images for morphing, (2) two random images, and (3) two dissimilar images. As a holistic approach for morph detection, fusion of the above-mentioned algorithms can be considered. In [46], two SVMs are trained using two different textures descriptors: LBPH, and BSIF. Another SVM is trained with the HOG, and deep embedding features are used to train another SVM. To integrate all approaches, the resulting scores from all the detectors are fused. The resulting noise artifact in the face morphing pipeline can be adopted for morph detection [58]. Another work [59] employs a denoised version of an image to find the residual noise of the image which can be utilized for identifying morphed samples. The paper aggregates several denoised versions of an image in the wavelet domain. Disentanglement of appearance and landmark is another method proposed for differential morph detection [27].

Interestingly, reflection inconsistencies are also employed to detect morphing attacks [16]. Pixel-wise supervision for morph detection was proposed in [60] to improve generalization of their morph detector. In [11], different modalities of a single image, such as eyes, nose, and mouth were used to improve morph detection accuracy. In addition, a multi-scale attention-based network was another method developed to detect morphed images, which use the attention mechanism for the images at different scales [61]. Moreover, feature-wise supervision was used in [62] to generate a prediction map for the single and differential morph detection. In [63], a GAN-based single image morph discriminator is trained, which leverages adversarial learning to detect single image morphing attacks.

1.3.3 Sparse Representation Learning

Sparse signal representation is an important class of representation learning methods which provides a compressed version of a high-dimensional signal [68]. Images are naturally sparse

Table 1.1: State-of-the-art methodologies on single image morph detection.

Publication	Venue	Methodology	Results
Towards generalized morphing attack detection by learning residuals [64]	Image and Vision Computing.	Learning morphing residuals using different encoder-decoder networks and color spaces.	The proposed approach has decreased the detection error rates at different thresholds compared to the baselines mentioned in the paper.
Single Image Face Morphing Attack Detection Using Ensemble of Features [65]	International Conference on Information Fusion.	Ensemble of features in different color spaces and high-frequency content extracted using Laplacian transform.	The proposed method decreased D-EER considerably to 5.99 and 6.34 for the datasets 1 and 2 introduced in the paper.
Towards making morphing attack detection robust using hybrid scale-space colour texture features [66]	International Conference on Identity, Security, and Behavior Analysis.	Different color space and scale space are adopted using Laplacian pyramid for feature extraction.	The framework substantially decreased the BPCER @ APCER=5% and BPCER @ APCER=10% to 7.59 and 0.86 compared to other deep and non-deep based approaches.
Accurate and robust neural networks for face morphing attack detection [67]	Journal of Information Security and Applications.	Data manipulation to limit the information provided to a deep network in order to force the a deep network to focus on different regions of an image.	The methodology increased the robustness against partial morphs form 20% to 87% and robustness against black-box attacks improved to 98% compared to naive training of 77%.

with respect to some predefined bases and that is why sparse representation learning is beneficial for image recognition tasks. More importantly, sparse representations have led to promising performance for face recognition tasks [69–71]. Structured group sparsity [21] has also proved to be compelling for learning representations which are more dsiscriminative. When features are arranged in a group setting, group Lasso [72], as one of the many feature selection methods, has shown impressive proficiency [21, 73–76]. When the L_1-norm of some structured parameters, such as grouped weights in a convolutional layer of a DNN, are added to the objective function in an optimization framework, which is known as Lasso regularization [77], sparsified grouped parameters leads to feature selection, which is known as structured group sparsity in a DNN [21]. Analogous to finding the principal components, learning a sparse representation limits degree of freedom when searching for an optimal hypothesis $\mathscr{H}$. Sparse representation learning has a close relationship with the Vapnik Chervonenkis (VC) dimension of a model and that is why learning the sparse representation leads to a better generalization based on statistical learning theory.

1.3.4 Attention Mechanism

Visual attention mechanisms [20,22,78] have introduced a paradigm shift for the mainstream computer vision tasks. Attention mechanism has been widely used for the visual recognition tasks such as image caption generation and visual question answering (VQA) [20, 79]. Two dominant categories of the attention mechanism are the soft deterministic attention and the hard stochastic attention [79]. The soft attention can either be adopted in a post-hoc manner, or it can be trained along with a DNN using back-propagation [20]. The hard attention mechanism is trained using a method called REINFORCE [80]. Attentive recurrent neural networks (RNNs) [81] are another variant of networks which exploit attention mechanism to amplify ROI and suppress background clutter.

In a typical attention mechanism, the correlation between each spatial location in a feature map of a deep network and the response of the network, which is usually the output of the last fully-connected layer in the network, designates the importance of the spatial location in terms of contributing to the final predicted output. Attention weights delineate the so-called importance of the spatial pixels. On the other hand, in the self-attention mechanism [82–84], the long-range dependencies between a pixel location in a feature map and all other pixel locations in the feature map are modeled irrespective of the output of the network.

The attention mechanism can guide a classifier into the most discriminative local patches of an input image where subtle anomalies in an image are captured. Morph detection can be thought of as a fine-grained classification because differences between a bona fide and morphed image are local and subtle, which is why the attention mechanism has proved to be useful for the task of morph detection [85]. The attention mechanism can be soft [20], which is a differentiable process trained using the back-propagation algorithm, or it can be hard, which adopts stochastic sampling to select the most discriminative pixels and is trained using the REINFORCE method [86].

Self-attention [82,87,88] has emerged as a powerful mechanism for boosting image recognition

performance. The self-attentional network can be implemented as a stand-alone framework without adopting any convolutional operations (e.g., vision transformers [88, 89]) for the downstream task of image recognition or object detection, which is not the scope of this study. On the other hand, several works have integrated the self-attentional modules into the convolutional layers of a DNN [23, 90, 91] as a feature augmentation method to capture long-range dependencies of features which are not revealed through the local convolution operation.

Chapter 2

Morph Detection Using Entropy Distributions Mismatch

2.1 Introduction

Morphing attack detection is of great significance in high-throughput border control applications. According to the CIA triad model, consisting of three main components, confidentiality, integrity, and availability of secure systems, morphed images violate the integrity of verification systems. A morphed image is generated using genuine face images from two different individuals. Because the resulting morphed image inherits characteristics of both subjects, it can be verified against both real subjects. Morphed images are generated using two approaches. In the first approach [5, 18, 53], two real face images are alpha blended in order to create a morphed image. To eliminate the ghosting effects in the morphed image, the average of the landmarks in both real images is used as the resulting landmark of the morphed image. In the second approach introduced in [7], a generative model, that is a Generative Adversarial Network (GAN), is trained to synthesize morphed images. Morph detection algorithms can be grouped into two main categories: single and differential morph detection. In the first category, an image under investigation is labeled as morphed or bona fide image, which is known as single image morph detection. In differential morph detection, a subject's image is compared with a live capture of the subject, and information from both images is used to detect morphed counterfeits.

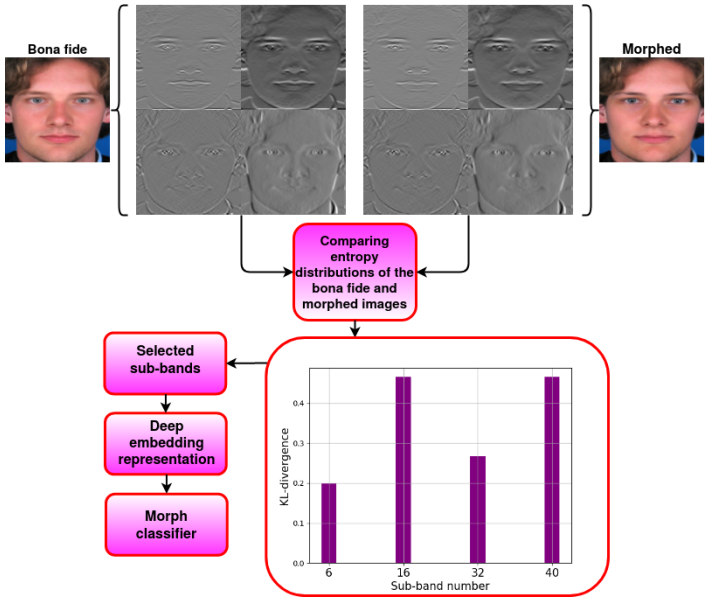


Figure 2.1: A bona fide and a morphed image along with the four corresponding wavelet sub-bands. Using all the bona fide and morphed images in the dataset, 48 pairs of entropy distributions are found for bona fide and morphed images. Given a sub-band, dissimilarity between the two entropy distributions represents how discriminative that sub-band is with respect to morph detection. In the figure, sub-bands 16 and 40 are more discriminative than 6 and 32. A deep classifier is trained using the selected informative sub-bands.

To detect morphed images, some of the previous research efforts employ hand-crafted features such as Binarized Statistical Image Features (BSIF) [92], Scale Invariant Feature Transform (SIFT) [93], Speeded Up Robust Features (SURF) [94], (Local Binary Patterns Histogram) LBPH [95], Fused Local Binary Pattern (FLBP), and Histogram of Gradianets (HOG). Recently, Deep Neural Networks (DNNs) have proved to be promising in detecting morphed images [47, 59]. Thus far, no wavelet-based morph detection algorithm has been proposed. In this work, we propose a single image morph detector which can distinguish between a bona fide and a morphed face image. To do so, we train a deep neural network with a small number of selected discriminative wavelet sub-bands that are chosen according to the following criterion: the relative entropy between the entropy distribution of real faces and morphed faces is found for each of the wavelet sub-bands. The higher the value of the relative entropy for a given sub-band, the more discriminative that sub-band is for the task of classification. Fig. 2.1 depicts our morph detection mechanism. Please note that only four wavelet sub-bands of a bona fide and its corresponding morphed image are selected in the figure for representation purpose. However, we consider all images in a given dataset

to find the histogram of entropy for each sub-band. Experiments on three datasets, i.e., VISAPP17 [5], MorGAN [7], and LMA [7] verifies the performance of our morph detector. Standard quantitative measures, set forth by ISO/IEC 30107-3 [96], are used to evaluate the effectiveness of our proposed method. The first measure is Attack Presentation Classification Error Rate (APCER), which is the percentage of morphed images that are classified as bona fide. The second measure is Bona Fide Presentation Classification Error Rate (BPCER), which represents the percentage of bona fide samples that are classified as morphed. If we label the morphed class as positive and the bona fide class as negative , APCER, and BPCER are equivalent to false negative rate and false positive rate, respectively. The contributions of this paper are as follows: the most discriminative wavelet sub-bands are selected based on the KL-divergence between the two entropy distributions of both real and morphed images for the wavelet sub-bands. A DNN is trained using the selected informative wavelet sub-bands to detect morphed images. Finally, an ablation study is performed to show the effectiveness of our sub-band selection scheme for tackling detecting morph attacks.

2.2 Our Framework

We employ undecimated 2D wavelet decomposition to address morphing attacks. Shannon entropy and Kullback-Liebler divergence [97] are utilized to identify the optimal discriminative sub-bands. In particular, the Shannon entropy [98] is used to measure embedded information in each sub-band of the wavelet decomposition. Since most of the morphing pipeline artifacts lie in the high frequency spectrum, we do not consider the Low-Low (LL) sub-band of the first level of decomposition to be decomposed further. Instead, the Low-High (LH), High-Low (HL), and High-High (HH) sub-bands are decomposed . After 3-level uniform decomposition, 48 sub-bands are obtained, for all of which the Shannon entropy is computed, and the distribution of the entropy is obtained for both real and morphed images for the three training datasets. The Kullback-Leibler divergence (relative entropy) is calculated between the entropy distribution of real and morphed sub-bands for each of the 48 sub-bands, and these 48 relative entropy values are sorted from highest to lowest. A final

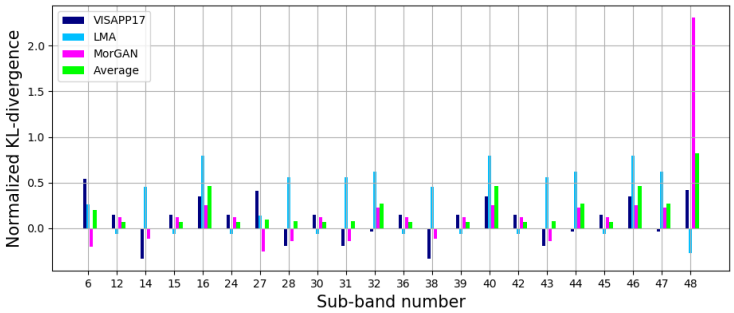


Figure 2.2: Zero-meaned KL-divergence values in the top 22 most discriminative wavelet sub-bands for three datasets: VISAPP17, LMA, and MorGAN. The zero-meaned average of the KL-divergence values in each sub-band, as related to the three datasets, is represented in green.

subset composed of 22 optimal discriminative sub-bands are selected that are used to train a DNN to detect morphed samples. As for the DNN, we employ a pre-trained Inception Resnet v1 architecture as our binary classifier.

2.2.1 Sub-band Selection Based on KL Divergence of Entropy Distributions

The pivotal point here is to distinguish morphed samples by leveraging the most discriminative sub-bands. To do so, we find the histograms of entropy of all 48 sub-bands for both bona fide and morphed images in the three datasets. Accordingly, 96 distributions are estimated using the histograms from the 48 sub-bands of both the bona fide and morphed presentations. The term $\hat{f}_{b_i}$ represents the estimated distribution for the ith sub-band pertinent to the bona fide images, and similarly, $\hat{f}_{m_i}$ represents the estimated distribution for the ith sub-band pertinent to the morphed images. The dissimilarity of the two probability distribution functions, namely $(\hat{f}_{b_i}, \hat{f}_{m_i})$ are calculated for all 48 sub-bands. The KL-divergence is the metric we employ to assess the dissimilarity between the distributions.

In order to select the most discriminative sub-bands, the KL-divergence values of each dataset are first normalized by removing the mean. The values are normalized to enable comparison of the distributions across the three datasets. Then, the zero-meaned values are averaged over

the three datasets for each sub-band. The higher the KL-divergence value for a single sub-band, the more informative and discriminative that sub-band is in terms of classification. By choosing the sub-bands that are based on the highest average KL-divergence values from all datasets instead of each dataset separately, we can find the sub-bands that are discriminative across the datasets, not just for a specific morphing technique. Fig. 2.2 shows the distribution of the zero-meaned KL-divergence values related to the 22 most discriminative wavelet sub-bands for the three morphed datasets, and their average values. Algorithm 1 illustrates our sub-band selection mechanism, in which $H(.)$ represents the entropy function.

Algorithm 1: Our Sub-band Selection

Input : Bona fide and Morphed Images
Output: A Set of Indices for Informative Sub-bands
$\mathbb{I} = \{\}$; `// index of sub-bands`
for $i = 1$ *to 48;* `// sub-bands`
do
  for $j = 1$ *to 3;* `// datasets`
  do
    $\hat{f}_{bij} \leftarrow distribution(H(S_{b_{ij}}))$
    $\hat{f}_{mij} \leftarrow distribution(H(S_{m_{ij}}))$
    $\mathbb{K}_{ij} \leftarrow D_{KL}(\hat{f}_{bij} \| \hat{f}_{mij})$
  end
end
for $i = 1$ *to 48* **do**
  $\bar{\mathbb{K}}_i \leftarrow \underset{j}{avg}(\mathbb{K}_{ij})$
  if $\bar{\mathbb{K}}_i > threshold$ **then**
    $\mathbb{I} \leftarrow i$
  end
end

It is worth mentioning that the threshold for selecting the informative sub-bands is chosen using a data-driven method. After sorting the KL-divergence values from highest to lowest, different subsets of sub-bands are selected, e.g., top-5, top-10, and so forth. Suppose that the top-5 values from the set of informative sub-bands are selected. A DNN having input channel size of five is trained on the three training datasets combined, coined the *universal dataset*. The performance of the corresponding DNN is reported through Area Under the Curve (AUC) metric using the validation portion of our universal dataset. Fig. 2.3 depicts the Area Under the Curve (AUC) when different numbers of sub-bands are chosen, based on which the optimal point for number of sub-bands is chosen as 22. The performance of

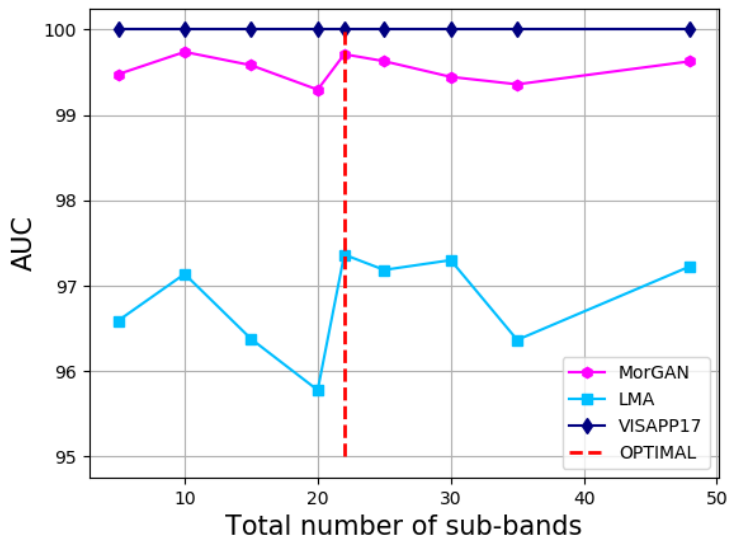


Figure 2.3: Area under the curve versus number of sub-bands used in the training. Results indicate that 22 is the optimal number of sub-bands. After selecting 22 sub-bands, the performance does not increase significantly enough to validate using more sub-bands.

the VISAPP17 dataset is consistent, irrelevant of the number of sub-bands used. This is primarily due to the small size of the VISAPP17 dataset (only 314 images–183 morphed and 131 real), and our DNN easily fits to VISAPP17 dataset regardless of the number of the selected sub-bands.

2.3 Experimental Setup

2.3.1 Datasets

Datasets used in this work are the VISAPP17 [5], MorGAN [7], and LMA [7]. The VISAPP17 dataset has been created using a landmark-based morphing attack, following by splicing, in which corresponding landmarks in two bona fide subjects are detected and the mean of each pair of the landmarks is calculated. Landmarks of each subject are then warped into the averaged landmark position, and the morphed image is generated using the blending of the

two subjects' samples using triangulation [99] and then spliced into one of the contributing images. This technique aims to avoid artifacts that commonly arise from landmark manipulation, such as those that occur around the hairline. The MorGAN dataset is generated using a GAN. The encoder in a GAN can transform images to a latent space, and when two latent spaces related to two different subjects are combined, a morphed subject is synthesized. The LMA dataset is also generated using the landmark manipulation in two subjects' face images.

2.3.2 Training Setup

In this work, the Inception-ResNet-v1 architecture [8] is adopted as our DNN, which integrates the residual skips introduced in [100], and a revised version of Inception architecture [101]. We fine-tune an Inception-ResNet-v1, already pretrained on VGGFace2 [102]. The DNN is additionally fine-tuned with the obtained 22 discriminative wavelet sub-bands of the VISAPP17, MorGAN, and LMA datasets. An Adam optimizer [103] is employed for updating parameters of our network, and two 12 GB TITAN X (Pascal) GPUs accelerate our training.

2.3.3 Training/Testing Using Selected Sub-bands

In order to find the optimal number of sub-bands, we combine the three morph image datasets into a *universal dataset*. From this universal dataset, the training set consists of 1631 bona fide, and 1183 morphed samples. The validation set consists of 462 bona fide, and 167 morphed subjects. Moreover, the test set includes 1631 bona fide, and 1183 morphed images. We train several Inception-ResNet-v1 networks using the training portion of the universal dataset for a different number of chosen wavelet sub-bands. We assess the performance of the trained networks using the validation portion of the universal dataset. In other words, we do a search over the number of wavelet sub-bands, which is the input channel

Table 2.1: Performance of single morph detection: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Train	Test	Algorithm	D-EER	5%	10%
VISAPP17	VISAPP17	BSIF+SVM [92]	16.51	35.61	26.79
		SIFT+SVM [93]	38.59	82.40	75.60
		LBP+SVM [95]	38.00	77.10	67.90
		SURF+SVM [94]	30.45	84.70	69.40
		Ours	**0.00**	**0.00**	**0.00**
	LMA	BSIF+SVM [92]	54.00	93.31	88.95
		SIFT+SVM [93]	37.00	79.00	70.00
		LBP+SVM [95]	33.00	71.80	59.90
		SURF+SVM [94]	39.30	86.10	75.70
		Ours	**31.86**	**83.80**	**71.21**
	MorGAN	BSIF+SVM [92]	54.80	92.32	88.87
		SIFT+SVM [93]	58.00	96.10	89.90
		LBP+SVM [95]	40.00	76.90	67.40
		SURF+SVM [94]	40.30	83.00	74.00
		Ours	**41.00**	**93.60**	**85.00**
LMA	VISAPP17	BSIF+SVM [92]	51.19	83.65	75.00
		SIFT+SVM [93]	38.00	90.80	86.30
		LBP+SVM [95]	36.60	77.80	71.80
		SURF+SVM [94]	30.80	70.00	65.60
		Ours	**68.80**	**100.00**	**98.90**
	LMA	BSIF+SVM [92]	33.05	78.34	62.86
		SIFT+SVM [93]	33.30	83.40	72.00
		LBP+SVM [95]	28.00	58.60	51.40
		SURF+SVM [94]	37.40	79.50	70.00
		Ours	**8.80**	**14.90**	**7.91**
	MorGAN	BSIF+SVM [92]	42.01	89.77	79.19
		SIFT+SVM [93]	50.70	95.00	89.80
		LBP+SVM [95]	35.00	72.60	61.30
		SURF+SVM [94]	41.27	84.60	78.00
		Ours	**32.22**	**76.22**	**62.50**
MorGAN	VISAPP17	BSIF+SVM [92]	63.00	100.00	100.00
		SIFT+SVM [93]	42.00	92.40	84.00
		LBP+SVM [95]	42.32	84.70	79.30
		SURF+SVM [94]	31.40	74.00	55.70
		Ours	**2.20**	**0.59**	**0.00**
	LMA	BSIF+SVM [92]	53.00	95.25	92.46
		SIFT+SVM [93]	40.20	90.70	80.00
		LBP+SVM [95]	39.18	75.90	67.7
		SURF+SVM [94]	39.40	81.00	71.60
		Ours	**39.11**	**89.55**	**80.25**
	MorGAN	BSIF+SVM [92]	1.57	1.42	1.30
		SIFT+SVM [93]	43.50	93.20	84.20
		LBP+SVM [95]	20.10	52.70	32.30
		SURF+SVM [94]	39.95	80.00	72.60
		Ours	**0.00**	**0.00**	**0.00**

Table 2.2: DET curves when our morph detector is trained and tested on the selected 22-sub-band datasets. The legend represents train-test datasets.

Train	Test	Algorithm	D-EER	5%	10%
Universal(VISAPP17+LMA+MorGAN)	VISAPP17	BSIF+SVM [92]	35.00	67.20	59.00
		SIFT+SVM [93]	27.00	83.20	70.90
		LBP+SVM [95]	37.67	72.50	59.50
		SURF+SVM [94]	31.00	79.40	70.10
		Ours	**0.00**	**0.00**	**0.00**
	LMA	BSIF+SVM [92]	30.00	70.42	57.60
		SIFT+SVM [93]	28.31	67.70	50.00
		LBP+SVM [95]	29.00	61.50	51.20
		SURF+SVM [94]	33.40	74.50	62.70
		Ours	**8.61**	**12.93**	**7.05**
	MorGAN	BSIF+SVM [92]	28.80	62.42	45.70
		SIFT+SVM [93]	47.60	92.30	88.60
		LBP+SVM [95]	31.20	62.00	55.60
		SURF+SVM [94]	38.67	76.00	70.00
		Ours	**3.10**	**2.04**	**3.89**
	Universal	BSIF+SVM [92]	23.74	51.42	38.67
		SIFT+SVM [93]	37.21	87.45	76.71
		LBP+SVM [95]	38.80	91.36	83.40
		SURF+SVM [94]	36.00	75.50	65.76
		Ours	**5.45**	**5.70**	**3.19**

size of our convolutional neural network. Please note that the wavelet sub-bands are already sorted based on the corresponding KL-divergence values from highest to lowest. According to the sub-band selection scheme mentioned in section 2.2.1 and Fig. 2.3, which shows the performance of the trained classifier using different number of wavelet sub-bands, the optimal number of informative sub-bands is 22; thus, our final DNN has 22 input channels consisting of the top 22 most discriminative sub-bands.

The performance of our morph detector, and the baseline methods for comparison are summarized in Table 2.1. Please note that we have considered all the possible training/testing scenarios using the three datasets, i.e., the VISAPP17, LMA, and MorGAN. The corresponding Detection Error Trade-off (DET) curves are displayed in Fig. 2.4. In addition, we have trained the morph classifier using the training portion of the universal dataset, and the performance of that network is also evaluated using the testing portion of each individual dataset, as well as the universal dataset. The results of the training using the universal dataset, and the corresponding baseline methods are provided in Table 2.2. Moreover, the related DET curves are shown in Fig. 2.5.

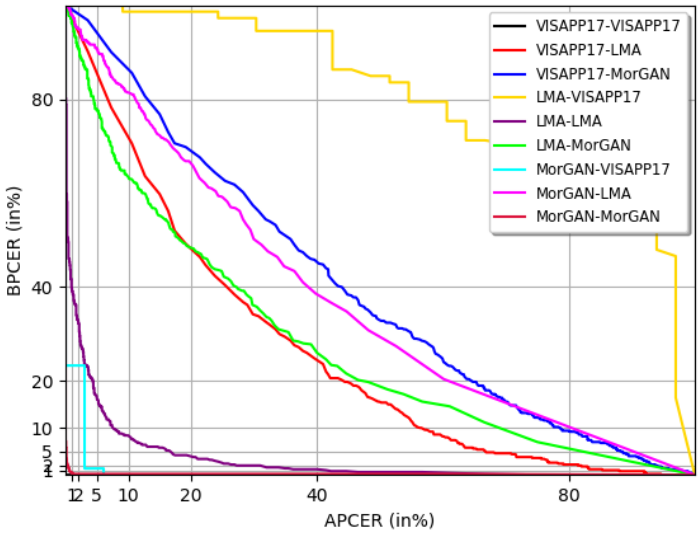


Figure 2.4: the left dataset designates the training dataset, and the right one represents the testing dataset. For example, MorGAN-LMA is for the case when the training set comes from the MorGAN dataset, and the testing set comes from the LMA dataset.

D-EER represents Detection Equal Error Rate, where APCER equals BPCER. BPCER5 designates BPCER rate for APCER=5%, and BPCER10 designates BPCER rate for APCER=10%. A close scrutiny of the DET curves in Fig. 2.4 reveals that our morph detector can accurately detect morphed samples in both the VISAPP17, and MorGAN datasets when both training and testing data originate from the same dataset. Fig 2.5. also shows that our morph detector is able to detect most of the morphed samples in the VISAPP17, and MorGAN datasets when the classifier is trained on the training portion of the universal dataset.

2.3.4 Class Activation Maps

Class activation maps, set forth in [104], show the extent to which different regions in a given image contribute to the final classification decision for every class in an already trained DNN. After training an Inception-ResNet-v1 morph detector, class activation maps were constructed using the feature embeddings from the last layer before fully connected and softmax layers. The results are interestingly indicative of the likelihood that an image will be

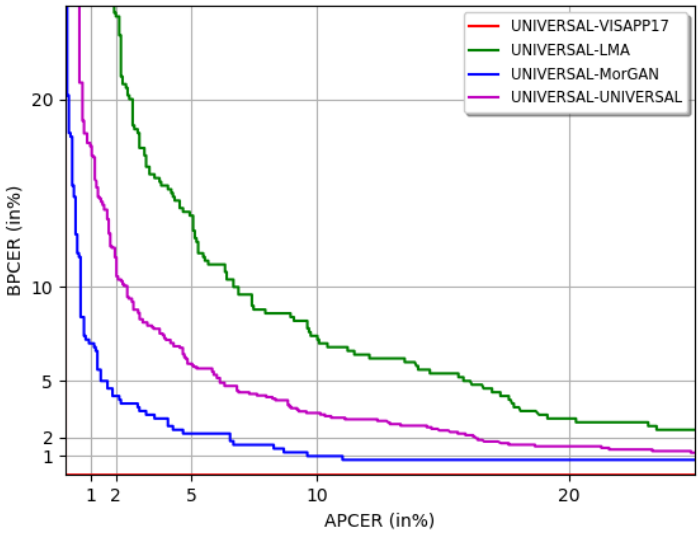


Figure 2.5: DET curves when our morph detector is trained on the selected 22-sub-band universal datasets.

Figure 2.6: Class activation maps. Left: bona fide subject 1, middle: morphed subject, right: bona fide subject 2.

classified as morphed or bona fide. For example, in Fig. 2.6, the middle image, representing a morphed one, has many more affected areas than the other two bona fide images. This is an indicator that the middle image is the most likely image among the three images to be classified as morphed. Given that, our trained DNN using 22 discriminative wavelet sub-bands is effectively distinguishing morphed images from the non-morphed images.

2.4 Ablation Study

In this section, the effect of sub-band selection is examined. To prove the effectiveness of band selection, a visualization method, namely t-SNE [105], is adopted. A total of 200 morphed, and 200 bona fide images are selected from the test set of MorGAN dataset. Fig. 2.7 shows the t-SNE visualizations for three scenarios using the MorGAN dataset, the first of which visualizes the original images, which is shown in the leftmost column. In the middle column, the 48 selected sub-band data is plotted. Finally, the 22 selected sub-band data is shown in the rightmost column. It is evident in Fig. 2.7 that sub-band selection contributes considerably to concentrating the morphed and bona fide data into separable clusters, which is highly desirable in terms of detecting morphed imagery.

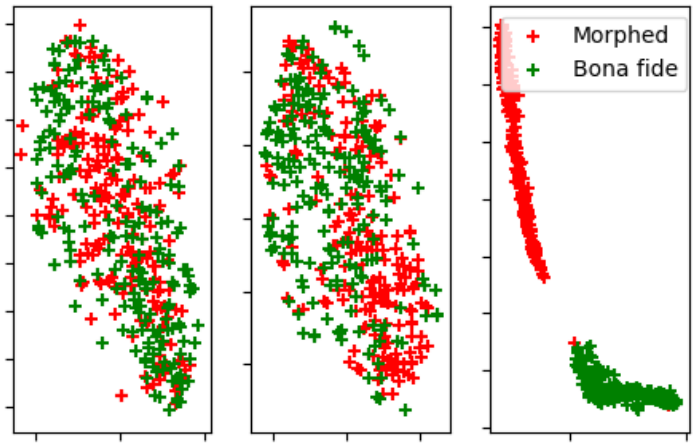


Figure 2.7: T-SNE visualization for the original images (left), 48 sub-band data (middle), and 22 sub-band data (right). The 22 sub-bands evidently separate the morph and bona fide classes into very distinct clusters.

2.5 Conclusion

We proposed a framework to detect morphed face images using undecimated 2D-DWT. To select the optimal and informative bands, we found the distribution of the entropy for all the 48 wavelet sub-bands considering both the bona fide, and morphed images. The KL-divergence between the given distributions, integrated in a data-driven approach, led us to select the 22 most discriminative sub-bands. Furthermore, a close look at the presented results in Tables 2.1 & 2.2 highlights the fact that our morph classifier can identify morphed samples with a high accuracy in both the VISAPP17, and MorGAN datasets. Moreover, the ablation study on the sub-band selection substantiates the effectiveness of our method and shows that our trained DNN can map data samples to a new space where two bona fide and morphed classes are aggregated into two well-separated clusters.

Chapter 3

Attention Aware Detection of Morphed Face Images

3.1 Introduction

Robust, reliable verification systems are the crucial backbones of biometric document authentication protocols, that are to operate flawlessly. Although image morphing is not a new paradigm, it was first identified as a security concern by Ferrara et al. [13], who explained how a criminal can dodge a border control checkpoint using a travel document that was issued with a morphed image. The goal of the face image morphing attack is to synthesize a forged imaged from two composing original images such that the artificially crafted morphed image can be verified against the two original images not only visually, but also in the feature space by a classifier [40]. Moreover, morphed samples can be labeled as hard positive samples in comparison to negative genuine samples because morphed samples are synthesized to intentionally lie on the negative samples' manifold. Similar to adversarially perturbed data samples that fool classification networks into a wrong predicted class [106, 107], morphed images are crafted to lead a verifier into a false acceptance.

Detecting morphed images has garnered a great deal of attention from the biometrics research community because of its crucial impact on the security protocols [108], especially those used for authenticating travel documents. The vast majority of research efforts has dealt with morphing attacks through either using hand-crafted texture features to find a discriminative

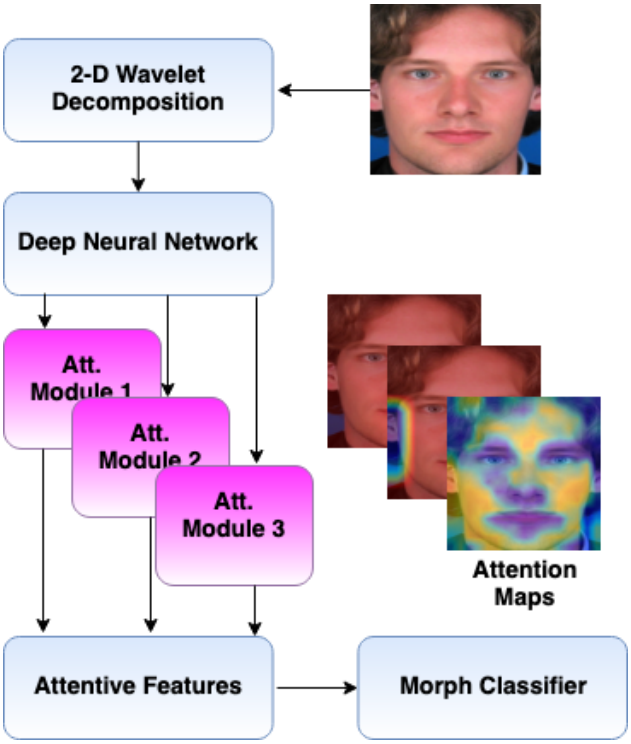


Figure 3.1: Our Proposed Deep Attention-based Morph Detector. Wavelet sub-bands of the input image is fed into our DNN during training phase of the network. Three Attention modules, i.e., Att. modules, generate the new attention weighted features, i.e., attentive features, as well as the attention maps. Attentive features are used to detect morphed images. Please note that the attention maps are generated after training the DNN.

hyperplane between the positive (morphed), and negative (genuine) samples [14–17], or harvesting those features for learning a deep classifier [46, 47]. Recently, the visual attention mechanism has taken computer vision community with storm. First introduced in [81], the visual attention mechanism has emerged as a powerful by-product of DNNs, which can boost visual recognition performance on a variety of datasets considerably [20, 79, 109–111].

In this paper, we present an attention-based DNN in the wavelet domain for detecting morphed samples. To the best of our knowledge, this is the first work which incorporates attention mechanism into a deep morph detector. Our proposed network employs attention to focus on Regions of Interest (ROI) in terms of morph detection, that are specifically landmarks around the eyes and hairline in the landmark-based facial image morphing attacks.

Wavelet sub-bands of an image represent information with different time-frequency granu-

larity that are adapted to our DNN as input. The soft attention mechanism used in a given layer of our DNN retains spatial regions in the layer's resulting feature maps that represent the discriminative regions, and discard those pixels that are outside the discriminative regions. Fig. 3.1 shows an overview of our proposed deep attention-based morph detector. We utilize wavelet sub-bands instead of the raw images since we can easily discard frequency contents, sub-bands, which are not discriminative for morph detection such as the low-low (LL) sub-bands. Most importantly, we validate performance of our method through extensive experiments on the three morph datastes: VISAPP17 [5], LMA [7], and MorGAN [7]. Moreover, estimated attention maps are obtained for both real and morphed images. The contribution of this work are as follows:

- Incorporating an end-to-end trainable soft attention mechanism into deep morph detector network.
- Tailoring wavelet sub-bands for our deep attention-based morph detector.
- Training our deep attention-based network using the three datasets, as well as a combination of all the three datasets, which is coined "universal" dataset.

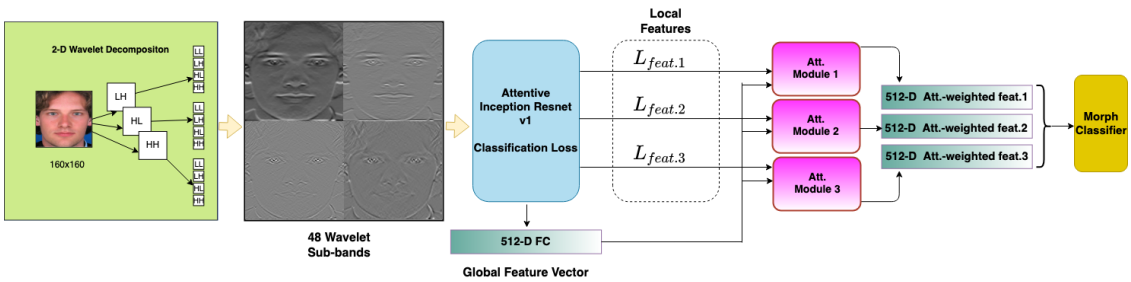


Figure 3.2: Our deep attention-based morph detector. The input images are initially decomposed into 48 uniform wavelet sub-bands, which are fed into our morph detector. Attention modules are placed at three convolutional layers, namely $\mathscr{L}_1$, $\mathscr{L}_2$, and $\mathscr{L}_3$. The $L_{feat.1}$, $L_{feat.2}$, and $L_{feat.3}$ represent the local features vectors of layers $\mathscr{L}_1$, $\mathscr{L}_2$, and $\mathscr{L}_3$, respectively. The 512-D attention weighted local features in a layer, shown by 512-D Att. weighted feat., are obtained using the local features of the layer, and the 512-D FC global feature vector. The three resulting attention weighted features are concatenated to form our new attended features.

3.2 Our Framework

Our attention-based morph detector is displayed in Fig. 3.2. Based on Fig. 3.2, the input images are initially decomposed into 48 uniform wavelet sub-bands that are further stacked channel-wise and then passed to our morph detector. Our morph detector leverages three attention modules at three different convolutional layers, denoted by $\mathscr{L}_1$, $\mathscr{L}_2$, and $\mathscr{L}_3$. The local feature vectors resulting from the three convolutional layers $\mathscr{L}_1$, $\mathscr{L}_2$, and $\mathscr{L}_3$ are denoted by $L_{feat.1}$, $L_{feat.2}$, and $L_{feat.3}$, respectively. The attention weighted local features for a given convolutional layer are obtained using the layer's local features, and the global feature vector resulting from the 512-D fully connected (FC) layer in our network, e.g., the first 512-D attention weighted local features in the $\mathscr{L}_1$ layer, shown by 512-D Att. weighted feat.1, are obtained using the local features of $\mathscr{L}_1$, that is to say $L_{feat.1}$, and the 512-D FC global feature vector. The three resulting attention weighted features are concatenated and passed into a new FC layer with 512×3 neurons.

3.2.1 Uniform Wavelet Decomposition

Most artifacts due to facial image morphing techniques lie within the high frequency spectrum, and using wavelet decomposition allows us to cherry-pick the desired wavelet sub-bands by discarding the low-frequency sub-bands. Therefore, using specific wavelet sub-bands instead of the original image is highly justified in our study. We apply three-level undecimated 2-D wavelet decomposition on both bona fide and morphed images. Analyzing the wavelet sub-bands of a bona fide and its corresponding morphed image justifies considering the high frequency spectra for the task of morph detection. In other words, we discard the low-low (LL) wavelet sub-band after first level of decomposition, and we keep the low-high (LH), high-low (HL), high-high (HH) for the second and third levels of decomposition. In total, 48 wavelet sub-bands are stacked channel-wise, which are utilized as the input to our attention-based morph detector. Decomposing an RGB image into 48 wavelet sub-bands leads to decoupled spectra, focusing on the frequency contents that are discriminative in terms of

distinguishing between bona fide and morphed images.

3.2.2 Integrating Attention-Weighted Features

To distinguish between bona fide and morphed images, we adopt the end-to-end trainable soft attention mechanism introduced in [20]. This soft attention mechanism is differentiable with respect to the network parameters. We show that our attention-based network can meticulously focus on the regions that contribute the most to detecting morphed images. We insert three attention modules at three different convolutional layers $\mathscr{L}_1$, $\mathscr{L}_2$, and $\mathscr{L}_3$ in our DNN. Therefore, as presented in Fig. 3.2, instead of a single global feature vector, that is the 512-D fully connected layer (FC) output, we concatenate the three attention-weighted local feature vectors at three different convolutional layers to accomplish the classification task. These attention maps at each convolutional layer reveals the importance of each spatial location in the layers' feature maps.

Suppose that a spatial local feature vector in the location $i \in \{1,2,..,n\}$ in the convolutional layer $\mathscr{L}_k$, $1 \leq k \leq 3$, is shown by $\boldsymbol{\ell}_i^{\mathscr{L}_k}$. As presented in Fig. 3.2, $L_{feat.k} = \{\boldsymbol{\ell}_1^{\mathscr{L}_k}, \boldsymbol{\ell}_2^{\mathscr{L}_k}, ..., \boldsymbol{\ell}_n^{\mathscr{L}_k}\}$. The compatibility score for each spatial location, i, represents the importance of that pixel for detecting morphed images. The compatibility score for local feature vector $\boldsymbol{\ell}_i^{\mathscr{L}_k}$ is given as:

$$c_i^{\mathscr{L}_k} = \langle \boldsymbol{\ell}_i^{\mathscr{L}_k}, \boldsymbol{g} \rangle, i \in \{1,2,..,n\}, \tag{3.1}$$

where $\boldsymbol{g}$ designates the global feature vector, that is the 512-D output of the fully connected layer and $\langle .,. \rangle$ represents the inner product. We further normalize the computed compatibility scores in a given convolutional layer $\mathscr{L}_k$ using the softmax normalization, which is given as:

$$a_i^{\mathscr{L}_k} = \frac{\exp(c_i^{\mathscr{L}_k})}{\Sigma_{i=1}^{i=n} \exp(c_i^{\mathscr{L}_k})}, i \in \{1,2,..,n\}. \tag{3.2}$$

A linear combination of the local feature vectors $\boldsymbol{\ell}_i^{\mathscr{L}_k}$ and the attention weights $a_i^{\mathscr{L}_k}$ yields the attentive local descriptor for the given convolutional layer $\mathscr{L}_k$. The global feature vector,

Table 3.1: Performance of single morph detection: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Dataset	Algorithm	D-EER	5%	10%
VISAPP17	BSIF+SVM [92]	16.51	35.61	26.79
	SIFT+SVM [93]	38.59	82.40	75.60
	LBP+SVM [95]	38.00	77.10	67.90
	SURF+SVM [94]	30.45	84.70	69.40
	RGB+DNN [8]	1.76	0.588	0.58
	Ours	**0.00**	**0.00**	**0.00**
LMA	BSIF+SVM [92]	33.05	78.34	62.86
	SIFT+SVM [93]	33.30	83.40	72.00
	LBP+SVM [95]	28.00	58.60	51.40
	SURF+SVM [94]	37.40	79.50	70.00
	RGB+DNN [8]	9.10	15.18	7.49
	Ours	**8.71**	**17.86**	**6.52**
MorGAN	BSIF+SVM [92]	1.57	1.42	1.30
	SIFT+SVM [93]	43.50	93.20	84.20
	LBP+SVM [95]	20.10	52.70	32.30
	SURF+SVM [94]	39.95	80.00	72.60
	RGB+DNN [8]	2.44	1.88	1.50
	Ours	**0.00**	**0.00**	**0.00**

i.e., attention-weighted feature vector, can be written as:

$$\boldsymbol{g}_a^{\mathscr{L}_k} = \Sigma_{i=1}^{i=n} a_i^{\mathscr{L}_k} \boldsymbol{\ell}_i^{\mathscr{L}_k}. \tag{3.3}$$

We concatenate the estimated attention weighted local features at three different convolutional layers which are fed into a FC layer having size of 512×3 followed by a 2-neuron FC layer, which generates the binary logits for detecting morphed images.

3.3 Experimental Setup

3.3.1 Datasets

In this study, three different morphed image datasets are used that are, the VISAPP17 [5], LMA [7], and MorGAN [7]. The VISAPP17 dataset is generated using landmark-based face morphing attack, followed by splicing. In the landmark-based morphing pipeline locations of the corresponding landmarks in two bona fide subjects are averaged, and facial regions are

Table 3.2: Performance of single morph detection: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Train	Test	Algorithm	D-EER	5%	10%
Universal(VISAPP17+LMA+MorGAN)	VISAPP17	BSIF+SVM [92]	35.00	67.20	59.00
		SIFT+SVM [93]	27.00	83.20	70.90
		LBP+SVM [95]	37.67	72.50	59.50
		SURF+SVM [94]	31.00	79.40	70.10
		RGB+DNN [8]	0.00	0.00	0.00
		Ours	**0.00**	**0.00**	**0.00**
	LMA	BSIF+SVM [92]	30.00	70.42	57.60
		SIFT+SVM [93]	28.31	67.70	50.00
		LBP+SVM [95]	29.00	61.50	51.20
		SURF+SVM [94]	33.40	74.50	62.70
		RGB+DNN [8]	7.80	13.00	6.10
		Ours	**8.11**	**14.21**	**6.83**
	MorGAN	BSIF+SVM [92]	28.80	62.42	45.70
		SIFT+SVM [93]	47.60	92.30	88.60
		LBP+SVM [95]	31.20	62.00	55.60
		SURF+SVM [94]	38.67	76.00	70.00
		RGB+DNN [8]	4.69	4.70	2.74
		Ours	**2.59**	**1.50**	**0.89**
	Universal	BSIF+SVM [92]	23.74	51.42	38.67
		SIFT+SVM [93]	37.21	87.45	76.71
		LBP+SVM [95]	38.80	91.36	83.40
		SURF+SVM [94]	36.00	75.50	65.76
		RGB+DNN [8]	5.57	6.08	3.00
		Ours	**6.42**	**7.58**	**3.46**

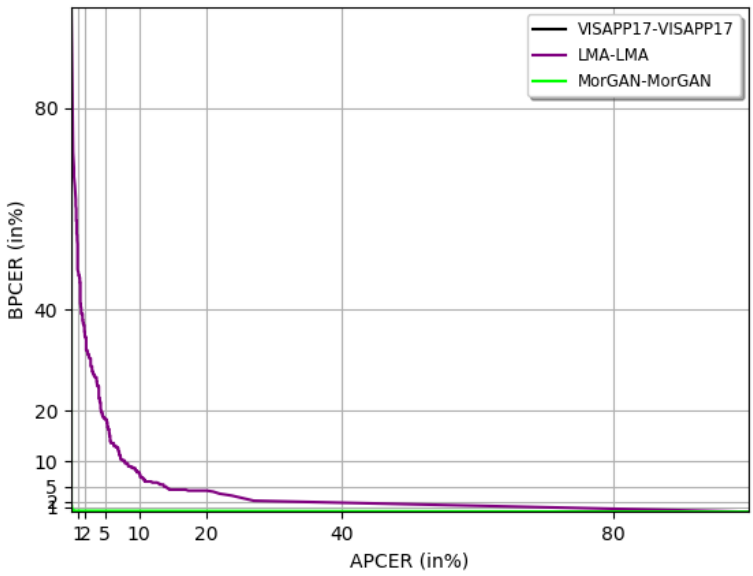


Figure 3.3: DET curves when our attention-based morph detector is trained using the individual datasets.

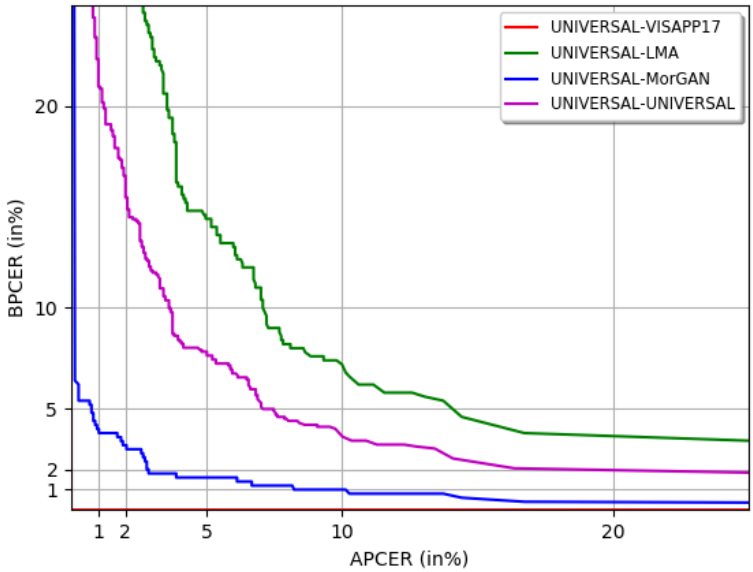


Figure 3.4: DET curves when our attention-based morph detector is trained using the training portion of the universal dataset.

divided using Delaunay triangulation before their alpha blending. LMA is a landmark-based morphed image dataset, and MorGAN dataset is generated using a generative model, GAN to be exact. Contrary to the landmark-based morphing attack, which captures geometry of underlying bona fide images, GAN-based morphing attacks synthesize morphed images after capturing the underlying distributions of bona fide facial images.

MTCNN [112] is utilized for face detection and alignment. Face images are resized to 160×160 pixels. For each dataset, 50% of the subjects are considered for training while the other 50% are used for the test set. In addition, 15% of the training set is selected during model optimization as the validation set. The train-test split is disjoint, with no overlapping bona fides, morphs, or bona fides contributing to morphs. In addition to the individual datasets, we combine the three datasets into a *universal dataset*. Regarding the universal dataset, the training set includes 1631 bona fide, and 1183 morphed samples. The validation set contains 462 bona fide, and 167 morphed subjects. In addition, the test set is composed of 1631 bona fide, and 1183 morphed images.

Table 3.3: Performance of single morph detection for the different number of attention-modules: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Dataset	Att. Layers	D-EER	5%	10%
VISAPP17	$\mathscr{L}_3$	00.00	00.00	00.00
	$\mathscr{L}_2$+$\mathscr{L}_3$	00.00	00.00	00.00
	$\mathscr{L}_1$+$\mathscr{L}_2$+$\mathscr{L}_3$	00.00	00.00	00.00
LMA	$\mathscr{L}_3$	12.45	21.23	15.18
	$\mathscr{L}_2$+$\mathscr{L}_3$	12.12	23.58	17.21
	$\mathscr{L}_1$+$\mathscr{L}_2$+$\mathscr{L}_3$	8.71	17.86	6.52
MorGAN	$\mathscr{L}_3$	00.00	00.00	00.00
	$\mathscr{L}_2$+$\mathscr{L}_3$	00.00	00.00	00.00
	$\mathscr{L}_1$+$\mathscr{L}_2$+$\mathscr{L}_3$	00.00	00.00	00.00

Table 3.4: Performance of the universal training set single morph detection for different number of attention-modules: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Train	Test	Att. Layers	D-EER	5%	10%
Universal	VISAPP17	$\mathscr{L}_3$	00.00	00.00	00.00
		$\mathscr{L}_2$+$\mathscr{L}_3$	00.00	00.00	00.00
		$\mathscr{L}_1$+$\mathscr{L}_2$+$\mathscr{L}_3$	00.00	00.00	00.00
	LMA	$\mathscr{L}_3$	14.37	27.23	16.54
		$\mathscr{L}_2$+$\mathscr{L}_3$	13.24	35.36	18.61
		$\mathscr{L}_1$+$\mathscr{L}_2$+$\mathscr{L}_3$	8.11	14.21	6.83
	MorGAN	$\mathscr{L}_3$	7.21	6.31	5.02
		$\mathscr{L}_2$+$\mathscr{L}_3$	7.14	7.86	4.91
		$\mathscr{L}_1$+$\mathscr{L}_2$+$\mathscr{L}_3$	2.59	1.50	0.89
	Universal	$\mathscr{L}_3$	8.91	12.21	8.27
		$\mathscr{L}_2$+$\mathscr{L}_3$	9.95	12.23	8.93
		$\mathscr{L}_1$+$\mathscr{L}_2$+$\mathscr{L}_3$	6.42	7.58	3.46

3.3.2 Training Setup

For the backbone of our attention-based morph detector, we employ Inception-ResNet-v1 [8] , which harnesses the residual skips [100], as well as the revised version of the Inception network [101]. We add three attention modules to the network at $\mathscr{L}_1$ = *"conv2d_4b"*, $\mathscr{L}_2$ = *"mixed_6a"*, and $\mathscr{L}_3$ = *"mixed_7a"*. Since the number of channels in the resulting feature vectors related to the three convolutional layers are not 512-D, we project the feature vectors to new vectors where number of channels are 512. The projection in each convolutional layer is achieved using the 1×1 convolutional filters, where 512 kernels with the size of 1×1 are employed. The Adam optimizer updates the weights of our DNN accelerated using two 12 GB TITAN X (Pascal) GPUs. Batch size of 8 is considered for training.

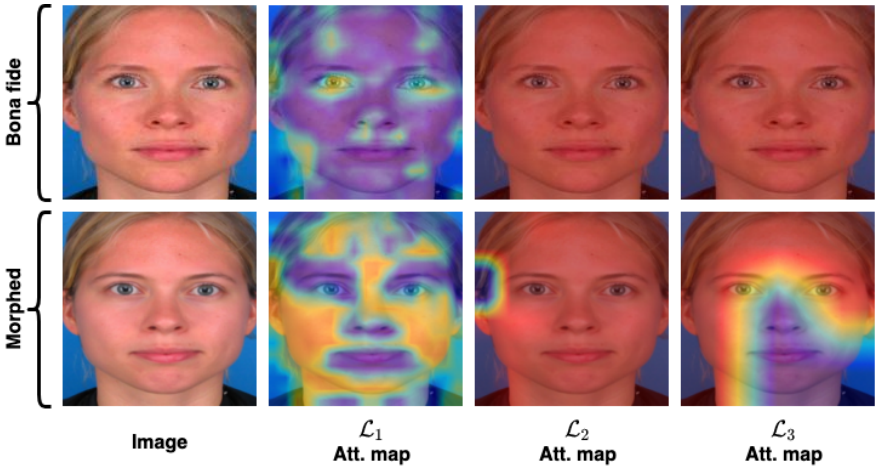


Figure 3.5: Estimated attention maps for a bona fide and the corresponding morphed image obtained from the three attention modules.

3.3.3 Performance of the Attention-based Morph Detector

Standard quantitative measures are used to evaluate the effectiveness of our proposed method. The first measure is Attack Presentation Classification Error Rate (APCER), which is the percentage of morphed images that are classified as bona fide. The second measure is Bona Fide Presentation Classification Error Rate (BPCER), which represents the percentage of bona fide samples that are classified as morphed. If we label the morphed class as positive and the bona fide class as negative , APCER, and BPCER are equivalent to false negative rate and false positive rate, respectively. Detection error trade-off (DET) curves represent performance of our attention-based DNN. D-EER stands for the Detection Equal Error Rate, where APCER equals BPCER. BPCER5 represents BPCER rate for APCER=5%, and BPCER10 represents BPCER rate for APCER=10%.

We train our attention-based DNN using the three datasets, that are the VISAPP17, LMA, and MorGAN. Table 3.1 delineates the performance of the baseline methods, as well as our attention-based morph detector for the three datasets. In addition, Fig. 3.3 depicts the detection error trade-off (DET) curves for the three datasets.

Moreover, we scrutinize the scenario where all three datasets are combined, which was coined the universal dataset. Therefore, we train our network using the training portion of the universal dataset, and test set comes from all individual datasets, as well as the testing portion

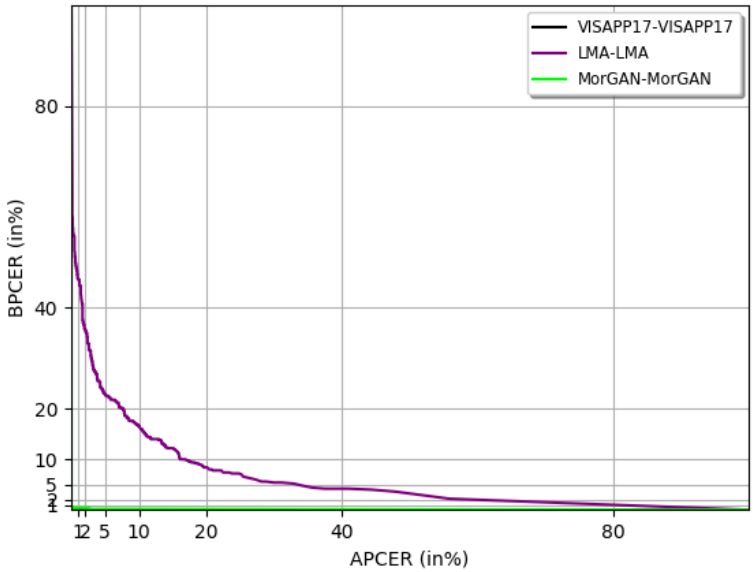


Figure 3.6: DET curves for the individual datasets for two attention modules.

of the universal dataset. The performance of our attention-based morph detector when trained on the universal dataset is summarised in Table 3.2, and Fig. 3.4 depicts the DET curves when the attention-based DNN is trained using the universal dataset. Our attention-based morph detector can detect morphed samples in the VISAPP17 and MorGAN datasets accurately when the network is trained on each dataset.

3.3.4 Estimated Attention Maps

The estimated attention maps, resulting from the three attentions modules are shown in Fig. 3.5. It is worth mentioning that the heatmaps of the resulting attention maps are applied to the image for visualization purpose. The first row, shows the bona fide image, and its corresponding attention maps from the three different convolutional layers. The second row is related to the attention maps of the morphed image. Comparing the attention map of the $\mathscr{L}_1$ for the bona fide image with that of the morphed image reveals that the morphed images has more attended areas that is caused by the morphing attack pipeline, which comprises landmark manipulation for this image, coming from the VISAPP17 dataset. Given the

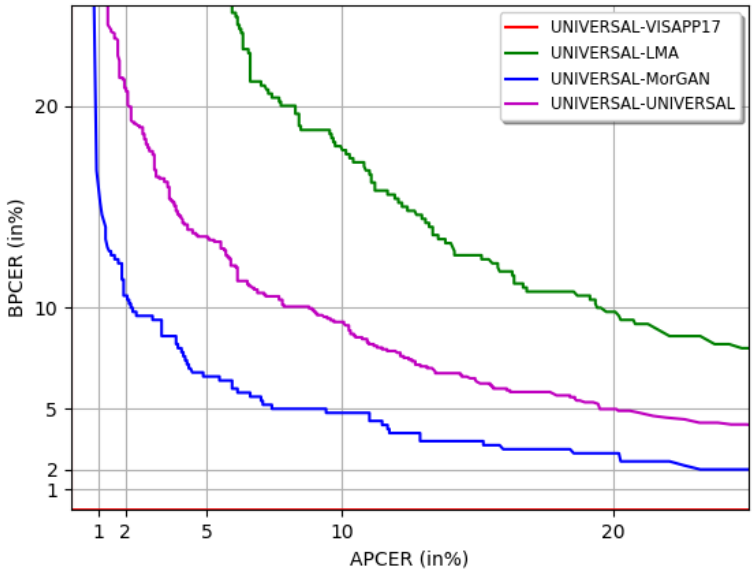


Figure 3.7: DET curves for the universal datasets for two attention modules.

attention maps of the $\mathscr{L}_2$ and $\mathscr{L}_3$, there are salient impacted regions in the feature maps of the morphed images, while there is no obvious attentive regions in the bona fide image.

3.3.5 Ablation Study

In this section, we delve into the effect of the attention modules on the performance of our attention-based morph detector. To this end, we compare the performance of the morph detector when the number of attention modules are one, two, or three. We have already studied the case where number of attention modules are three in section 3.3.3. We plot the DET curves for the individual datasets for every number of attention modules. Table 3.3 delineates the performance of our morph detector when trained on the individual datasets for the following cases: 1- One attention module placed at $\mathscr{L}_3$ = *"mixed_7a"*, 2- two attention modules placed at $\mathscr{L}_3$ = *"mixed_7a"* and $\mathscr{L}_2$ = *"mixed_6a"*, 3- three attention modules at $\mathscr{L}_3$ = *"mixed_7a"*, $\mathscr{L}_2$ = *"mixed_6a"*, and $\mathscr{L}_1$ = *"conv2d_4b"*. Also, Table 3.4 summarizes the performance of our morph detector when trained on the universal dataset for the above-mentioned cases.

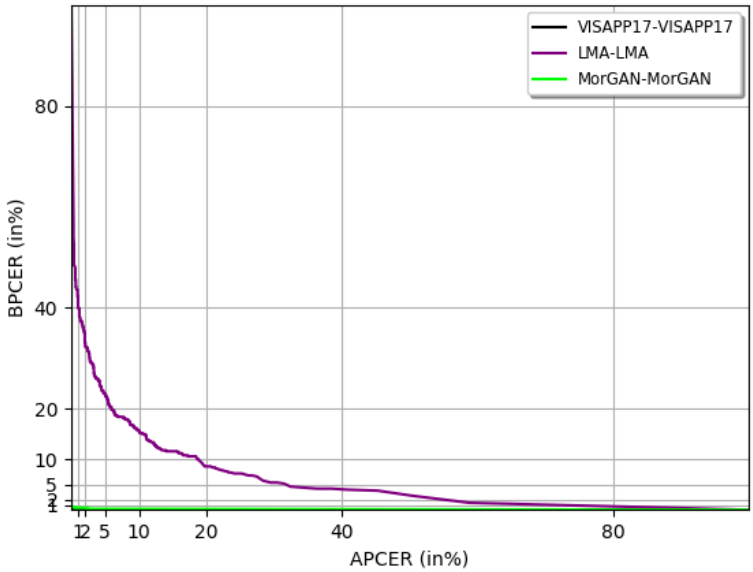


Figure 3.8: DET curves for the individual datasets for one attention module.

Fig. 3.6 depicts the performance of our morph detector using the two attention modules when our morph detector is trained using the individual datasets. Fig. 3.7 shows the performance of our morph detector using the two attention modules when our DNN is trained using the universal dataset. Moreover, Fig. 3.8 displays the performance of our attention-based morph detector with the one attention module that is trained on the individual datasets, and Fig. 3.9 displays the performance of our attention-based morph detector using the one attention module when the network is trained on the universal dataset. It is evident from Table 3.3 that the most accurate morph detection for the LMA dataset is achieved when there are three attention modules in our proposed network. Concerning Table 3.4, the attention-based DNN with three attention modules outperforms the network which has either one or two attention modules.

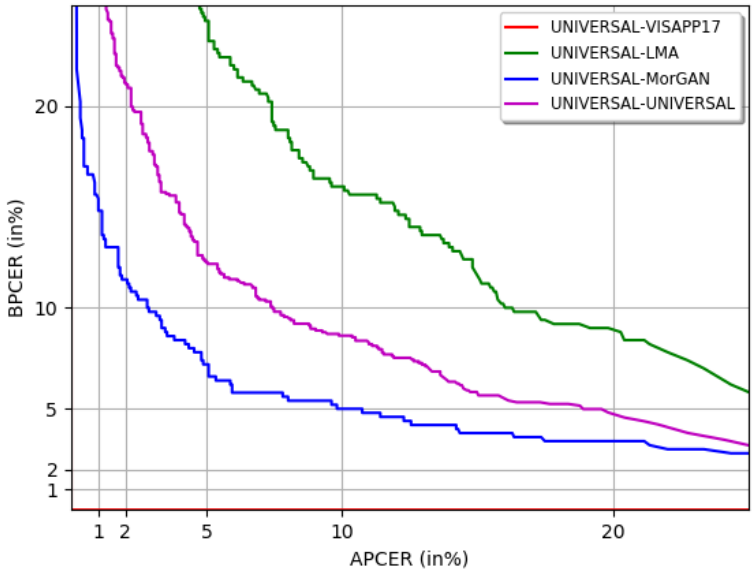


Figure 3.9: DET curves for the universal datasets for one attention module.

3.4 Conclusion

In this chapter, we studied the application of attention mechanism for detecting morphed images. More importantly, our attention-based model is adapted to a wavelet-based Inception-ResNet-v1, where all input images are decomposed into 48 wavelet sub-bands. The three integrated attention modules can emphasize the artifacts stem from the morphing attack, leading to detecting morphed images accurately. Most importantly, our attention-based morph detector can detect morphed images in the VISAPP17 and MorGAN datasets accurately. Displayed attention maps substantiates the effectiveness our algorithm in detecting morphed images, because morphed images have substantial attentive pixels compared to bona fide images. Finally, our ablation study proves the superior performance of our attention-based morph detector that uses three attention modules in comparison to a network that has either one or two attention modules.

Chapter 4

Morph Detection Enhanced by Structured Group Sparsity

4.1 Introduction

Face forgery detection has gained momentum recently in the biometric community owing to its vast application, especially in commercial face recognition systems [9,53,113–116]. Photo-realistic forged images tamper with the functionality and integrity of security checkpoints, where, ideally, there must be a zero-tolerance policy to false acceptance [13, 15, 117, 118]. Introduced in [13], facial morph images, as one of the categories of the forged face images, can bypass established automated face recognition systems, as well as border control officers, where both struggle to distinguish a bona fide image from a morphed one [119] due to delicacy in synthesizing morphed samples. Face morphing attacks are synthesized using two look-alike genuine images, for example one representing a criminal and one for an innocent subject, in which the final morphed image can be verified against both subjects. Two underlying genuine images are sampled from two distributions, and the resulting morphed sample is characterized using the blended features of the two genuine images. If we assume the support of high-dimensional genuine and morphed samples are two underlying embedding low-dimensional manifolds, morphed samples are intentionally crafted near the discriminating boundary of the two manifolds, which justifies its verifiability against both real subjects.

Face morphing attacks are forged in either the image domain or in the latent domain. In

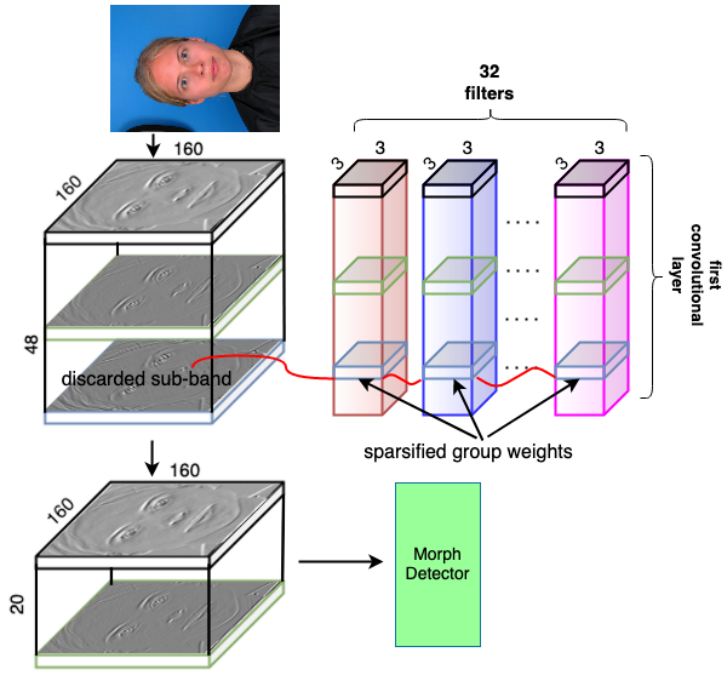


Figure 4.1: Group Lasso regularization, as a representation learning, leads to selecting the most discriminative sub-bands for detecting a morphed image.

image domain morphing, two genuine images are translated into a set of aligned averaged landmarks and a morphed sample is synthesized after warping and alpha blending. In latent space morphing, a generative adversarial network, which has an attached encoder, first captures the distribution of genuine images, and converts two genuine images into two latent vectors that can be mixed using the convex combination of both vectors. From the detection standpoint, previous research efforts have considered two approaches to detect morphed samples. Single morphed image detection, which is the focus of this study, labels an image as either genuine (negative) or morphed (positive). On the other hand, differential morph detection employs a probe image and an auxiliary one, which is usually a live photo of a subject, to define morph detection framework.

Deep learning-based techniques have shown compelling results for detecting morphed images through harnessing representation learning [120] by mapping data samples into an embedding space where the separability between genuine and morphed samples is guaranteed through the aligning parameter space of a Deep Neural Network (DNN). Since the discrepancies between a bona fide and its corresponding morphed image are subtle and local, fine-grained

feature learning [115,121] can be tailored for our morph detection algorithm. To mitigate the curse of dimensionality, feature selection is a powerful tool to find the most discriminating hyperplane in a binary classification setting. A DNN can pick the most discriminative structured group features of its input space by adjusting its kernel parameters used in its first convolutional layer. Enforcing a group sparsity constraint over the weights of the first convolutional layer in a DNN through the manipulation of its loss function can guide the parameter space to convergence on a set of parameters that picks the most discriminative channels of input data (see Figure 4.1).

In this work, we tackle single image morph detection. Inspired by the aforementioned feature selection scheme, we investigate the application of group-L1 sparsity [21] over the weights of the first convolutional layer in a DNN as the criterion for selecting the most discriminative input samples' wavelet sub-bands. Discriminative wavelet sub-bands, which can be thought of as fine-grained features, are learned during the training of our DNN, guaranteeing that an optimal hyperplane will be found between genuine and morphed samples. We conduct experiments on three morph image datasets, i.e., VISAPP17 [5], LMA [7], and MorGAN [7]. Given supervised data samples, our DNN sparsifies the kernel parameters of the first convolutional layer on-the-fly while training, which drives our sparsity-guided DNN into detecting morphed samples.

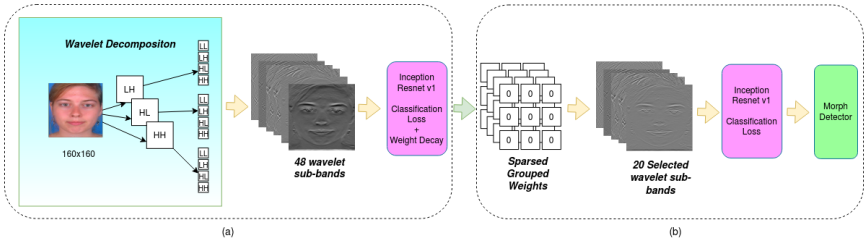


Figure 4.2: (a): Our modified Inception-ResNet-v1 [8] is trained with the 48-channel samples of both real and morphed images. Group Lasso constraint sparsifies the grouped weights of the first convolutional layer leading to sub-band selection.
(b): We retrain our modified DNN using the 20 discriminative wavelet sub-bands, where the input channel size of our DNN is reduced to 20. A binary classifier is learned for morph detection.

4.2 Proposed Framework

To incorporate fine-grained spatial-frequency features into our proposed framework, we leverage wavelet domain analysis as the basis of our deep morph detector. At varied granularity, wavelet sub-bands contain local discriminative information, where morphing artifacts are uncovered in this domain. Our wavelet-based deep morph detection mechanism is two-fold. First, we accomplish the sub-band selection to find the most informative subset of features to increase the confidence of our deep morph detector as far as inference is concerned. As presented in Figure 4.1, the optimization framework in the structured group sparsity zeros out the grouped weights corresponding to convolutional filters in a given layer of a DNN morph detector. Once some grouped weights in a convoluational layer converge to zero, those involved wavelet sub-bands are discarded, that is an implicit feature selection method. Secondly, we train our DNN detector using the selected wavelet sub-bands for the task of morph detection (see Figure 4.2).

4.2.1 Sub-band Selection Based on Group Lasso

We utilize the valuable spatial-frequency information provided by the wavelet decomposition to enhance the accuracy of our proposed morph detector. We pre-processed every input image using a three-level uniform wavelet decomposition. Since the morphing artifacts are revealed in the high frequency spectra, we discard the Low-Low (LL) sub-band after the first level of decomposition, and the remaining 48 wavelet sub-bands are extracted to be used by our deep morph detector.

Since we intend to select the most discriminative wavelet sub-bands, our focus is on the first convolutional layer of our DNN. Please note that the input consists of C wavelet sub-bands (channels) and the first convolutional layer is defined in the space of $R^{N\times C\times H\times V}$ where N, C, H, and V represent the number of filters, number of kernels, height, and width of a kernel, respectively. In this study, the filter and the kernel terms are distinguished. A kernel is a 2D

array of size $H \times V$. A filter is a 3D array of size $C \times H \times V$, which are stacked 2D kernels over channel axis. Input images are decomposed into 48 concatenated wavelet sub-bands as the input of DNN. Therefore, number of kernels in each filter of the first convolutional layer is equal to 48. 32 different filters are employed in the first convolutional layer, where the size of each kernel is 3×3. Thus, the dimensions of the first convolutional layer for the purpose of channel-wise feature selection are as follows: $N = 32, C = 48, H = 3$, and $V = 3$. There are 48 different grouped weights that are shown by $w_{l1}(:,c,:,:)$ for c $\in \{1,...,48\}$, where the first layer weights are denoted by w_{l1}. Discriminative wavelet sub-bands are selected according to a supervised feature selection algorithm. To select wavelet sub-bands, i.e. channel-wise feature selecton, we impose a group sparsity constraint on the parameters of the first convolutional layer of our DNN. Integrating weight decay in the classification loss of our DNN on the weights of the first convolutional layer, known as structured sparsity regularization penalty, drives our network into sparsifying the grouped weights of the first convolutional layer, which implicitly results in discarding irrelevant wavelet sub-bands. Consecuently, a limited number of informative wavelet sub-bands, out of 48, are selected, with which we train our DNN morph detector.

4.2.2 Rewriting the Classification Loss of the DNN Detector

If we denote the classification loss of our DNN detector as $\mathscr{L}_{cl.}(w)$, the set of parameters of our network as w, the first layer weights as w_{l1}, and each grouped weight in the first layer as $w_{l1}^{(g)}$, the regularized loss function, denoted by $\mathscr{L}_{\mathscr{R}}(w)$, for training the deep neural network is as follows:

$$\mathscr{L}_{\mathscr{R}}(w) = \mathscr{L}_{cl.}(w) + \lambda \|w_{l1}\|_{1,2} = \mathscr{L}_{cl.}(w) + \lambda \sum_{g \in \mathscr{G}_{l1}} \|w_{l1}^{(g)}\|_2, \tag{4.1}$$

where $\mathscr{G}_{l1}$ is a set composed of all the group weights of the convolutional filters in the first layer, and λ is a parameter controlling the amount of sparsity. The regularized loss can be re-written as:

$$\mathscr{L}_{\mathscr{R}}(w) = \mathscr{L}_{cl.}(w) + \lambda \sum_{c=1}^{C} \sqrt{\sum_{n=1}^{N} \sum_{h=1}^{H} \sum_{v=1}^{V} w_{l1}^2(n,c,h,v)}, \tag{4.2}$$

where $\mathscr{L}_{cl.}(w)$ delineates the binary cross-entropy classification loss and $N = 32, C = 48, H = 3$, and $V = 3$.

4.2.3 Learning Deep Morph Detector

Regarding our feature selection scheme, as mentioned in Eqs. 1 and 2, there is a hyperparameter λ in the optimization framework of our DNN detector, which is the regularization coefficient. To find the optimal regularization coefficient we utilize the validation sets of our datasets. To incorporate all three datasets in the selection process of hyperparameter λ, we combine all the images in the three datasets, i.e., VISAPP17 [5], LMA [7], and MorGAN [7], and we create a "universal dataset". A hyperparameter search is performed over different values of λ. For each value of λ, our group sparsity-constrained DNN is trained using the training portion of the universal dataset, and the performance of the trained DNN detector is evaluated through Area Under the Curve (AUC) using the validation portion of the universal dataset. The highest AUC corresponds to the optimal value for λ. Once, we obtained the most discriminative wavelet sub-bands, we retrain our DNN using the selected wavelet sub-bands. Please note that since the number of input wavelet sub-bands is reduced due to feature selection, the number of channels C in the filters of the first convolutional layer is also reduced.

4.3 Evaluations

4.3.1 Datasets

We employ the VISAPP17 [5], MorGAN [7], and LMA [7] datasets in this work. The VISAPP17 and LMA datasets were generated using facial landmark manipulation techniques, which is an alpha blending of the warped bona fide images. On the other hand,

the MorGAN dataset is synthesized using a GAN, including a decoder network that utilizes transposed convolutional layers to transform a convex combination of two generated latent vectors into the image domain. The VISAPP17 dataset has been generated using the images in the Utrecht FCVP dataset, and the LMA and MorGAN datasets were generated using the CelebA dataset [122]. For all images, face detection is performed via MTCNN and all images are resized to 160×160. We apply 2D undecimated wavelet decomposition on all images, and since we get 48 concatenated wavelet sub-bands for each image, the dimension of each data sample is $48 \times 160 \times 160$.

As for the universal dataset introduced in section 4.2.3, the training set includes 1,631 bona fide, and 1,183 morphed images, the validation set consists of 462 bona fide, and 167 morphed samples and the test set includes 1,631 bonafide, and 1,183 morphed images.

4.3.2 Experimental Setup

We adopt a modified version of the Inception-ResNet-v1 [8] as the backbone of our DNN architecture for learning the discriminative sub-bands, as well as distinguishing morphed samples. We change the number of channels in the first convolutional layer of the Inception-ResNet-v1 to 48 during the sub-band selection stage. As mentioned in section 4.2.1, a sample input consists of 48 stacked wavelet sub-bands; thus, the convolutional filters of the first layer have 48 channels, as seen in Figure 4.1. Inspired by [21], we use group L_1-regularization to impose structured group sparsity constraint on the grouped weight parameters in the first convolutional layer of our deep neural network. For training our modified Inception-ResNet-v1, we adopt the Adam optimizer and training is done for 150 epochs. The learning schedule is as follows: the learning rate is initialized with 0.001, and it is divided by 10 after every 20 epochs. The training phase is accelerated using two 12 GB TITAN X (Pascal) GPUs.

4.3.3 Evaluation Metrics

We use the APCER, BPCER, and D-EER metrics to assess the performance of our deep morph detector. The first metric, Attack Presentation Classification Error Rate (APCER), is the percentage of morphed images that are classified as bona fide, and the second metric, Bona Fide Presentation Classification Error Rate (BPCER), represents percentage of bona fide samples that are classified as morphed. D-EER stands for Detection Equal Error Rate, at which APCER equals BPCER. The BPCER5 is the BPCER rate when APCER=5%, and similarly the BPCER10 is the BPCER rate when APCER=10%.

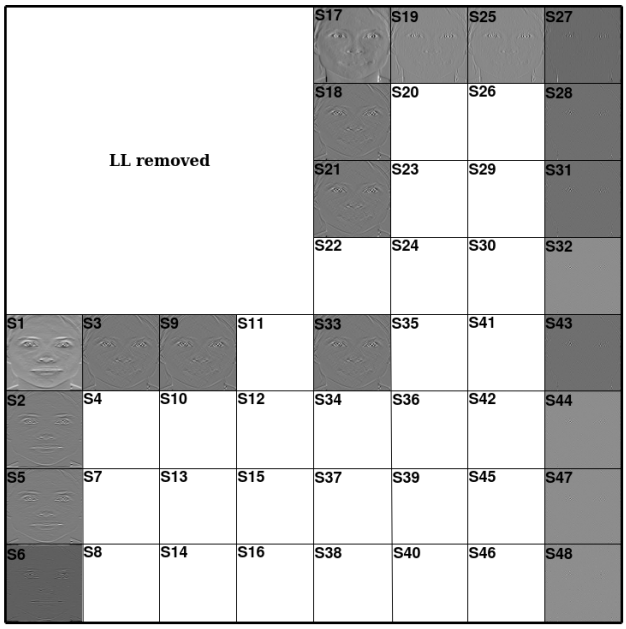


Figure 4.3: Selected discriminative sub-bands using structured group sparsity. The white areas represent the irrelevant sub-bands that are discarded. The remaining informative sub-bands are displayed.

4.3.4 Tuning the Group Sparsity Regularization Hyperparameter λ

We performed a search over the group sparsity regularization coefficient λ. To find the optimal value for λ, as discussed in Section 4.2.3, we fine tune our modified Inception-ResNet-v1 DNN [8], already pre-trained on VGGFace2, using the training portion of the universal

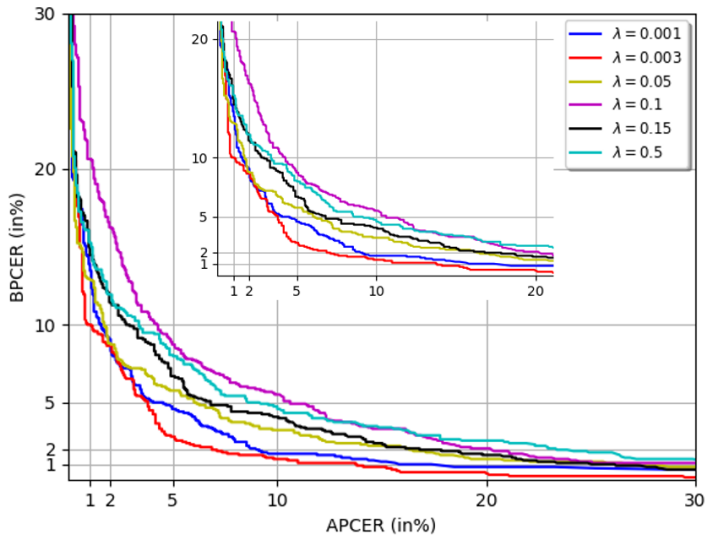


Figure 4.4: DET curves corresponding to different values of λ evaluated on the validation portion of the universal dataset.

dataset. For each value of hyperparameter λ, we trained our modified Inception-ResNet-v1 DNN using the training portion of the universal dataset, and we found the AUC metric when our trained DNN is evaluated using the validation portion of the universal dataset. It should be noted that we created the universal dataset for hyperparameter selection since we wanted to train a morph detector that performs well across different morphing techniques. In addition, after our network is fully trained, we zero out any grouped weight with a weight parameters norm smaller than 0.001. It was found that $\lambda = 0.003$ yields the top 20 most discriminative sub-bands, with the corresponding highest AUC of 99.31%. Figure 4.3 depicts the selected sub-bands along with the corresponding numbers after training our network using universal dataset, and $\lambda = 0.003$.

Considering the aforementioned hyperparameter tuning process, we displayed the performance of our morph detector for some selected values of λ using the validation portion of the universal dataset. Figure 4.4 shows the Detection Error Trade-off (DET) curves corresponding to different group sparsity regularization parameter λ. We can clearly see that the performance of our deep morph detector is at its best when $\lambda = 0.003$.

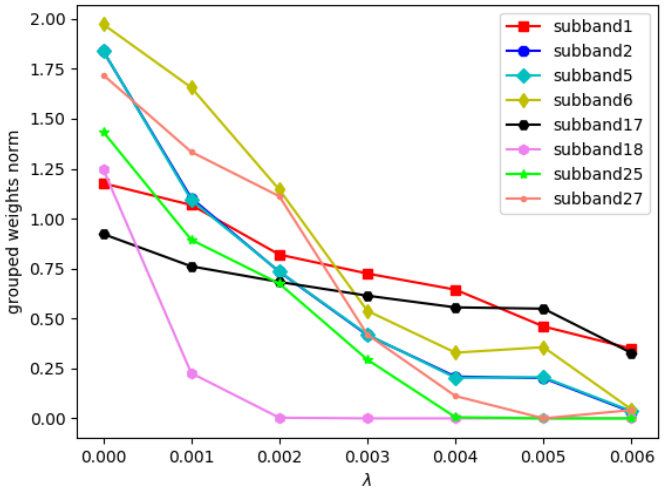


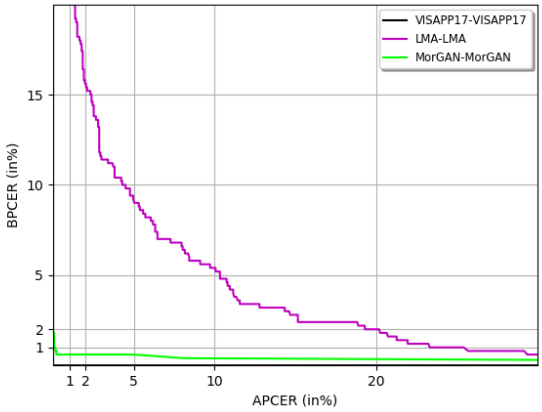
Figure 4.5: Displaying grouped weights decay related to some of the selected 20 sub-bands with respect to group sparsity hyperparameter λ.

4.3.5 Grouped Weights Decay

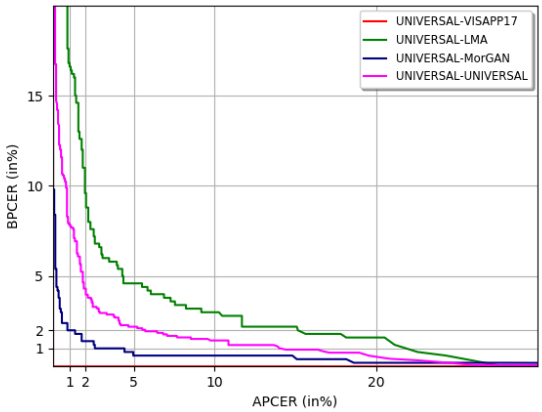
To show the functionality of the group sparsity regularization coefficient λ, we plot the norms of the grouped weights in the first convolutional layer of our DNN as a function of the group sparsity regularization coefficient λ for some of the selected wavelet sub-bands (see Figure 4.5). Increasing the group sparsity penalty coefficient λ will push the grouped weight norms toward zero as expected.

4.3.6 Performance of the Deep Morph Detector

Once we found the top 20 discriminative wavelet sub-bands, we stacked the selected 20 wavelet sub-bands as data samples, and we retrained our modified Inception-ResNet-v1. It should be pointed out that the number of channels is reduced to 20. In the first scenario, we train the DNN using the 20-stacked wavelet sub-bands of each individual dataset, and the performance of the our morph detector is reported using the evaluation metrics introduced in section 4.3.3. Table 4.1 delineates the performance of our deep morph detector when



(a)



(b)

Figure 4.6: DET curves which display the performance of our morph detector when (a) trained and evaluated on individual datasets and (b) trained on the universal dataset.

evaluated on our three datasets. Moreover, the corresponding DET curves are shown in Figure 4.6a.

In the second scenario, we trained our DNN detector using the training portion of the universal dataset, and the performance of the deep morph detector is reported using the test set of the universal dataset, as well as the test sets of individual datasets. Table 4.2 summarizes the evaluation of our morph detector when trained on the universal dataset, and the pertinent DET curves are plotted in Figure 4.6b.

Table 4.1: Performance of single morph detection: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Dataset	Algorithm	D-EER	5%	10%
VISAPP17	BSIF+SVM [92]	16.51	35.61	26.79
	SIFT+SVM [93]	38.59	82.40	75.60
	LBP+SVM [95]	38.00	77.10	67.90
	SURF+SVM [94]	30.45	84.70	69.40
	RGB+DNN [8]	1.76	0.588	0.58
	48-sub-bands	0.00	0.00	0.00
	Ours	**0.00**	**0.00**	**0.00**
LMA	BSIF+SVM [92]	33.05	78.34	62.86
	SIFT+SVM [93]	33.30	83.40	72.00
	LBP+SVM [95]	28.00	58.60	51.40
	SURF+SVM [94]	37.40	79.50	70.00
	RGB+DNN [8]	9.10	15.18	7.49
	48-sub-bands	5.04	4.38	2.75
	Ours	**6.80**	**8.60**	**4.89**
MorGAN	BSIF+SVM [92]	1.57	1.42	1.30
	SIFT+SVM [93]	43.50	93.20	84.20
	LBP+SVM [95]	20.10	52.70	32.30
	SURF+SVM [94]	39.95	80.00	72.60
	RGB+DNN [8]	2.44	1.88	1.50
	48-sub-bands	0.81	0.59	0.32
	Ours	**0.42**	**0.38**	**0.22**

Please note that in Table 4.1 and Table 4.2, RGB+DNN represents our baseline when the Inception-ResNet-v1 is trained on the original RGB images, and 48-sub-band data indicates the data samples that consist of 48 wavelet sub-bands without utilizing the structured group sparsity. Our results mentioned in Table 4.1 and Table 4.2 prove the effectiveness of feature selection scheme for detecting morphed samples. In particular, the morphed samples in the VISAPP17 and MorGAN datasets are detected precisely compared with the LMA dataset. Please note that the performance of our RGB+DNN baseline, which is trained on the Inception-ResNet-v1, is on par with the performance of other DNNs. In particular, regarding the state-of-the-art results, in [47] the morph detection results on the LMA dataset is as follows: D-EER: 0.00, BPCER@APCER=10%: 0.00 , BPCER@APCER=20%: 0.00. Also, their results on the MorGAN dataset is as follows: D-EER: 34.00, BPCER@APCER=10%: 67.00, BPCER@APCER=20%: 78.00. Comparing the state-of-the-art results with ours reveals that our method outperforms on the MorGAN dataset considerably and perform close to this baseline on the LMA dataset.

Table 4.2: Performance of single morph detection: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Train	Test	Algorithm	D-EER	5%	10%
Universal(VISAPP17+MorGAN+LMA)	VISAPP17	BSIF+SVM [92]	35.00	67.20	59.00
		SIFT+SVM [93]	27.00	83.20	70.90
		LBP+SVM [95]	37.67	72.50	59.50
		SURF+SVM [94]	31.00	79.40	70.10
		RGB+DNN [8]	0.00	0.00	0.00
		48-sub-bands	0.00	0.00	0.00
		Ours	**0.00**	**0.00**	**0.00**
	LMA	BSIF+SVM [92]	30.00	70.42	57.60
		SIFT+SVM [93]	28.31	67.70	50.00
		LBP+SVM [95]	29.00	61.50	51.20
		SURF+SVM [94]	33.40	74.50	62.70
		RGB+DNN [8]	7.80	13.00	6.10
		48-sub-bands	4.62	4.22	2.73
		Ours	**4.44**	**4.11**	**2.21**
	MorGAN	BSIF+SVM [92]	28.80	62.42	45.70
		SIFT+SVM [93]	47.60	92.30	88.60
		LBP+SVM [95]	31.20	62.00	55.60
		SURF+SVM [94]	38.67	76.00	70.00
		RGB+DNN [8]	4.69	4.70	2.74
		48-sub-bands	1.11	0.43	0.34
		Ours	**1.53**	**0.32**	**0.30**
	Universal	BSIF+SVM [92]	23.74	51.42	38.67
		SIFT+SVM [93]	37.21	87.45	76.71
		LBP+SVM [95]	38.80	91.36	83.40
		SURF+SVM [94]	36.00	75.50	65.76
		RGB+DNN [8]	5.57	6.08	3.00
		48-sub-bands	3.12	1.78	0.97
		Ours	**2.78**	**1.75**	**1.21**

4.4 Visualizing the Functionality of the Deep Morph Detector

In this section, we adopt a few visualization techniques to explain the underlying mechanism of our morph detector. First, we utilize the t-distributed Stochastic Neighbor Embedding (t-SNE) [123] visualization technique to explain the classification performance improvement due to the imposed structured sparsity. Second, we utilize the Gradient-weighted Class Activation Mapping (Grad-CAM) [124] to show the most attended spatial regions in the input images when our trained classifier labels an input image as a bona fide or morphed image.

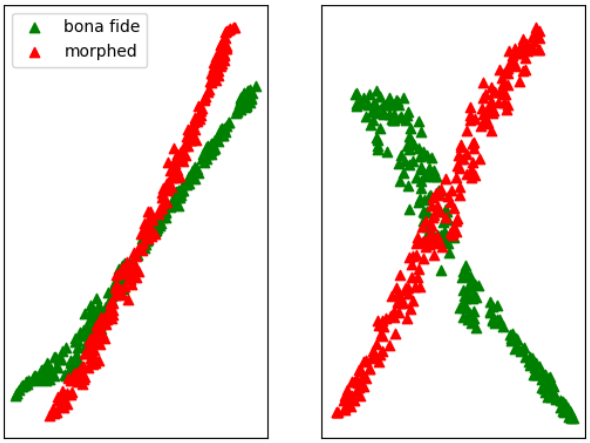


Figure 4.7: T-SNE visualization. The left figure depicts the 48-sub-band data, and the right figure shows the 20-sub-band data from the MorGAN dataset.

4.4.1 Visualizing the Functionality of Structured Group Sparsity

To display the efficacy of our feature selection scheme, which is selecting the most discriminative sub-bands, the t-SNE visualization technique is employed as a representative medium which preserves the local structure of samples when visualizing high dimensional data samples in a low dimensional space. We randomly select 200 bona fide, and 200 morphed samples from the MorGAN dataset for the following two cases. In the first case, we extract the DNN embedding features for the original 48-sub-band data samples for the 200 bona fide and 200 morphed sample. Please note that the employed DNN for feature extraction was already trained on the 48-sub-band data. As the second case, we find the deep features using a DNN trained on the 20-sub-band data for the same 200 bona fide and 200 morphed samples. The point here is that we use the same data samples for both cases, but with different number of wavelet sub-bands. These two subsets of data points are plotted using t-SNE as shown in Figure 4.7, and we see that 20-sub-band data, in the right column, are more separable compared to the 48-sub-band data in the left column, which substantiates the effectiveness of our feature selection algorithm.

4.4.2 Grad-CAM Visualization

Understanding the key spatial areas in an input image, in terms of detection or classification, has been a long-standing topic of interest in the vision community. It is worth mentioning that our DNN is a non-attention-based architecture, and we do not use any attention mechanism in our DNN. In this section we adopt another useful visualization technique to observe which regions in the input images are paid more attention to from the DNN perspective in time of inference. In other words, we want to see which pixels are considered discriminative given our morph classification task. To explain the functionality of our trained DNN, we employ the Grad-CAM, which represents the gradient-weighted class activation maps. In this visualization method, the gradient of a class-specific logit is obtained with respect to all spatial locations in a given feature map of the last convolutional layer in the DNN under scrutiny. The calculated gradients are averaged-pooled globally for each feature map, and these coefficients are used for a weighted average of the feature maps along with a final ReLU activation function to produce the class specific Grad-CAM. In accordance with the notation used in [124], the importance of the weights incorporating the pixels in the feature map k of the last convolutional layer is as follows:

$$\alpha_k^{class} = \frac{1}{Z}\sum_i\sum_j \frac{\partial y^{class}}{\partial A_{ij}^k}, \tag{4.3}$$

where Z represents the total number of spatial locations ij in the feature map k, y^{class} delineates the score or logit for the class $class$, and A denotes the activation or feature map. Consequently, the Grad-CAM produced for class $class$ with respect to the final convolutional layer in the DNN is as follows:

$$L_{Grad-CAM}^{class} = ReLU(\sum_k \alpha_k^{class} A^k). \tag{4.4}$$

To produce the class-specific Grad-CAM for a bona fide image and its corresponding morphed image, we choose the last convolutional layer in our modified Inception-ResNet-v1, which has 1792 feature maps with spatial size of 3×3. Based on Figure 4.8, the right images, that are morphed faces have substantially more attended regions compared to the left images,

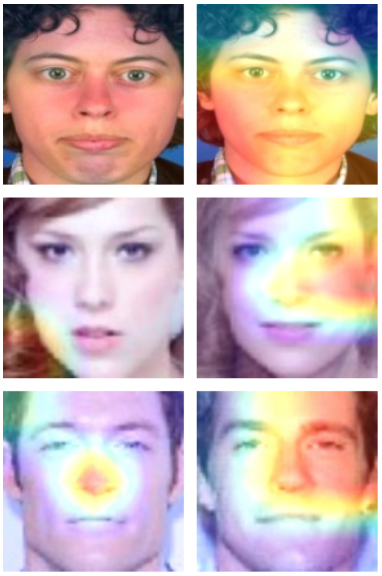


Figure 4.8: Grad-CAM visualizations for bona fides (left) and the corresponding morphed images (right). The first, second, and third rows represent samples from the VISAPP17, LMA, and MorGAN datasets, respectively.

which are the bona fide samples.

4.5 Conclusion

In this chapter, we employed structured group sparsity to force a DNN into finding the most discriminative subset of wavelet sub-bands, that are the wavelet sub-bands. To isolate the discriminating artifacts in the spatial-frequency feature domain, we adapted our framework into the wavelet domain. As far as learning the parameters of our DNN is concerned, the cost function of the DNN is constrained to meet the group Lasso condition, which is imposed on the grouped weights of the first convolutional layer. Our adjusted cost function results in finding the top 20 discriminative wavelet sub-bands, further enabling accurate morph detection with respect to our datasets. The D-EER, APCER5, and APCER10 rates obtained using our trained network with the optimal number of sub-bands substantiate the effectiveness of our framework. In particular, the morphed samples in the VISAPP17, and MorGAN datasets are detected accurately compared to the LMA dataset. In addition,

to make the effectiveness of our morph detector transparent, we utilized two visualization techniques to explain the functionality of the proposed single image morph detector.

Chapter 5

Attention Augmented Face Morph Detection

5.1 Introduction

Morphed face detection has gained a surge of interest among the biometric and vision communities [9–12]. Facial image morphing attacks have posed a serious threat to the functionality of face recognition systems, especially those adopted at borders [13]. Using a morphed image, a criminal can share a passport with his/her innocent accomplice to evade identification and detection. Both the criminal and accomplice faces are verified against the morphed images which allows the criminal to get a passport. A large body of research is devoted to generating morphed facial images mostly using either manipulating geometrical characteristic of two bona fide subjects' images [5, 7, 18] or generative networks such as Generative Adversarial Networks (GANs) [1, 7, 19].

The mainstream methods for morph detection are categorized into either single image morph detection [34, 35] or differential morph detection [34, 35]. In the former case, the goal is set to identify a probe image either as a bona fide or morphed image without utilizing any other auxiliary information. On the other hand, the latter case takes into account a live image of the subject under investigation, to classify a probe image as a bona fide or morphed. State-of-the-art multi-class classifiers or object detection frameworks benefit from rich visual abstractions realized through representation learning techniques [120]. Generally speaking,

face morph detection can be implicitly reformulated as learning discriminative informative cues that are taken into account for finding a decision boundary, separating bona fide images from morphed ones in a binary classification setting. Since artifacts in a morphed image are local, the discrepancy between a morphed image and the corresponding bona fide image can be detected using fine-grained features [115].

The 2D wavelet decomposition [38, 125] provides a useful insight into the joint spatial-frequency information embedded in a given 2D image. Wavelet sub-bands can be thought of as fine-grained features with variable granularity. Moreover, wavelet decomposition reveals the embedded hidden information through providing spatial-frequency representation. Resulting sub-bands can be harnessed to isolate artifacts in a given morphed image at different spatial-frequency granularity.

Feature learning plays a pivotal role for the mainstream computer vision tasks such as image classification. In particular, sparse representation learning methods have proved to be powerful tools for face recognition applications. Due to the NP-hardness of any sparsity-constrained optimization framework, an alternative relaxed version of the sparsity condition is enforced using the $\ell 1$ relaxation [126]. Most importantly, group sparsity [127] is defined as a class of sparse representation learning methods where a feasible solution for the formulated optimization framework converges when some of the grouped coefficients are zeroed out. In this study, we leverage the group sparsity to increase the accuracy of our morph detection framework.

Recently, visual attention mechanisms have initiated a renaissance in the image recognition and classification tasks. "Visual Explanation" [109], underlying the attention mechanism, uncovers what regions a deep neural network focuses on to form its final decision for a defined downstream task. In other words, an attention-based deep neural network forces the DNN into focusing on the most informative regions which contribute the most to learning a hypothesis, leading to a more accurate classification [20, 22, 109, 128, 129]. The feature refinement realized by the spatial- and channel-wise attentions [22] has provided a rich representation that can increase inter-class separability while minimizing intra-class dispersion.

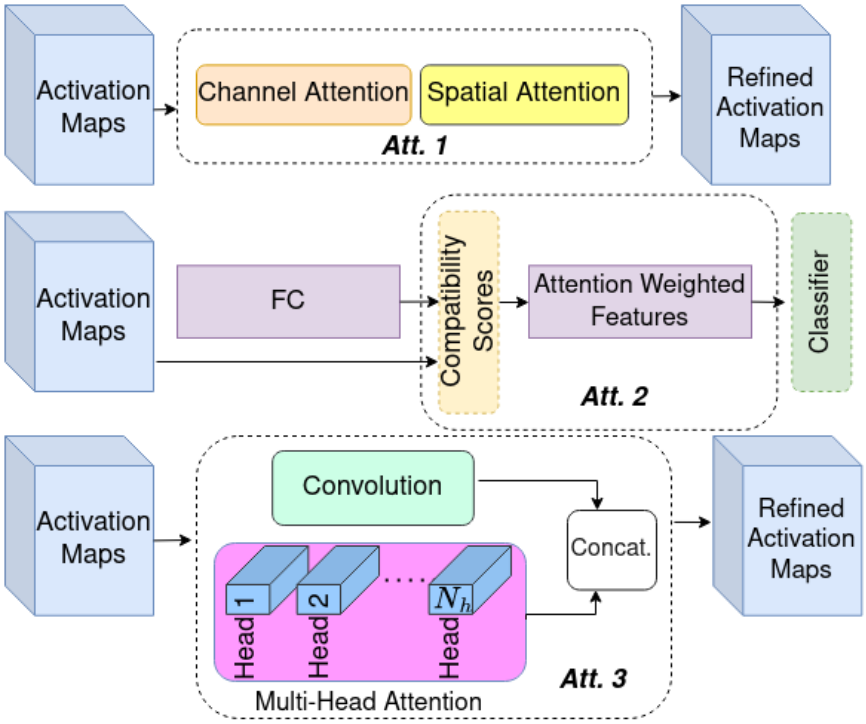


Figure 5.1: Our attention augmented framework which adopts three different attention mechanisms, i.e., *Att. I*, *Att. II*, and *Att. III* to increase the morph detection accuracy. The *Att. I* module which is the convolutional block attention adopts the max-pooling or average-pooling to find channel and spatial attention maps to highlight discriminative spatial pixels in a given set of feature maps. The *Att. II* determines the informative spatial pixel locations through finding the correlation of each spatial location in a given feature map, known as the local feature vectors, and the output of a Fully Connected (FC) layer, known as the global feature vector, in a given DNN. In addition, the *Att. III* yields augmented feature maps by concatenation of the convolutional feature maps and their corresponding self-attentional feature maps.

Also, vision transformers [89], which have shifted the paradigm in terms of classification accuracy without using any convolutional operations, have benefited from the multi-headed self-attention mechanism which also plays a pivotal role in this study.

This work investigates the application of group sparsity [21], soft attention mechanism [20], and self-attention [22, 23] for morph detection. The spatial-frequency content of an image provides useful information such as subtle discrepancies between a bona fide and its morphed image. We decompose every input image using a multilevel 2D wavelet decomposition to extract coarse-to-fine spatial-frequency wavelet sub-bands which are considered powerful representations for training a DNN morph detector. Our group sparsity constraint opti-

mization framework leads our DNN morph classifier to converge into a sparse solution where some of the wavelet sub-bands of input images, bearing minimal discriminative information, are discarded. Thus, we can select a subset of the most discriminative wavelet sub-bands, which is an implicit feature selection mechanism. On the other hand, attention modules customized to our model, guide the network to pinpoint the most informative spatial pixels as well as the most information bearing channels in a given intermediate feature map. As shown in Fig. 5.1, we incorporate three types of visual attention mechanisms into our DNN-based morph detector which guide our detector into mining the spatial regions with the highest density of morphing artifacts. Namely, we employ the spatial and channel attention modules introduced in the CBAM [22], which we call Att. I, the end-to-end soft attention mechanism delineated in [20] which is called Att. II in this study, and the self-attention augmented feature maps which is called Att. III hereafter. Through extensive experiments, we demonstrate advantage of these three mentioned attention mechanisms for improving morph detection accuracy. To increase intra-class compactness and inter-class dispersion, we employ the additive angular margin loss function (ArcFace) to obtain highly discriminative features. We demonstrate the efficacy of our framework through extensive experiments on several morph detection datasets mentioned in Section 5.3.1.

Organization of the paper is as follows: In section 5.2, we delineate our methodologies to improve morph detection accuracy. In section 5.3, we present our experiments and results. Finally, in section 5.4 we conclude our work. Our contributions in this paper are outlined as follows:

- Instead of the RGB domain, we leverage the wavelet domain to find rich spatial-frequency features of input images, i.e., wavelet sub-bands of input images.
- We employ group sparsity to select most discriminative wavelet sub-bands as a feature selection scheme for increasing morph detection accuracy.
- We integrate three different types of visual attention modules in our DNN to highlight informative spatial areas of input images which can decrease morph detection error rates.

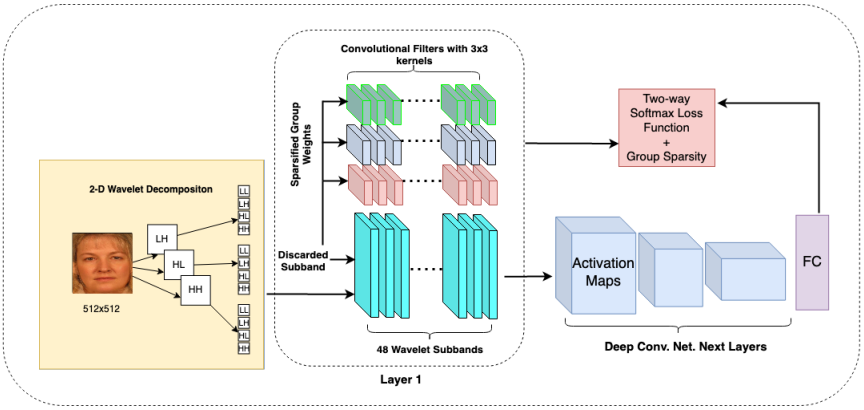


Figure 5.2: Our morph detection methodology selects the most discriminative wavelet subbands of input images which results in increasing the morph detection accuracy based on the extensive experimental evaluations.

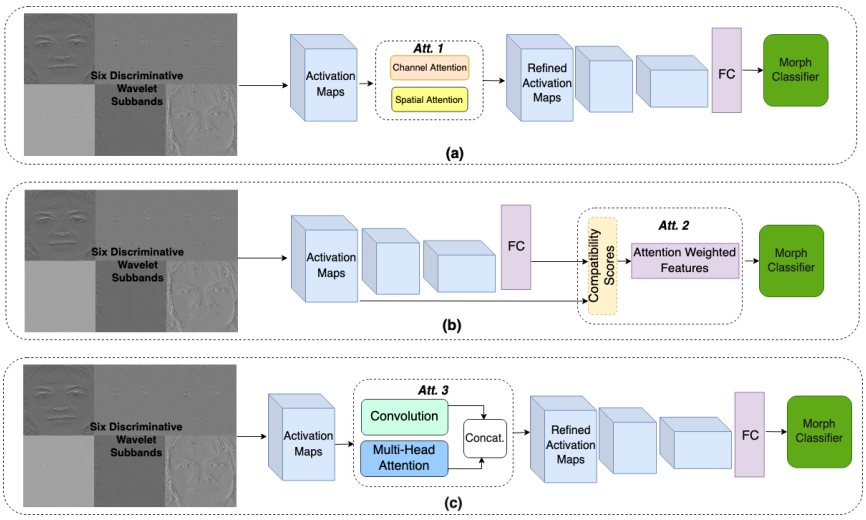


Figure 5.3: Our morph detection framework focuses on discriminative spatial regions in the selected wavelet subbands through using three different types of visual attention mechanism, called (a) *Att. I*, (b) *Att. II*, and (c) *Att. III* which results in increasing the morph detection accuracy based on the extensive experimental evaluations.

5.2 Methodology

In this paper, we propose a morph detector which leverages: (1) group sparsity for capturing the most discriminative wavelet subbands of a given facial image (see Fig. 5.2) and (2) a visual attention mechanism which drives our morph detector into the most informative spatial- and channel-wise regions to facilitate detecting morphed faces (see Fig. 5.3). To evaluate both the group sparsity and attention mechanisms in detail, we first delve into the application of group sparsity as a representation learning scheme. Moreover, the effect of different attention mechanisms is investigated separately to assess the improvement of the morph detection due to an attention-based network. Finally, we train our wavelet-based attention augmented morph detector which includes three different types of attention modules. The final objective of this paper is the joint optimization of the group sparsity and attention mechanisms.

From the information theoretic perspective, an optimal DNN architecture must meet the following conditions [130]: (1) Minimizing the mutual information between an intermediate feature map at a given layer L, denoted by $\mathbf{F}_L$, and the next layer feature map $\mathbf{F}_{L+1}$ in the hierarchy of a DNN. In other words, $I(\mathbf{F}_L;\mathbf{F}_{L+1})$ must be minimized. (2) Maximizing the mutual information between a given intermediate feature map $\mathbf{F}_L$ and output of the DNN, denoted by Y. In other words, $I(\mathbf{F}_L;Y)$ must be maximized. Employing structured group sparsity for selecting the most discriminative wavelet sub-bands and the attention mechanism to find the most discriminative spatial regions aids our DNN in minimizing the $I(\mathbf{F}_L;\mathbf{F}_{L+1})$ while maximizing $I(\mathbf{F}_L;Y)$. In a similar vein to skip connections introduced in the ResNet [100] architecture, which precludes information flow loss, our adopted feature refinement schemes enable minimizing $I(\mathbf{F}_L;\mathbf{F}_{L+1})$ to prevent losing information when data abstraction becomes compact in the higher layers of a DNN, while at the same time, $I(\mathbf{F}_L;Y)$ is maximized through finding the most relevant information in a given feature map.

5.2.1 Channel-wise Feature Selection

In this study, instead of experimenting on images in the original RGB spatial domain, we decompose all images using 2D wavelet decomposition which enables us to experiment on the fine-grained information in the spatial-frequency domain. The wavelet domain has proved to be a rich representation which provides information with different granularity. We extract the most useful spatial-frequency information, which is realized through sub-band selection detailed in this subsection, helping us to localize morphing artifacts accurately compared with the RGB domain. To this end, we adopt an undecimated uniform 2D wavelet decomposition. Needless to say, wavelet decomposition cannot be applied on the RGB images which have three channels. Therefore, the RGB images are first converted to the grayscale version using the Open CV RGB to grayscale conversion function in order to be passed to the wavelet decomposition module. We decompose three levels of wavelet decomposition, and from the resulting 64 wavelet sub-bands, we keep 48 sub-bands which represent the high frequency spectra. In other words, we discard the Low-Low (LL) sub-band after one level of wavelet decomposition. Our objectives are as follows: (1) channel-wise feature selection for selecting the most discriminative wavelet sub-bands from these 48 sub-bands which can help us distinguish bona fide images from morphed ones, and (2) spatial feature selection by employing different attention mechanisms to localize the most discriminative pixels in the selected wavelet sub-bands.

We adopt a group sparsity feature selection scheme to select the most discriminative wavelet sub-bands, mentioned above as the sub-band selection, for a given input image. Our implicit feature selection scheme based on the group sparsity is realized by imposing a group sparsity constraint on the parameters of the first convolutional layer in our DNN morph detector (see Fig. 5.2). We select the most discriminative wavelet sub-bands by discarding wavelet sub-bands that their corresponding kernel weights in the first convolutional layer of our DNN converge to zero thanks to the enforced group sparsity constraint on the parameters of the first convolutional layer. Note that the input images are composed of C wavelet sub-bands (channels) and the first convolutional layer is defined in the space of $\mathbb{R}^{N \times C \times H \times V}$

where N, C, H, and V represent the number of filters, number of kernels, height, and width of a kernel, respectively. In this study, the filter and the kernel terms are distinguished. A kernel is a 2D array of size $Q \times V$. A filter is a 3D array of size $C \times Q \times V$, which are stacked 2D kernels over the channel axis. Input images are decomposed into $C = 48$ stacked wavelet sub-bands as the input of our DNN. Therefore, the number of kernels in each filter of the first convolutional layer is equal to 48. 32 different filters are employed in the first convolutional layer, where the size of each kernel is 3×3. Thus, the dimensions of the first convolutional layer for the purpose of channel-wise feature selection are as follows: $N = 32, C = 48, Q = 3$, and $V = 3$. There are 48 different grouped weights that are shown by $w_{l1}(:, c, :, :)$ for c $\in \{1, ..., 48\}$, where the first layer weights are denoted by w_{l1}.

As discussed above, to select the wavelet sub-bands, i.e., channel-wise feature selection, we impose a group sparsity constraint on the parameters of the first convolutional layer of our DNN. Integrating a group sparsity term in the classification loss of our DNN on the weights of the first convolutional layer, known as a structured sparsity regularization penalty, drives our network into sparsifying the grouped weights of the first convolutional layer, which implicitly results in discarding non-discriminative wavelet sub-bands. Consequently, a subset of informative wavelet sub-bands, out of 48 sub-bands, are selected. In other words, after training our DNN, some of the grouped weights $w_{l1}(:, c, :, :)$ for c $\in \{1, ..., 48\}$ are zeroed out which means their corresponding wavelet sub-bands are discarded. Those grouped weights that do not converge to zero determine the most useful discriminative wavelet sub-bands which are used for the downstream task of morph detection.

5.2.2 ArcFace Loss Function

Suppose we denote the set of parameters of our network as w, first layer weight parameters as w_{l1}, the binary classification loss of our DNN as $\mathscr{L}_{cl.}(w)$, and each grouped weight in the first layer as $w_{l1}^{(g)}$. Note that group weight corresponding to the Cth channel is defined as all the 2D kernels in the Cth channels of all the N=32 filters in our first convolutional layer. The regularized loss function, denoted by $\mathscr{L}_{\mathscr{R}}(w)$, for training our DNN morph detector to

select the most discriminative wavelet sub-bands is as follows:

$$\mathscr{L}_{\mathscr{R}}(w) = \mathscr{L}_{cl.}(w) + \lambda \|w_{l1}\|_{1,2} = \mathscr{L}_{cl.}(w) + \lambda \sum_{g \in \mathscr{G}_{l1}} \|w_{l1}^{(g)}\|_2, \tag{5.1}$$

where $\mathscr{G}_{l1}$ is a set composed of all the group weights of the convolutional filters in the first layer, and λ is a parameter controlling the amount of sparsity. The regularized loss can be written as:

$$\mathscr{L}_{\mathscr{R}}(w) = \mathscr{L}_{cl.}(w) + \lambda \sum_{c=1}^{C} \sqrt{\sum_{n=1}^{N} \sum_{q=1}^{Q} \sum_{v=1}^{V} w_{l1}^2(n,c,h,v)}, \tag{5.2}$$

where $\mathscr{L}_{cl.}(w)$ is the binary classification loss and $N = 32, C = 48, Q = 3$, and $V = 3$.

As for the binary classification loss in Eq. 5.2, we adopt the additive angular margin loss (ArcFace) [131] which has proved to enhance the intra-class compactness and inter-class separation. Thus we can write the $\mathscr{L}_{cl.}(w)$ as:

$$-\frac{1}{M} \sum_{i=1}^{i=M} \log \frac{\exp(s\cos(\theta_{y_i} + m))}{\exp(s\cos(\theta_{y_i} + m)) + \sum_{j=1, j \neq i}^{j=C} \exp(s\cos(\theta_j))}, \tag{5.3}$$

where M is the number of training samples in a given batch, s is the scale factor for learned feature embeddings and class weight vectors are normalized to 1 ($\|W_c\|_2 = 1$). θ_{y_i} is the angle between the ith input feature embedding of a ground truth class y and learned weight of class y. Moreover, θ_j represents the angle between the ith input embedding relevant to the ground truth class y and learned weight of class j. The additive angular margin m reduces the variance of the learned features in a given class while increasing the inter-class feature dispersion.

5.2.3 Spatial Feature Selection and Refinement

To select the most discriminative pixels, we investigate the application of three attention mechanisms which can suppress spatial regions that do not contribute to the final decision of our morph detector. In other words, our integrated attention modules allow our DNN morph detector to focus on discriminative spatial regions (see Fig. 5.3). The three attention mechanism are as follows:

Attention Mechanism I: Convolutional Block Attention Module (CBAM)

In our first attention module, shown in Fig. 5.3. (a), and called Att. I, we employ the channel and spatial attention. Specifically, we employ the Convolutional Block Attention Module (CBAM) [22, 132], to refine an intermediate feature map $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$. Given intermediate feature map $\mathbf{F}$, the CBAM module captures interdependencies between spatial-channel pixels in the feature map through inferring a 1-D channel attention map $\mathbf{M_c} \in R^{C\times 1\times 1}$ and a 2D spatial attention map $\mathbf{M_s} \in R^{1\times H\times W}$ which are as follows:

$$\mathbf{M_c}(\mathbf{F}) = \sigma(MLP(AvgPool(\mathbf{F})) + MLP(MaxPool(\mathbf{F}))), \tag{5.4}$$

$$\mathbf{M_s}(\mathbf{F}) = \sigma(conv2D[AvgPool(\mathbf{F}), MaxPool(\mathbf{F})]), \tag{5.5}$$

where MLP stands for the Multi-Layer Perceptron, which is typically a two layer fully-connected network and σ is the non-linear activation function. To find the channel attention map $\mathbf{M_c}$, we reduce the size of the hidden layer in the MLP by setting the variable "reduction_ratio=16" [132] to reduce complexity of the problem, which means that the size of the hidden layer in the MLP is $\frac{1}{16}$ of the input layer size. Also, conv2D is a convolution applied on the concatenation of (1) the average pooled feature map along the channel axis and (2) the max pooled feature map along the channel axis. The refined attentive feature map $\mathbf{F}''$ is found consecutively which is as follows:

$$\mathbf{F}' = \mathbf{M_c} \bigotimes \mathbf{F}, \mathbf{F}'' = \mathbf{M_s} \bigotimes \mathbf{F}'. \tag{5.6}$$

Please note that we employ up to two attention modules of type Att. I in different intermediate feature maps related to two different convolutional layers in our DNN-based morph detector.

Attention Mechanism II: Learn To Pay Attention

The soft attention mechanism [20, 133], called Att. II, and shown in Fig. 5.3. (b), finds the correlation of each spatial location in a given intermediate feature map $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$ and the output

of the underlying DNN, which can be one of the fully connected layers, which precedes the logits layer of our DNN, to assess how much that spatial location is deemed discriminative in the eye of the DNN. The correlation is also known as the compatibility score. Mathematically speaking, assume a spatial local feature vector in the location $i \in \{1,2,..,n\}$ in the intermediate feature map $\mathbf{F}$ is shown by $\boldsymbol{\ell}_i^{\mathbf{F}}$. Note that n is the total number of pixel locations in the given feature map which is $H \times W$. The compatibility score for local feature vector $\boldsymbol{\ell}_i^{\mathbf{F}}$ is given as:

$$c_i^{\mathbf{F}} = \langle \boldsymbol{\ell}_i^{\mathbf{F}}, \boldsymbol{g} \rangle, i \in \{1,2,..,n\}, \tag{5.7}$$

where $\boldsymbol{g}$ designates the global feature vector, that is the 512-D output feature embedding of the last fully connected layer in the DNN detector and $\langle .,. \rangle$ represents the inner product. Compatibility scores for the feature map $\mathbf{F}$ are normalized using the softmax normalization function. Normalized compatibility scores represent attention weights, which are given as:

$$a_i^{\mathbf{F}} = \frac{\exp(c_i^{\mathbf{F}})}{\Sigma_{i=1}^{i=n} \exp(c_i^{\mathbf{F}})}, i \in \{1,2,..,n\}. \tag{5.8}$$

A convex combination of the local feature vectors $\boldsymbol{\ell}_i^{\mathbf{F}}$, which is the attention weighted sum of the local feature vectors $\boldsymbol{\ell}_i^{\mathbf{F}}$ gives the refined attentive descriptors, known as attentive global feature vector, for the given feature map $\mathbf{F}$. The attentive global feature vector, i.e., attention-weighted sum of local feature vectors, can be written as:

$$\boldsymbol{g}_a^{\mathbf{F}} = \Sigma_{i=1}^{i=n} a_i^{\mathbf{F}} \boldsymbol{\ell}_i^{\mathbf{F}}. \tag{5.9}$$

The attentive global feature vector $\boldsymbol{g}_a^{\mathbf{F}}$ replaces the global feature vector $\boldsymbol{g}$ to be used for finding new logits to perform classification. Please note that we employ up to two attention modules of type Att. II in the intermediate feature maps of two different convolutional layers in our DNN-based detector.

Attention Mechanism III: Self-attentional Feature Maps

To further refine the intermediate feature maps of our DNN-based morph detector, we integrate the self-attentional feature maps [23, 24] into our deep architecture, which is shown in Fig. 5.3. (c). The attention augmented convolutional network employs the multi-headed self-attention used

in the vision transformer architecture [87, 88]. In this type of attention, the multi-headed self-attentions are applied on a given intermediate feature map in our DNN which leads to a new set of augmented feature maps. In accordance with [23], concatenation of the convoluted feature maps and the self-attentional feature maps result in the best performance.

Suppose F delineates an intermediate set of feature maps where $F \in \mathbb{R}^{H \times W \times F_{in}}$. The feature maps are reshaped to $F \in \mathbb{R}^{HW \times F_{in}}$. The number of attention heads is represented by N_h and d_v, d_k delineates the depth of values, queries/keys, respectively. Also, the depth of values and queries/keys per attention head are denoted by d_v^h and d_k^h respectively. The input F is mapped into queries, keys, and values through learned weights $W_q \in \mathbb{R}^{F_{in} \times d_k^h}$, $W_k \in \mathbb{R}^{F_{in} \times d_k^h}$, and $W_v \in \mathbb{R}^{F_{in} \times d_v^h}$. The queries, keys, and values are as follows:

$$q = FW_q, k = FW_k, v = FW_v. \tag{5.10}$$

In the multi-head attention setting, the output of the first attention head O_h can be written as:

$$O_h = Softmax(\frac{(FW_q)(FW_k)^T}{\sqrt{d_k^h}})(FW_v). \tag{5.11}$$

More importantly, the output of the multi-head self-attention module (MHA) is denoted as:

$$MHA(F) = [O_1||O_2||....||O_{N_h}]W^O, \tag{5.12}$$

where $||$ represents the concatenation operation and $W^O \in \mathbb{R}^{d_v \times d_v}$ is a matrix of learned weights. The attention augmented feature maps (AAConv.) stem from the concatenation of the conventional feature maps due to the convolution operation (Conv.(F)) and the self-attentional feature maps MHA(F). In other words:

$$AAConv(F) = [Conv.(F)||MHA(F)]. \tag{5.13}$$

5.2.4 Arrangement of Feature Selection Schemes

There are different permutations for employing the group sparsity for channel-wise feature selection as well as three different attention modules Att. I, Att. II, and Att. III. We follow the rules set forth

by the curriculum learning paradigm [134–136] to incrementally incorporate channel-wise and spatial feature selection/refinement modules. The curriculum learning premise highlights the benefits of providing initially easy tasks to a DNN for training purposes and presenting more difficult tasks in later stages which increases the complexity of the network's parameter space. Thus, we first train our wavelet-based DNN which is constrained to the group sparsity constraint. Consequently, we fine-tune our trained DNN using a modified structure that incorporates different attention modules Att. I, Att. II, and Att. III. More importantly, we consider different number of attention modules in Section 5.3.5.

We demonstrate in our experiments, delineated in the following sections, contribution of each attention mechanism to accuracy of our deep morph detector. In other words, our results prove efficacy of the Att. I, Att. II, and Att. III for capturing morphing artifacts.

5.2.5 Training Schedule

To find the most discriminative wavelet sub-bands we first fine-tune our DNN, which is an Inception-ResNet-v1 [8] pretrained on VGGFace2 [137] with a modified loss function integrating weight decay on the parameters of its first convolutional layer, using the input images that have been decomposed into 48 wavelet sub-bands. Due to our 48-channel input data, we change the number of channels in the first layer of the original Inception-ResNet-v1 to 48. There is a hyperparameter λ for the group sparsity which is empirically searched for using the validation set of our data. It is expected that after training, a subset of kernel weights in the first layer are zeroed out, leading to an implicit selection of a subset of wavelet sub-bands.

Once the number of selected sub-bands are obtained, which is the easy task in the context of curriculum learning, we continue fine-tuning our DNN using these selected sub-bands. In other words, we shrink the number of input channels in our DNN, and investigate the effect of adding each individual attention modules Att. I, Att. II, and Att. III. Different convolutional layers are assimilating attention modules to improve accuracy of detecting morphed images.

Table 5.1: Size of our datasets.

Dataset	Morphing tool	Bona fide	Morph
Twin-Landamrk [138]	Landmark-based	9860	14230
Twin-StyleGAN [138]	StyleGAN	1488	450
Twin-Perturbed [138]	Adversarial	1488	1240
FERET [19, 139]	FaceMorpher	1413	529
	StyleGAN2	1413	529
	OpenCV	1413	529
FRLL [19, 140]	OpenCV	204	1221
	FaceMorpher	204	1222
	StyleGAN2	204	1222
	WebMorpher	204	1221
FRGC [19, 141]	FaceMorpher	3038	964
	OpenCV	3038	964
	StyleGAN2	3038	964
Landamrks-I [56, 142]	FaceMorpher	528	800
Landamrks-II [143]	OpenCV	528	941
StyleGAN [26, 28]	StyleGAN	528	941
MIPGAN-II [31]	StyleGAN2	374	747

5.3 Evaluations

5.3.1 Datasets

We utilize the WVU Identical Twin Face Morph dataset [138] which consists of samples generated using four techniques, i.e., (1) Landmark-based face morph generation, (2) StyleGAN-based face morph generation, (3) Wavelet-based face morph generation, and (4) adversarially perturbed face morph generation. From these four morph generation methods, in this study, we use the Landmark-based, StyleGAN-based and adversarially perturbed morphs which are dubbed Twin-Landmark, Twin-StyleGAN, and Twin-Perturbed, respectively. FRLL-Morphs [19, 140], FERET-Morphs [19, 139], and FRGC-Morphs [19, 141] are other datasets we employ in this work. The FRLL-Morphs dataset is built upon the Face Research London Lab dataset using four different face morphing tools: (1) OpenCV [144], (2) FaceMorpher [145], (3) StyleGAN2 [31], and (4) WebMorpher [146]. The FERET-Morphs dataset which is based on the color FERET database are morphed using the (1) OpenCV, (2) FaceMorpher, and (3) StyleGAN2 morphing modules. The FRGC-Morphs dataset is constructed using the (1) OpenCV, (2) FaceMorpher, and (3) StyleGAN2 morphing tools. Size of each deataset is detailed in Table 5.1.

In addition to the above-mentioned datasets, we utilize the datasets employed in the single image morph detection tables of the MIPGAN paper [1], which were all constructed using the FRGC-V2 [141] face database. The datasets we use from the MIPGAN paper are as follows: Landmarks-I [56, 142], Landmarks-II [143], StyleGAN [26, 28], and MIPGAN-II [31].

5.3.2 Experimental Setup and Evaluation Metrics

Our core DNN is the Inception-ResNet-v1 [8]. The Inception-ResNet-v1 layers are as follows: 1) Stem block, 2) Five Inception-resnet-A blocks, 3) Reduction-A block, 4) 10 Inception-resnet-B blocks, 5) Reduction-B block, 6) Five Inception-resnet-C blocks, 7) Average Pooling, 8) Dropout, 9) Softmax. Details of each block used in the Inception-ResNet-v1 can be found in the [8]. However, as mentioned in section 5.2.1, we modify the original DNN architecture to account for the 48-wavelet-sub-band input data where we replace 3-channel RGB filters in the first convolutional layer with 48-channel filters. The number of the channels in the filters of the first convolutional layer of the original Inception-ResNet-v1 deep network is three since natural RGB images have three channels. However, we want to feed 48-channel data. Therefore, we increase the channel size of the filters to 48. Our DNNs are trained using the Adam [103] optimizer for 150 epochs accelerated using two 12 GB TITAN X (Pascal) GPUs. We have trained our DNNs in the PyCharm 2022.3.1 environment using PyTorch libraries in a Ubuntu 20.04.3 operating System. The learning rate is initially set at 0.001 which is divided by 10 every 20 epochs. As for the ArcFace loss function parameters, we set the scaling factor s=64.0 and margin m=0.5 [147].

We have reported our results using the following metrics based on the ISO/IEC 30107-3 [148]: Bona fide Presentation Classification Error Rate (BPCER) which represents proportion of bona fide presentations that are incorrectly classified as attack presentations (morph) by the classifier, Attack Presentation Classification Error Rate (APCER) that is proportion of attack presentations (morph) that are incorrectly classified as bona fide, Detection Equal Error Rate (D-EER) the point where APCER is equal to BPCER, and Area Under Receiver Operating Characteristic Curve (AUROC). In the context of binary classification, considering morph class as the positive class, BPCER and APCER are nothing but the False Positive Rate (FPR) and False Negative Rate (FNR), respectively. Especially, we are interested in the following three thresholds: 1) BPCER @

APCER=5% 2) BPCER @ APCER=10% 3) BPCER @ APCER=30%. AUROC is a threshold independent metric representing a fair evaluation of our learned hypotheses.

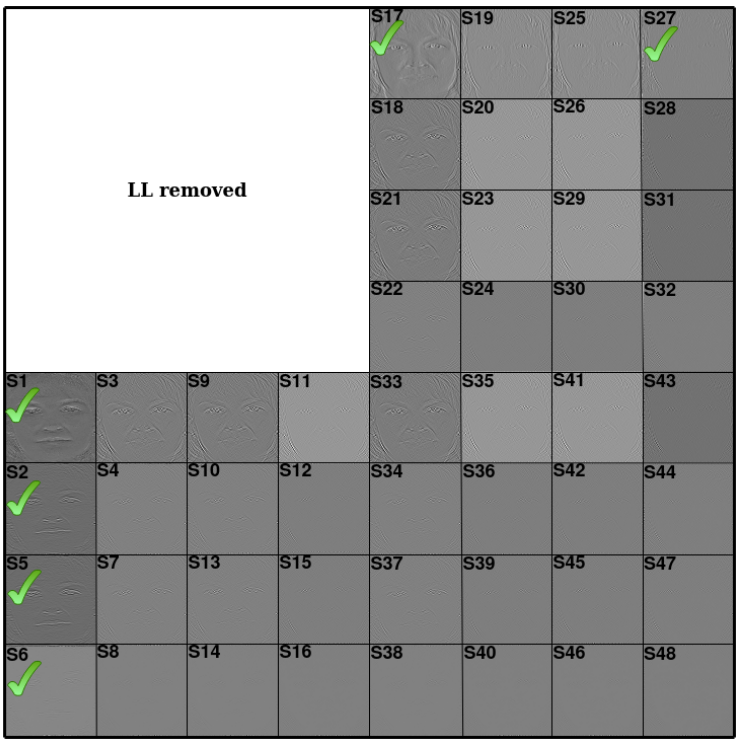


Figure 5.4: 48 wavelet sub-bands are depicted for a given morphed image. Our channel-wise feature selection scheme leads to selection of the six most discriminative wavelet sub-bands which are ticked.

5.3.3 Channel-wise Feature Selection via Group Lasso Weight Decay

In order to select the most discriminative wavelet sub-bands, we first train our DNN using the 48-wavelet-sub-band data using the WVU Twin-Landmark dataset. We do a random search for tuning the hyperparameter λ and we train our DNN for several selected values of hyperparameter λ as mentioned in Eq. 5.2 using the training portion of the Twin-Landmark dataset. We assess the performance of the trained DNNs using the validation portion of the Twin-Landmark dataset, and it is revealed that $\lambda = 0.003$ leads to the highest accuracy on the validation set of the WVU Twin-Landmark dataset retaining six wavelet sub-bands out of 48 as depicted in Fig. 5.4. We further assess the generalization of our trained DNN on all of the datasets. Table 5.2 delineates the benchmarked morph detection results for different datasets when the input samples are either

Table 5.2: Comparing single morph detection performance using the RGB and six wavelet sub-band channels: D-EER%, BPCER@APCER=5%, BPCER@APCER=10%, and BPCER@APCER=30%. Our subband selection has resulted in increasing the accuracy of morph detection as the improved results are highlighted.

Test	Train: Twin-Landmark (RGB)					Train: Twin-Landmark (six wavelet subband)				
	D-EER	5%	10%	30%	AUROC	D-EER	5%	10%	30%	AUROC
Twin-Landmark	5.50	5.60	2.63	0.33	98.73	**2.82**	**1.61**	**1.20**	**0.24**	**99.51**
Twin-StyleGAN	53.52	99.49	82.30	69.08	45.49	**53.5**	**92.00**	**82.11**	**68.30**	**56.14**
Twin-Perturbed	8.57	15.33	6.53	0.98	97.04	12.74	22.09	15.08	3.70	94.20
FERET-FaceMorpher	22.81	40.87	33.27	19.20	87.31	**11.80**	**25.87**	**20.41**	**19.20**	**88.82**
FERET-StyleGAN2	19.81	40.57	28.01	11.92	89.30	**12.66**	**24.38**	**14.36**	**6.80**	**91.82**
FERET-OpenCV	24.57	39.70	31.57	20.42	86.15	**17.90**	**27.03**	**22.11**	**13.90**	**87.80**
FRLL-OpenCV	0.08	0.0	0.0	0.08	99.95	2.94	2.70	1.76	0.98	99.11
FRLL-FaceMorpher	0.16	0.16	0.16	0.16	99.92	0.24	0.16	0.05	0.03	99.93
FRLL-StyleGAN2	4.83	2.82	2.58	0.0	99.06	4.3	3.1	2.00	0.0	97.33
FRLL-WebMorpher	23.52	59.10	36.89	18.13	84.60	**22.67**	**58.34**	**35.55**	**17.06**	**86.9**
FRGC-FaceMorpher	3.50	2.5	1.50	0.0	99.21	**2.07**	**0.41**	**0.20**	**0.0**	**99.50**
FRGC-OpenCV	3.62	2.03	1.49	0.0	99.37	**2.59**	**1.03**	**0.1**	**0.0**	**99.50**
FRGC-StyleGAN2	14.35	25.48	17.45	4.07	94.43	**6.01**	**6.95**	**2.59**	**0.29**	**98.17**

RGB images or the six wavelet sub-band data samples. The results reveal that selecting the top six most discriminative wavelet sub-bands can conspicuously decrease the predicted error rates of our classifier or equivalently increase AUROC on several datasets. In particular, all D-EER, BPCER@APCER=5%, BPCER@APCER=10%, and BPCER@APCER=30% error rates decrease for all morphing types of FERET and FRGC datasets in addition to the Twin-Landmark and Twin-StyleGAN which are highlighted in Table 5.2. In addition, selecting six discriminative sub-bands resulted in an increase of AUROC for all eight datasets. Therefore, our sub-band selection scheme leads to a more accurate morph detector compared to the one trained on the RGB data. Please note that, for the rest of the following experiments and tables, we use the six selected wavelet subbands as the input to our deep morph detector.

5.3.4 Feature Refinement via Attention Mechanisms

We have integrated our three different attention modules after the following layers: 1) "conv2d-3b" where size of the feature maps are $80 \times 126 \times 126$, 2) "conv2d-4b" where size of the feature maps are $256 \times 61 \times 61$, 3) "mixed-7a" where size of the feature maps are $1792 \times 14 \times 14$. To increase the accuracy of our morph detector and to focus on the most discriminative spatial regions, where the density of morphing artifacts is higher, we integrate Att. I module which is an instantiation of the CBAM self-attentional class. This attention module provides us with refined intermediate

Table 5.3: Attention-based single morph detection performance using the six selected wavelet sub-bands and *Att. I* trained on the Twin-Landmark dataset: D-EER%, BPCER@APCER=5%, BPCER@APCER=10%, BPCER@APCER=30%.

Test	***Att. I***@conv2d-3b					***Att. I***@conv2d-4b				
	D-EER	5%	10%	30%	AUROC	D-EER	5%	10%	30%	AUROC
Twin-Landmark	**2.58**	**1.58**	**1.04**	**0.24**	**99.53**	3.62	2.58	0.64	0.0	99.52
Twin-StyleGAN	61.10	96.00	93.70	90.66	34.90	**52.80**	**90.66**	**81.66**	**67.66**	**56.26**
Twin-Perturbed	10.56	17.50	11.45	3.79	95.39	10.16	18.22	10.16	2.58	96.08
FERET-FaceMorpher	20.98	30.43	25.51	15.80	86.18	**11.52**	**25.37**	**19.84**	**12.09**	**89.60**
FERET-StyleGAN2	14.55	26.27	20.03	7.20	91.36	18.52	44.61	33.83	11.15	89.64
FERET-OpenCV	22.11	34.21	29.11	17.00	84.66	**16.79**	**26.72**	**22.08**	**13.74**	**88.39**
FRLL-OpenCV	2.53	1.14	0.24	0.24	98.64	0.16	0.16	0.16	0.03	99.92
FRLL-FaceMorpher	0.49	0.24	0.15	0.12	99.86	**0.16**	**0.16**	**0.16**	**0.11**	**99.93**
FRLL-StyleGAN2	5.20	5.97	3.00	1.90	97.20	12.50	23.81	16.20	2.80	94.62
FRLL-WebMorpher	40.1	67.07	58.80	45.40	65.84	23.3	50.61	42.83	21.00	82.36
FRGC-FaceMorpher	2.59	0.82	0.31	0.23	99.71	2.59	0.93	0.31	0.0	99.54
FRGC-OpenCV	**2.59**	**1.02**	**0.08**	**0.0**	**99.63**	3.52	1.65	0.62	0.25	99.32
FRGC-StyleGAN2	**6.00**	**6.60**	**2.46**	**0.20**	**98.18**	8.60	15.24	6.63	0.51	96.90

activation maps increasing mutual information with respect to the ground truth labels. We fine-tune our augmented Inception-ResNet-v1 by adding a CBAM module separately at two different layers of the Inception-ResNet-v1. We report our attention-based morph detection results in Table 5.3 where a single CBAM module is inserted after the convolutional layers "conv2d-3b" and "conv2d-4b". Our inserted CBAM modules enjoy both the spatial and channel gates as discussed in subsection 5.2.3. The channel attention gate in the CBAM module adopts a multilayer perceptron (MLP) where the size of the hidden layer is $floor(\frac{input-channels}{reduction-ratio})$. In our experiments, we set $reduction-ratio = 16$. The number of channels in the feature maps "conv2d-3b" and "conv2d-4b" are 80 and 256, respectively. Please note that we use the six selected most discriminative wavelet subbands as the input of our deep morph detector. Based on the results benchmarked in Table 5.3, the attention module Att. I results in the refinement of intermediate features leading to a decrease in morph detection error rates as well as an increase in the corresponding AUROC. Improved results compared to Table 5.2 where there was no attention module are highlighted. Adding attention module Att. I has increased morph detection accuracy on several datasets. In particular, employing Att. I has resulted in decreasing error rates when detecting morph images in the Twin-Landmark, Twin-StyleGAN, FERET-FaceMorpher, FERET-OpenCV, FRLL-FaceMorpher, FRGC-OpenCV, and FRGC-StyleGAN2 datasets.

We incorporate the attention mechanism Att. II discussed in Section 5.2.3 into our DNN, to acquire new set(s) of weighted feature vectors. To this end, correlations of spatial locations in an intermediate feature map and the 512-D fully connected (FC) vector before the logits of our DNN

Table 5.4: Attention-based single morph detection performance using the six selected wavelet sub-bands and *Att. II* trained on the Twin-Landmark dataset: D-EER%, BPCER@APCER=5%, BPCER@APCER=10%, BPCER@APCER=30%.

Test	***Att. II*** @conv2d-3b					***Att. II*** @mixed-7a				
	D-EER	5%	10%	30%	AUROC	D-EER	5%	10%	30%	AUROC
Twin-Landmark	3.06	2.09	1.29	0.75	97.42	**2.58**	**1.60**	**0.72**	**0.0**	**99.60**
Twin-StyleGAN	58.22	99.77	98.66	88.00	38.80	51.2	74.1	94.44	91.77	50.47
Twin-Perturbed	12.58	23.06	15.16	4.35	94.28	10.24	16.85	10.24	2.9	95.03
FERET-FaceMorpher	17.58	38.56	25.14	7.18	91.67	20.98	28.92	26.46	14.5	86.75
FERET-StyleGAN2	14.17	27.03	18.33	6.23	92.78	17.39	28.54	22.49	9.8	89.63
FERET-OpenCV	**14.55**	**24.95**	**19.28**	**5.10**	**93.45**	23.80	36.10	32.89	20.7	82.67
FRLL-OpenCV	7.20	9.41	5.48	1.80	97.43	0.16	0.16	0.16	0.04	99.90
FRLL-FaceMorpher	**0.16**	**0.16**	**0.16**	**0.08**	**99.93**	0.24	0.24	0.16	0.0	99.89
FRLL-StyleGAN2	20.21	38.95	29.62	15.95	85.85	17.30	33.79	22.74	13.2	87.32
FRLL-WebMorpher	21.53	53.23	40.54	17.36	85.98	**19.1**	**51.18**	**27.43**	**16.00**	**87.25**
FRGC-FaceMorpher	3.52	2.69	1.45	0.62	99.28	**0.24**	**0.24**	**0.16**	**0.0**	**99.89**
FRGC-OpenCV	3.52	3.11	1.65	0.72	99.22	3.73	3.52	2.48	0.62	99.14
FRGC-StyleGAN2	10.58	16.59	10.78	3.63	96.10	13.58	22.51	18.04	4.14	94.70

Table 5.5: Attention-based single morph detection performance using the six selected wavelet sub-bands and *Att. III* (Self-attentional feature maps) trained on Twin-Landmark dataset: D-EER%, BPCER@APCER=5%, BPCER@APCER=10%, and BPCER@APCER=30%.

Test	***Att. III***@conv2d-3b					***Att. III***@mixed-7a				
	D-EER	5%	10%	30%	AUROC	D-EER	5%	10%	30%	AUROC
Twin-Landmark	6.85	9.11	5.24	1.85	96.42	9.51	14.35	8.95	3.30	96.39
Twin-StyleGAN	59.77	94.44	93.11	84.00	35.55	54.66	93.33	90.00	76.66	43.84
Twin-Perturbed	17.82	43.70	30.56	10.08	88.25	20.56	50.96	38.54	12.41	93.24
FERET-FaceMorpher	**10.65**	**23.78**	**19.73**	**11.64**	**92.50**	23.62	52.36	37.99	18.90	85.48
FERET-StyleGAN2	**12.50**	**22.56**	**13.25**	**5.43**	**92.60**	21.36	44.04	32.70	15.68	86.05
FERET-OpenCV	20.81	30.71	25.43	15.26	88.65	29.67	62.57	50.47	28.54	78.99
FRLL-OpenCV	1.55	0.49	0.16	0.16	98.82	1.44	0.45	0.14	0.13	99.04
FRLL-FaceMorpher	2.53	1.14	0.24	0.24	98.64	2.94	2.70	1.76	0.98	99.11
FRLL-StyleGAN2	16.85	41.32	23.07	8.26	90.17	15.54	29.29	20.94	9.00	90.34
FRLL-WebMorpher	**18.45**	**50.85**	**26.45**	**11.80**	**88.30**	25.63	71.00	58.72	21.37	79.77
FRGC-FaceMorpher	**0.24**	**0.24**	**0.14**	**0.0**	**99.90**	16.59	44.70	23.65	6.22	90.54
FRGC-OpenCV	**2.48**	**0.98**	**0.03**	**0.0**	**99.65**	18.56	53.94	33.42	8.60	87.96
FRGC-StyleGAN2	6.53	7.78	3.93	0.81	76.58	22.82	68.77	48.23	12.65	84.15

are computed. The normalized correlation values decide which pixel locations to remain active for morph detection or which pixels are to be suppressed. Please note that, from an information-theoretic perspective, this kind of attention module looks for the spatial feature locations that have the highest mutual information with respect to the ground truth label Y. The Att. II is inserted after the "conv2d-3b" or "mixed-7a" where the number of channels are respectively 64 and 1,792. Since the number of channels in the feature map and the dimension of the FC layer's output are not consistent, we use a 1×1 convolution to reach 512 channels for the intermediate feature maps. Finally, the attention-weighted feature locations replace the output of the FC layer for finding the two-class logits in our DNN. The results of the morph detection on different datasets using this kind of attention module are summarized in Table 5.4. Based on the benchmarked results in Table 5.4, adopting Att. II has resulted in the improvement of morph detection accuracy for several datasets compared to Table 5.2 where there was not any attention module. In particular, employing Att. II has resulted in decreasing error rates when detecting morph images in the Twin-Landmark, FERET-OpenCV, FRLL-FaceMorpher, FRLL-WebMorpher, and FRGC-FaceMorpher datasets.

We integrate the Att. III module in our DNN. The self-attentional augmented feature maps, detailed in Subsection 5.2.3, are concatenated with the "vanilla" convolutional feature maps to diversify learned features. We assess the effectiveness of this multi-headed self-attention scheme through inserting this attention module at the layers "conv2d-3b" and "mixed-7a" which have respectively 80 and 1,792 feature maps. The results for this kind of attention augmented morph detection are benchmarked in Table 5.5. According to the benchmarked results, incorporating Att. III yields improvement in morph detection accuracy for several datasets compared to Table 5.2 where there was not any attention module. Improved morph detection results are highlighted in Table 5.5. In particular, employing Att. III has resulted in decreasing error rates when detecting morph images in the FERET-FaceMorpher, FERET-StyleGAN2, FRLL-WebMorpher, FRGC-FaceMorpher, and FRGC-OpenCV datasets.

5.3.5 Comparison with the the State-of-the-art

We compare the results of our attention-based morph detector with the results benchmarked in the MIPGAN [1] paper. The methodologies used in the MIPGAN paper are Ensemble Features [65]

and Hybrid Features [66] which are abbreviated as Ensemble and Hybrid respectively in Table 5.6. The ensemble of features method fuses the score level morph detection results using three different feature descriptors, which are LBP, HOG, and BSIF. On the other hand, the Hybrid Features adopts the Laplacian Pyramids using two different image spaces YCbCr and HSV at three different scales where LBP is used to extract features from every sub image. LBP features are fed to a classifier that is Spectral Regression Kernel Discriminant Analysis (SRKDA) and scores for all sub images are fused for morph detection. Attention-based results on the datasets used in the MIPGAN paper are summarized in Table 5.6. Based on the benchmarked results, our attention augmented morph detector has resulted in decrease of the error rates for different Train/Test scenarios. The improved results are highlighted in Table 5.6. In particular, employing Att. I has resulted in decreasing error rates when detecting morph images in the Landmarks-II dataset regardless of the used training set, StyleGAN dataset when our DNN is trained using the Landmarks-I and MIPGAN-II datasets, and MIPGAN-II dataset when our DNN is trained using the Landmarks-I and StyleGAN datasets. In addition, employing Att. II has resulted in decreasing error rates when detecting morph images in the Landmarks-II dataset regardless of the used training set, StyleGAN dataset when our DNN is trained using the MIPGAN-II dataset, and MIPGAN-II dataset when our DNN is trained using the Landmarks-I, Landmarks-II, and StyleGAN datasets. Employing Att. III has resulted in decreasing error rates when detecting morph images in the Landmarks-II dataset when our DNN is trained on the Landmarks-II dataset, StyeleGAN dataset when our DNN is trained on the MIPGAN-II dataset, and MIPGAN-II dataset when our DNN is trained on the StyleGAN dataset.

Also, it is not uncommon for a given travel document issuing/authentication agency to scan a submitted hard copy facial image. To further make our morph detector more realistic and inclusive, we employ the printed and scanned (re-digitized) datasets used in the MIPGAN [1] paper for testing our morphed detectors. The summary of the morph detection performance on the printed and scanned version of the datasets are tabulated in Table 5.7. In accordance with the benchmarked results, our attention augmented morph detector has decreased the detection error rates in several highlighted Train/Test scenarios, which substantiates the efficacy of our wavelet-based attention augmented morph detector. In particular, employing Att. I has resulted in decreasing error rates when detecting morph images in the Landmarks-I dataset when our DNN is trained on the Landmarks-II and MIPGAN-II datasets , Landmarks-II dataset when our DNN is trained using the Landmarks-I and MIPGAN-II datasets, and MIPGAN-II dataset when our DNN is trained using the Landmarks-II dataset. In addition, employing Att. II has resulted in decreasing error rates when detecting morph

Table 5.6: Comparison with the MIPGAN [1] results. Attention-based single morph detection performance using the six selected wavelet sub-bands and *Att. I*, *Att. II*, and *Att. III* modules all @conv2d-3b fine-tuned on the landmarks-I, landmarks-II, StyleGAN, and MIPGAN-II datasets: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Test	MAD	Train: Landmarks-I			Train: Landmarks-II			Train: StyleGAN			Train: MIPGAN-II		
		D-EER	5%	10%	D-EER	5%	10%	D-EER	5%	10%	D-EER	5%	10%
Landmarks-I	Ensemble	0.0	0.0	0.0	0.0	0.0	0.0	0.32	0.0	0.0	13.08	29.15	15.78
	Hybrid	0.16	0.0	0.0	0.16	0.0	0.0	0.42	0.0	0.0	40.14	77.7	67.23
	Att. I	0.0	0.0	0.0	0.0	0.0	0.0	7.95	7.95	4.63	20.5	23.4	21.9
	Att. II	0.65	0.0	0.0	0.66	0.0	0.0	20.1	21.16	20.6	19.1	21.4	20.5
	Att. III	1.98	0.0	0.0	0.66	0.0	0.0	88.07	95.45	96.55	37.8	58.4	55.5
Landmarks-II	Ensemble	49.55	92.22	88.85	3.62	2.22	0.68	44.72	89.53	80.61	32.37	84.9	70.32
	Hybrid	49.16	99.31	97.59	1.53	0.17	0.0	45.65	90.22	84.56	23.88	63.8	45.62
	Att. I	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**8.59**	**10.85**	**5.42**	**22.1**	**26.9**	**21.2**
	Att. II	**0.90**	**0.0**	**0.0**	**0.90**	**0.0**	**0.0**	**23.70**	**28.80**	**27.40**	**23.50**	**29.0**	**27.2**
	Att. III	0.90	0.45	0.45	**0.45**	**0.0**	**0.0**	87.79	94.77	95.78	37.6	56.9	53.4
StyleGAN	Ensemble	0.22	0.0	0.0	29.67	61.92	52.48	0.0	0.0	0.0	12.51	22.29	15.78
	Hybrid	0.16	0.0	0.0	34.76	74.44	62.95	0.0	0.0	0.0	24.7	49.74	41.85
	Att. I	**0.0**	**0.0**	**0.0**	33.84	83.58	65.64	0.0	0.0	0.0	**0.0**	**0.0**	**0.0**
	Att. II	0.65	0.0	0.0	16.92	30.25	18.97	0.0	0.0	0.0	**0.0**	**0.0**	**0.0**
	Att. III	48.52	93.5	88.90	25.12	72.82	57.43	0.0	0.0	0.0	**0.0**	**0.0**	**0.0**
MIPGAN-II	Ensemble	34.13	70.49	61.57	27.13	58.83	45.45	39.93	73.58	66.89	0.0	0.0	0.0
	Hybrid	44.96	83.7	75.47	46.82	85.53	75.81	44.72	82.16	73.75	0.0	0.0	0.0
	Att. I	**25.3**	**80.48**	**68.29**	34.13	70.2	60.85	**0.0**	**0.0**	**0.0**	0.0	0.0	0.0
	Att. II	**17.88**	**52.84**	**29.26**	**12.19**	**23.57**	**13.00**	**0.0**	**0.0**	**0.0**	0.0	0.0	0.0
	Att. III	50.21	93.1	88.4	25.20	73.98	58.53	**0.0**	**0.0**	**0.0**	0.0	0.0	0.0

Table 5.7: Comparison with the MIPGAN print and scanned results. Attention-based single print and scanned morph detection performance using the six selected wavelet sub-bands and *Att. I*, *Att. II*, and *Att. III* modules all @conv2d-3b fine-tuned on the landmarks-I, landmarks-II, and MIPGAN-II datasets: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Test	MAD	Train: Landmarks-I			Train: Landmarks-II			Train: MIPGAN-II		
		D-EER	5%	10%	D-EER	5%	10%	D-EER	5%	10%
Landmarks-I	Ensemble	2.35	1.45	0.96	24.19	52.48	43.22	4.28	3.94	2.22
	Hybrid	1.85	0.85	0.34	32.26	77.87	66.55	5.49	5.48	2.4
	Att. I	48.52	100.0	99.96	**21.50**	**73.47**	**60.61**	**4.21**	**3.33**	**2.2**
	Att. II	49.9	93.3	87.7	52.1	93.8	83.8	48.1	88.5	82.5
	Att. III	48.66	93.8	87.8	53.4	94.9	85.9	49.2	89.7	83.7
Landmarks-II	Ensemble	41.93	81.45	76.25	6.32	7.97	2.42	39.2	90.12	82.32
	Hybrid	44.17	86.48	80.24	5.21	5.19	3.14	40.22	88.9	79.2
	Att. I	**40.51**	**80.05**	**75.96**	22.1	26.9	21.2	**7.50**	**10.02**	**6.56**
	Att. II	**40.50**	**80.4**	**73.1**	49.20	93.2	88.2	**7.20**	**9.44**	**6.3**
	Att. III	**41.56**	**80.8**	**75.4**	50.22	94.6	89.7	**8.56**	**10.55**	**8.45**
MIPGAN-II	Ensemble	5.32	6.68	2.57	33.57	77.35	65.52	0.0	0.0	0.0
	Hybrid	5.90	8.42	3.23	33.91	77.18	65.24	0.0	0.0	0.0
	Att. I	21.55	57.83	39.08	**23.56**	**64.12**	**44.71**	31.39	97.99	94.10
	Att. II	36.6	90.00	81.30	**0.0**	**0.0**	**0.0**	18.3	24.6	19.00
	Att. III	37.5	91.34	85.46	**25.6**	**66.7**	**46.66**	20.78	29.88	22.34

Table 5.8: Generalization performance: comparison with the state-of-the-arts: D-EER%, BPCER@APCER=10%, and APCER@BPCER=10%.

Reference	Train	Testing	D-EER	BPCER-10%
[10, 35]	FERET-FaceMorpher	FRGC-FaceMorpher	19.8	36.4
Ours-*Att. I*	FERET-FaceMorpher	FRGC-FaceMorpher	**1.86**	**0.20**
Ours-*Att. II*	FERET-FaceMorpher	FRGC-FaceMorpher	**4.04**	**1.34**
Ours-*Att. III*	FERET-FaceMorpher	FRGC-FaceMorpher	55.5	97.82
[10, 35]	FERET-OpenCV	FRGC-FaceMorpher	20.1	36.2
Ours-*Att. I*	FERET-OpenCV	FRGC-FaceMorpher	50.51	83.81
Ours-*Att. II*	FERET-OpenCV	FRGC-FaceMorpher	**4.56**	**2.07**
Ours-*Att. III*	FERET-OpenCV	FRGC-FaceMorpher	63.34	90.87
[10, 35]	FERET-FaceMorpher	FRGC-OpenCV	20.7	37.8
Ours-*Att. I*	FERET-FaceMorpher	FRGC-OpenCV	2.59	0.31
Ours-*Att. II*	FERET-FaceMorpher	FRGC-OpenCV	4.77	2.17
Ours-*Att. III*	FERET-FaceMorpher	FRGC-OpenCV	52.3	96.78
[10, 35]	FERET-OpenCV	FRGC-OpenCV	21.1	35.8
Ours-*Att. I*	FERET-OpenCV	FRGC-OpenCV	**2.48**	**1.03**
Ours-*Att. II*	FERET-OpenCV	FRGC-OpenCV	**5.60**	**3.00**
Ours-*Att. III*	FERET-OpenCV	FRGC-OpenCV	42.50	91.18

Table 5.9: Attention-based single morph detection performance using the six selected wavelet sub-bands trained on the Twin-Landmark dataset using two modules of the *Att. I* and *Att. II*: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Test	***Att. I***@conv2d-3b, @conv2d-4b				***Att. II***@conv2d-4b, @mixed-7a			
	D-EER	5%	10%	AUROC	D-EER	5%	10%	AUROC
FERET-FaceMorpher	19.65	41.77	30.24	89.96	21.36	33.08	28.54	86.56
FERET-StyleGAN2	13.61	27.03	18.90	93.83	14.55	22.49	17.39	91.68
FERET-OpenCV	**13.51**	**23.39**	**18.22**	**93.77**	21.1	32.70	28.93	85.21
FRLL-OpenCV	0.40	0.16	0.16	99.86	0.16	0.16	0.02	99.77
FRLL-FaceMorpher	0.81	0.32	0.24	99.83	0.24	0.16	0.01	99.72
FRLL-StyleGAN2	8.67	13.58	7.61	96.79	9.40	19.80	9.1	92.89
FRLL-WebMorpher	**18.50**	**51.05**	**27.03**	**88.32**	30.00	60.6	37.6	76.26
FRGC-FaceMorpher	6.53	8.60	4.14	98.03	**0.24**	**0.13**	**0.0**	**99.91**
FRGC-OpenCV	6.53	8.60	3.83	98.20	**2.67**	**2.01**	**0.9**	**99.86**
FRGC-StyleGAN2	**5.96**	**6.02**	**2.25**	**98.56**	6.54	10.41	7.33	98.08

images in the Landmarks-II dataset when our DNN is trained on the Landmarks-I and MIPGAN-II datasets, and MIPGAN-II dataset when our DNN is trained using the Landmarks-II dataset. Employing Att. III has resulted in decreasing error rates when detecting morph images in the Landmarks-II dataset when our DNN is trained on the Landmarks-I and MIPGAN-II datasets, and MIPGAN-II dataset when our DNN is trained on the Landmarks-II dataset.

We also assess the generalization ability of our framework against the state-of-the-art [10, 35] in Table 5.8 which has assessed morph detection performance on FRGC-FaceMorpher, and FRGC-OpenCV. To this end, we fine-tune our trained Inception-ResNet-v1, including attention modules Att. I, Att. II, and Att. III, on FERET-FaceMorpher and FERET-OpenCV datasets. Please note that, in a PyTorch environment, we freeze all layers' parameters by setting "requires-grade = False" except the final linear classifier layer. Based on the benchmarked results, our wavelet-based attention augmented morph detector surpasses the prior works by a large margin on different train/test scenarios, which are highlighted in Table 5.8. In particular, employing Att. I has resulted in decreasing error rates when detecting morph images in the FRGC-FaceMorpher dataset when our DNN is trained on the FERET-FaceMorpher dataset , FRGC-OpenCV dataset when our DNN is trained using the FERET-OpenCV dataset. In addition, employing Att. II has resulted in decreasing error rates when detecting morph images in the FRGC-FaceMorpher dataset when our DNN is trained on the FERET-FaceMorpher dataset, FRGC-FaceMorpher dataset when our DNN is trained using the FERET-OpenCV, and FRGC-OpenCV dataset when our DNN is trained on the FERET-OpenCV dataset.

We also delve into the different number of attention modules used for training our deep morph detector. We add attention modules Att. I, Att. II, and Att. III to several convolutional layers simultaneously and we benchmark the results for the FERET, FRLL, and FRGC datasets, as shown in Table 5.9 and Table 5.10. Considering the results, having two modules, mainly the Att. I, and Att. II considerably improved the morph detection performance on the FRGC-FaceMorpher, FRGC-OpenCV, and FRGC-StyleGAN2 datasets. In addition, we assess the performance of our morph detector when all three attention modules, Att. I, Att. II, and Att. III, are added to our deep architecture. The resulting performance of this scenario are benchmarked in Table 5.10. Integrating all three attention modules in our DNN has resulted in reduction of detection error rates when assessing morphed images in the FRLL-OpenCV, FRLL-WebMorpher, and FRGC-FaceMorpher datasets. All in all, our wavelet-based attention augmented morph detector has contributed to a

Table 5.10: Attention-based single morph detection performance using the six selected wavelet sub-bands trained on the Twin-Landmark dataset using two modules of the *Att. III* and three modules of the *Att. I*, *Att. II*, and *Att. III*: D-EER%, BPCER@APCER=5%, and BPCER@APCER=10%.

Test	***Att. III***@conv2d-3b, @mixed-7a				***Att. I***@conv2d-3b, ***Att. II***@mixed-7a, ***Att. III***@conv2d-3b			
	D-EER	5%	10%	AUROC	D-EER	5%	10%	AUROC
FERET-FaceMorpher	27.5	69.7	56.1	81.69	21.55	48.55	35.91	87.95
FERET-StyleGAN2	20.79	57.65	44.2	87.02	26.07	64.08	48.20	82.18
FERET-OpenCV	**13.10**	**22.17**	**15.31**	**94.28**	24.57	44.42	36.29	85.58
FRLL-OpenCV	33.57	65.19	59.29	73.05	**0.08**	**0.0**	**0.0**	**99.96**
FRLL-FaceMorpher	23.56	40.67	35.84	85.99	2.53	2.20	1.06	98.83
FRLL-StyleGAN2	26.92	44.27	41.89	81.90	30.52	74.87	66.03	77.16
FRLL-WebMorpher	30.05	53.89	49.22	78.40	**17.65**	**49.54**	**26.77**	**88.80**
FRGC-FaceMorpher	**0.23**	**0.11**	**0.0**	**99.91**	**0.23**	**0.1**	**0.0**	**99.92**
FRGC-OpenCV	30.39	99.89	94.50	73.1	5.70	6.74	4.04	98.51
FRGC-StyleGAN2	19.29	95.64	59.64	84.73	15.56	31.43	21.47	93.37

Table 5.11: Comparison with the NIST FRVT report [2] on the MIPGAN-II dataset: APCER@BPCER = 1%, and APCER@BPCER = 10%.

Algorithm	1%	10%
wvusingle-002 [2, 85]	0.001	0.111
wvusingle-001 [2, 85]	0.015	0.200
visteam-000 [85, 149]	0.323	0.639
unibo-000 [85]	0.037	0.810
Ours	**0.001**	**0.10**

decrease in detection error rate in several highlighted datasets.

Most importantly, we contrast our attention augmented morph detection performance on the MIPGAN-II dataset mentioned in the latest NIST Face Recognition Vendor Test (FRVT) report [2] updated on July 14, 2022. We compare our results with the NIST report using the two criterion APCER@BPCER = 0.01 and APCER@BPCER = 0.1. We report results on the MIPGAN-II dataset while our network with the Att. I is fine-tuned on a universal dataset. Our so-called universal dataset that we use for training our wavelet-based attention augmented morph detector includes all the datasets mentioned in Section 5.3.1 plus the AMSL dataset [4]. The AMSL dataset consists of 2,175 morph and 204 bona fide samples. Results of morph detection on the MIPGAN-II dataset using the NIST report format is summarized in Table 5.11. The benchmarked result delineates the efficiency of our morph detector as the APCER@BPCER = 10 % error rate is decreased.

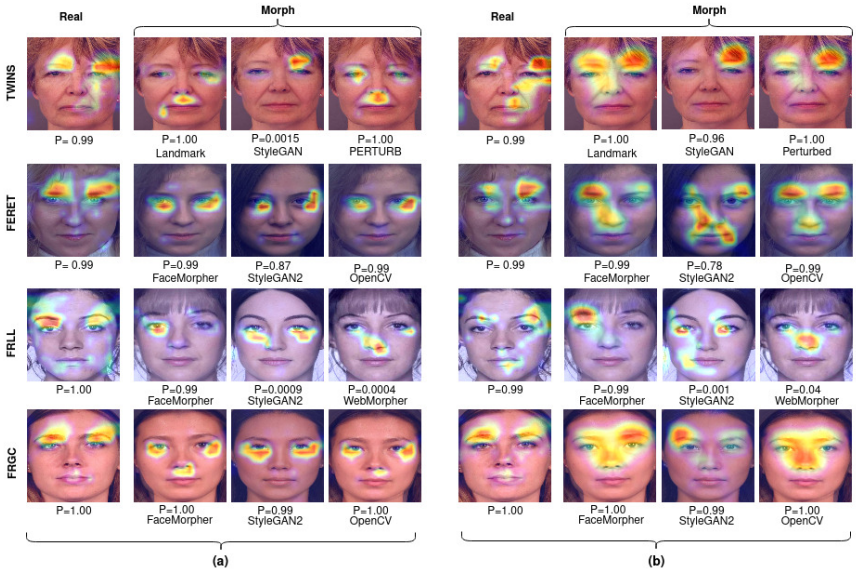


Figure 5.5: Grad-CAM visualizations of CBAM-integrated deep morph detector (Table 5.3): (a) CBAM@conv2d-3b (b) CBAM@conv2d-4b.

5.3.6 Deep Morph Detector Visualization

In this section, the interpretability of our attention-based deep morph detector is investigated through two visualization tools: (1) Attention Maps (2) Gradient-weighted Class Activation Maps (Grad-CAM). Attention maps [20,22,150] are powerful visualization as well as attribution [124,151, 152] techniques which represent a visual explanation for the decision-making of a DNN by highlighting spatial regions that are most relevant for generating output scores by a DNN. In particular, attention maps are obtained by overlaying the heat maps of attention weights into the original RGB images to highlight the most discriminative spatial regions in the eye of a classifier. Grad-CAM is another visualization scheme to demonstrate functionality of our DNN. Given a morphed image, the logits related to the morphed class are supposed to fire which is revealed in the grad-CAM plots. We follow the protocols adopted in the literature [20, 22] corresponding to the Att. I and Att. II modules, which demonstrate efficacy of the CBAM-integrated deep networks through visualizing Grad-CAMs and plotting attention maps for the adjusted network used in [20].

The Grad-CAMs pertinent to Table 5.3 for both convolutional layer of "conv2d-3b" and "conv2d-4b" are shown in Fig. 5.5. Moreover, the estimated attention maps of Table 5.4 for the "mixed-7a" convolutional layer are displayed in Fig. 5.6. As expected, the most discriminative spatial regions

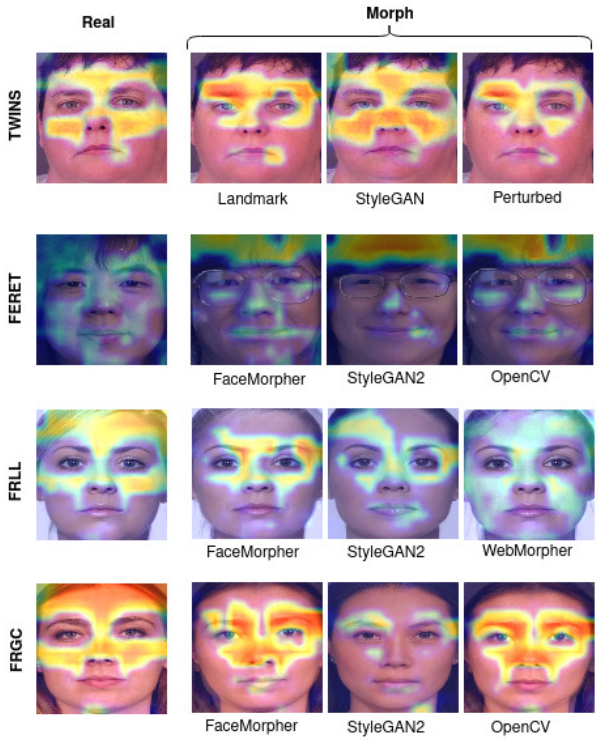


Figure 5.6: Estimated attention maps stemming from the feature maps of "mixed-7a" convolutional layer (Table 5.4).

in the view of a morph detector are in the vicinity of a subject's eyes.

5.4 Conclusion

This chapter addressed single image morphing attack detection where emphasizing on discriminative regions is realized through spatial and channel attention modules. In particular, we quantitatively demonstrated the efficacy of the three visual attention modules for the downstream task of morph detection in a binary classification setting. The integrated attention modules are intended for feature refinement as well as feature selection as a kind of representation learning. In particular, a trainable soft attention mechanism, convolutional block attention module, and multi-headed attention-augmented feature maps were utilized to improve accuracy of morph detection on several datasets. In addition, we have shifted the input data domain from the RGB space into the wavelet domain to take advantage of fine-grained spatial-frequency information represented through wavelet

decomposition.

Our benchmarked results on morph detection using several datasets proves effectiveness of our attention-based morph detector. Most importantly, we have contrasted the generalization performance of our attention augmented morph detection scheme with the state-of-the-art results to demonstrate efficacy of our proposed architectures. Moreover, estimated attention maps and Grad-CAM visualizations were included to demonstrate interpretability of our morph detector. Heatmaps applied on the original images reveal the most discriminative spatial regions of the images that drive our attention augmented morph detectors into an accurate decision for labeling probe images as bona fide or morphed. Finally, to realize multi-attentional morph detection, we assessed our morph detection performance using two instantiations of our attention modules Att. I, Att. II, and Att. III. In addition, we trained our attention augmented morph detector using all three Att. I, Att. II, and Att. III attention modules and the corresponding results were benchmarked in the table mentioned in Section 5.3.5.

Chapter 6

Conclusion and Future Works

6.1 Conclusion

In this dissertation, several frameworks have been proposed to detect morphed face images. We have harnessed the insightful information provided by frequency domain through wavelet decomposition to improve detection of morphed face images. In the first proposed morph detection methodology, I proposed to employ mismatches between entropy distributions of real and morph subbands. Low-low subband was discarded due to existence of morphing artifacts in the high-frequency spectra, which resulted in 48 total subbands. The more mismatch between entropy distributions of real and morph subands, the more discriminative the subband is. Optimal number of selected subbands are selected in accordance with the morph detection accuracy on the validation set. Once optimal number of subbands are found, a deep morph detector is trained. In the second methodology of wavelet-based morph detection, I have used an attention mechanism to find the compatibility score between feature maps' spatial locations and the fully connected layer of the employed Deep Neural Network (DNN). As the third method, I incorporated structured group sparsity to select the most discriminative subbands where loss function of our DNN includes a groups sparsity term on the grouped weights of the first convolutional layer. My fourth framework integrated three types of attention mechanisms, Convolutional Block Attention Mechanism (CBAM), a trainable end-to-end attention finding correlation of output fully connected features and a feature map's spatial pixels , and Self-attentional feature maps to improve accuracy of

morph detection.

6.2 Future Works

To envision my future research, I plan to approach morph detection using other state-of-the-art methodologies such as the vision transformers and its variants which have appeared to be groundbreaking for classification tasks . In addition, I have been developing novel frameworks to detect DeepFake images which is another significant open problem encountered in social medias. DeepFakes are threatening integrity of posted online contents which can propagate false information affecting society attitude.

References

[1] H. Zhang, S. Venkatesh, R. Ramachandra, K. Raja, N. Damer, and C. Busch, "MIPGAN—Generating strong and high quality morphing attacks using identity prior driven GAN," IEEE Transactions on Biometrics, Behavior, and Identity Science, vol. 3, no. 3, pp. 365–383, 2021.

[2] `https://pages.nist.gov/frvt/reports/morph/frvt_morph_report.pdf`. Accessed: Aug 1, 2022.

[3] P. Hancock, "Psychological image collection at stirling (pics)–2d face sets–utrecht ecvp," 2017.

[4] T. Neubert, A. Makrushin, M. Hildebrandt, C. Kraetzer, and J. Dittmann, "Extended stirtrace benchmarking of biometric and forensic qualities of morphed face images," IET Biometrics, vol. 7, no. 4, pp. 325–332, 2018.

[5] A. Makrushin, T. Neubert, and J. Dittmann, "Automatic generation and detection of visually faultless facial morphs.," in VISIGRAPP (6: VISAPP), pp. 39–50, 2017.

[6] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in Proceedings of International Conference on Computer Vision (ICCV), December 2015.

[7] N. Damer, A. M. Saladié, A. Braun, and A. Kuijper, "Morgan: Recognition vulnerability and attack detectability of face morphing attacks created by generative adversarial network," in 2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS), pp. 1–10, IEEE, 2018.

[8] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in Thirty-first AAAI conference on artificial intelligence, 2017.

[9] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. K. Jain, "On the detection of digital face manipulation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern recognition, pp. 5781–5790, 2020.

[10] M. Hamza, S. Tehsin, H. Karamti, and N. S. Alghamdi, "Generation and detection of face morphing attacks," IEEE Access, 2022.

[11] R. Raghavendra and G. Li, "Multimodality for reliable single image based face morphing attack detection," IEEE Access, vol. 10, pp. 82418–82433, 2022.

[12] R. Ramachandra, K. Raja, and C. Busch, "Algorithmic fairness in face morphing attack detection," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 410–418, 2022.

[13] M. Ferrara, A. Franco, and D. Maltoni, "The magic passport," in IEEE International Joint Conference on Biometrics, pp. 1–7, IEEE, 2014.

[14] U. Scherhag, L. Debiasi, C. Rathgeb, C. Busch, and A. Uhl, "Detection of face morphing attacks based on prnu analysis," IEEE Transactions on Biometrics, Behavior, and Identity Science, vol. 1, no. 4, pp. 302–317, 2019.

[15] L.-B. Zhang, F. Peng, and M. Long, "Face morphing detection using fourier spectrum of sensor pattern noise," in 2018 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6, IEEE, 2018.

[16] C. Seibold, A. Hilsmann, and P. Eisert, "Reflection analysis for face morphing attack detection," in 2018 26th European Signal Processing Conference (EUSIPCO), pp. 1022–1026, IEEE, 2018.

[17] L. Debiasi, C. Rathgeb, U. Scherhag, A. Uhl, and C. Busch, "Prnu variance analysis for morphed face image detection," in 2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS), pp. 1–9, IEEE, 2018.

[18] C. Seibold, W. Samek, A. Hilsmann, and P. Eisert, "Accurate and robust neural networks for security related applications exampled by face morphing attacks," arXiv preprint arXiv:1806.04265, 2018.

[19] E. Sarkar, P. Korshunov, L. Colbois, and S. Marcel, "Vulnerability analysis of face morphing attacks from landmarks and generative adversarial networks," arXiv preprint arXiv:2012.05344, 2020.

[20] S. Jetley, N. A. Lord, N. Lee, and P. H. Torr, "Learn to pay attention," arXiv preprint arXiv:1804.02391, 2018.

[21] W. Wen, C. Wu, Y. Wang, Y. Chen, and H. Li, "Learning structured sparsity in deep neural networks," in Advances in neural information processing systems, pp. 2074–2082, 2016.

[22] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," in Proceedings of the European conference on computer vision (ECCV), pp. 3–19, 2018.

[23] I. Bello, B. Zoph, A. Vaswani, J. Shlens, and Q. V. Le, "Attention augmented convolutional networks," in Proceedings of the IEEE/CVF international conference on computer vision, pp. 3286–3295, 2019.

[24] `https://github.com/leaderj1001/Attention-Augmented-Conv2d`. Accessed: March 15, 2022.

[25] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1867–1874, 2014.

[26] S. Venkatesh, H. Zhang, R. Ramachandra, K. Raja, N. Damer, and C. Busch, "Can gan generated morphs threaten face recognition systems equally as landmark based morphs? - vulnerability and detection," in 2020 8th International Workshop on Biometrics and Forensics (IWBF), pp. 1–6, 2020.

[27] S. Soleymani, A. Dabouei, F. Taherkhani, J. Dawson, and N. M. Nasrabadi, "Mutual information maximization on disentangled representations for differential morph detection," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1731–1741, 2021.

[28] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4401–4410, 2019.

[29] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in Advances in neural information processing systems, pp. 2672–2680, 2014.

[30] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," arXiv preprint arXiv:1312.6114, 2013.

[31] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8110–8119, 2020.

[32] R. Abdal, Y. Qin, and P. Wonka, "Image2stylegan: How to embed images into the stylegan latent space?," in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4432–4441, 2019.

[33] N. Damer, K. Raja, M. Süßmilch, S. Venkatesh, F. Boutros, M. Fang, F. Kirchbuchner, R. Ramachandra, and A. Kuijper, "ReGenMorph: Visibly realistic GAN generated face morphing attacks by attack re-generation," arXiv preprint arXiv:2108.09130, 2021.

[34] S. Venkatesh, R. Ramachandra, K. Raja, and C. Busch, "Face morphing attack generation & detection: A comprehensive survey," IEEE Transactions on Technology and Society, 2021.

[35] U. Scherhag, C. Rathgeb, J. Merkle, R. Breithaupt, and C. Busch, "Face recognition systems under morphing attacks: A survey," IEEE Access, vol. 7, pp. 23012–23026, 2019.

[36] U. Scherhag, C. Rathgeb, J. Merkle, and C. Busch, "Deep face representations for differential morphing attack detection," arXiv preprint arXiv:2001.01202, 2020.

[37] S. Soleymani, B. Chaudhary, A. Dabouei, J. Dawson, and N. M. Nasrabadi, "Differential morphed face detection using deep Siamese networks," in International Conference on Pattern Recognition, pp. 560–572, Springer, 2021.

[38] B. Chaudhary, P. Aghdaie, S. Soleymani, J. Dawson, and N. M. Nasrabadi, "Differential morph face detection using discriminative wavelet sub-bands," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1425–1434, 2021.

[39] U. Scherhag, D. Budhrani, M. Gomez-Barrero, and C. Busch, "Detecting morphed face images using facial landmarks," in International Conference on Image and Signal Processing, pp. 444–452, Springer, 2018.

[40] U. Scherhag, R. Raghavendra, K. B. Raja, M. Gomez-Barrero, C. Rathgeb, and C. Busch, "On the vulnerability of face recognition systems towards morphed face attacks," in 2017 5th International Workshop on Biometrics and Forensics (IWBF), pp. 1–6, 2017.

[41] L. Spreeuwers, M. Schils, and R. Veldhuis, "Towards robust evaluation of face morphing detection," in 2018 26th European Signal Processing Conference (EUSIPCO), pp. 1027–1031, IEEE, 2018.

[42] U. Scherhag, C. Rathgeb, and C. Busch, "Performance variation of morphed face image detection algorithms across different datasets," in 2018 International Workshop on Biometrics and Forensics (IWBF), pp. 1–6, 2018.

[43] M. Ferrara, A. Franco, and D. Maltoni, "Face morphing detection in the presence of printing/scanning and heterogeneous image sources," arXiv preprint arXiv:1901.08811, 2019.

[44] S. Autherith and C. Pasquini, "Detecting morphing attacks through face geometry features," Journal of Imaging, vol. 6, no. 11, p. 115, 2020.

[45] U. Scherhag, C. Rathgeb, and C. Busch, "Towards detection of morphed face images in electronic travel documents," in 2018 13th IAPR International Workshop on Document Analysis Systems (DAS), pp. 187–192, IEEE, 2018.

[46] U. Scherhag, C. Rathgeb, and C. Busch, "Morph deterction from single face image: A multi-algorithm fusion approach," in Proceedings of the 2018 2nd International Conference on Biometric Engineering and Applications, pp. 6–12, 2018.

[47] L. Debiasi, N. Damer, A. M. Saladié, C. Rathgeb, U. Scherhag, C. Busch, F. Kirchbuchner, and A. Uhl, "On the detection of gan-based face morphs using established morph detectors," in International Conference on Image Analysis and Processing, pp. 345–356, Springer, 2019.

[48] R. Raghavendra, K. B. Raja, and C. Busch, "Detecting morphed face images," in 2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS), pp. 1–7, 2016.

[49] R. Raghavendra, K. Raja, S. Venkatesh, and C. Busch, "Face morphing versus face averaging: Vulnerability and detection," in 2017 IEEE International Joint Conference on Biometrics (IJCB), pp. 555–563, 2017.

[50] U. Scherhag, J. Kunze, C. Rathgeb, and C. Busch, "Face morph detection for unknown morphing algorithms and image sources: a multi-scale block local binary pattern fusion approach," IET Biometrics, vol. 9, no. 6, pp. 278–289, 2020.

[51] R. Raghavendra, K. B. Raja, S. Venkatesh, and C. Busch, "Transferable deep-cnn features for detecting digital and print-scanned morphed face images," in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1822–1830, IEEE, 2017.

[52] K. Raja, S. Venkatesh, R. Christoph Busch, et al., "Transferable deep-cnn features for detecting digital and print-scanned morphed face images," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 10–18, 2017.

[53] C. Seibold, W. Samek, A. Hilsmann, and P. Eisert, "Detection of face morphing attacks by deep learning," in International Workshop on Digital Watermarking, pp. 107–120, Springer, 2017.

[54] Y. Lu, K. Xu, T. Sun, K. Qi, and L. Yao, "Face morphing detection with convolutional neural network based on multi-features," in Proceedings of the 2020 International Conference on Aviation Safety and Information Technology, pp. 611–616, 2020.

[55] M. Ferrara, A. Franco, and D. Maltoni, "Face demorphing," IEEE Transactions on Information Forensics and Security, vol. 13, no. 4, pp. 1008–1017, 2017.

[56] R. Raghavendra, K. Raja, S. Venkatesh, and C. Busch, "Face morphing versus face averaging: Vulnerability and detection," in 2017 IEEE International Joint Conference on Biometrics (IJCB), pp. 555–563, IEEE, 2017.

[57] N. Damer, A. M. Saladie, S. Zienert, Y. Wainakh, P. Terhörst, F. Kirchbuchner, and A. Kuijper, "To detect or not to detect: The right faces to morph," in 2019 International Conference on Biometrics (ICB), pp. 1–8, IEEE, 2019.

[58] S. Venkatesh, R. Ramachandra, K. Raja, L. Spreeuwers, R. Veldhuis, and C. Busch, "Morphed face detection based on deep color residual noise," in 2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA), pp. 1–6, 2019.

[59] S. Venkatesh, R. Ramachandra, K. Raja, L. Spreeuwers, R. Veldhuis, and C. Busch, "Detecting morphed face attacks using residual noise from deep multi-scale context aggregation network," in The IEEE Winter Conference on Applications of Computer Vision, pp. 280–289, 2020.

[60] N. Damer, N. Spiller, M. Fang, F. Boutros, F. Kirchbuchner, and A. Kuijper, "Pw-mad: Pixel-wise supervision for generalized face morphing attack detection," in International Symposium on Visual Computing, pp. 291–304, Springer, 2021.

[61] L.-B. Zhang, J. Cai, F. Peng, and M. Long, "MSA-CNN: Face morphing detection via a multiple scales attention convolutional neural network," in International Workshop on Digital Watermarking, pp. 17–31, Springer, 2021.

[62] L. Qin, F. Peng, and M. Long, "Face morphing attack detection and localization based on feature-wise supervision," IEEE Transactions on Information Forensics and Security, 2022.

[63] Z. Blasingame and C. Liu, "Leveraging adversarial learning for the detection of morphing attacks," in 2021 IEEE International Joint Conference on Biometrics (IJCB), pp. 1–8, IEEE, 2021.

[64] K. Raja, G. Gupta, S. Venkatesh, R. Ramachandra, and C. Busch, "Towards generalized morphing attack detection by learning residuals," Image and Vision Computing, vol. 126, p. 104535, 2022.

[65] S. Venkatesh, R. Ramachandra, K. Raja, and C. Busch, "Single image face morphing attack detection using ensemble of features," in 2020 IEEE 23rd International Conference on Information Fusion (FUSION), pp. 1–6, IEEE, 2020.

[66] R. Ramachandra, S. Venkatesh, K. Raja, and C. Busch, "Towards making morphing attack detection robust using hybrid scale-space colour texture features," in 2019 IEEE 5th International Conference on Identity, Security, and Behavior Analysis (ISBA), pp. 1–8, 2019.

[67] C. Seibold, W. Samek, A. Hilsmann, and P. Eisert, "Accurate and robust neural networks for face morphing attack detection," Journal of Information Security and Applications, vol. 53, p. 102526, 2020.

[68] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang, "A survey of sparse representation: algorithms and applications," IEEE access, vol. 3, pp. 490–530, 2015.

[69] Y. Xu, Z. Zhong, J. Yang, J. You, and D. Zhang, "A new discriminative sparse representation method for robust face recognition via L_2 regularization," IEEE transactions on neural networks and learning systems, vol. 28, no. 10, pp. 2233–2242, 2016.

[70] Q. Feng, C. Yuan, J.-S. Pan, J.-F. Yang, Y.-T. Chou, Y. Zhou, and W. Li, "Superimposed sparse parameter classifiers for face recognition," IEEE transactions on cybernetics, vol. 47, no. 2, pp. 378–390, 2016.

[71] P. Görgel and A. Simsek, "Face recognition via deep stacked denoising sparse autoencoders (DSDSA)," Applied Mathematics and Computation, vol. 355, pp. 325–342, 2019.

[72] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," Journal of the Royal Statistical Society: Series B (Statistical Methodology), vol. 68, no. 1, pp. 49–67, 2006.

[73] S. Zhang, J. Huang, H. Li, and D. N. Metaxas, "Automatic image annotation and retrieval using group sparsity," IEEE transactions on systems, man, and cybernetics, Part B (Cybernetics), vol. 42, no. 3, pp. 838–849, 2012.

[74] S. Xiang, X. Tong, and J. Ye, "Efficient sparse group feature selection via nonconvex optimization," in International Conference on Machine Learning, pp. 284–292, PMLR, 2013.

[75] L. Yang, C. Gao, D. Meng, and L. Jiang, "A novel group-sparsity-optimization-based feature selection model for complex interaction recognition," in Asian Conference on Computer Vision, pp. 508–521, Springer, 2014.

[76] J. Gui, Z. Sun, S. Ji, D. Tao, and T. Tan, "Feature selection based on structured sparsity: A comprehensive study," IEEE transactions on neural networks and learning systems, vol. 28, no. 7, pp. 1490–1507, 2016.

[77] R. Tibshirani, "Regression shrinkage and selection via the lasso," Journal of the Royal Statistical Society: Series B (Methodological), vol. 58, no. 1, pp. 267–288, 1996.

[78] X. Chen, N. Mishra, M. Rohaninejad, and P. Abbeel, "Pixelsnail: An improved autoregressive generative model," in International Conference on Machine Learning, pp. 864–872, PMLR, 2018.

[79] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in International conference on machine learning, pp. 2048–2057, PMLR, 2015.

[80] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," Machine learning, vol. 8, no. 3-4, pp. 229–256, 1992.

[81] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," arXiv preprint arXiv:1406.6247, 2014.

[82] H. Zhao, J. Jia, and V. Koltun, "Exploring self-attention for image recognition," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10076–10085, 2020.

[83] F. Wu, A. Fan, A. Baevski, Y. N. Dauphin, and M. Auli, "Pay less attention with lightweight and dynamic convolutions," arXiv preprint arXiv:1901.10430, 2019.

[84] P. Ramachandran, N. Parmar, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens, "Stand-alone self-attention in vision models," arXiv preprint arXiv:1906.05909, 2019.

[85] P. Aghdaie, B. Chaudhary, S. Soleymani, J. Dawson, and N. M. Nasrabadi, "Attention aware wavelet-based detection of morphed face images," in 2021 IEEE International Joint Conference on Biometrics (IJCB), pp. 1–8, IEEE, 2021.

[86] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," Advances in neural information processing systems, vol. 12, 1999.

[87] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in Advances in neural information processing systems, pp. 5998–6008, 2017.

[88] 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.

[89] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022, 2021.

[90] Y. Chen, Y. Kalantidis, J. Li, S. Yan, and J. Feng, "Aˆ 2-nets: Double attention networks," Advances in neural information processing systems, vol. 31, 2018.

[91] J. Hu, L. Shen, S. Albanie, G. Sun, and A. Vedaldi, "Gather-Excite: Exploiting feature context in convolutional neural networks," Advances in neural information processing systems, vol. 31, 2018.

[92] J. Kannala and E. Rahtu, "Bsif: Binarized statistical image features," in Proceedings of the 21st international conference on pattern recognition (ICPR2012), pp. 1363–1366, IEEE, 2012.

[93] D. G. Lowe, "Object recognition from local scale-invariant features," in Proceedings of the seventh IEEE international conference on computer vision, vol. 2, pp. 1150–1157, Ieee, 1999.

[94] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in European conference on computer vision, pp. 404–417, Springer, 2006.

[95] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," in Proceedings of 12th International Conference on Pattern Recognition, vol. 1, pp. 582–585, IEEE, 1994.

[96] "International organization for standardization. iso/iec dis 30107-3:2016: Information technology biometric presenta- tion attack detection part 3: Testing and reporting," Standard, 2017.

[97] S. Kullback and R. A. Leibler, "On information and sufficiency," The annals of mathematical statistics, vol. 22, no. 1, pp. 79–86, 1951.

[98] C. E. Shannon, "A mathematical theory of communication," Bell system technical journal, vol. 27, no. 3, pp. 379–423, 1948.

[99] D.-T. Lee and B. J. Schachter, "Two algorithms for constructing a delaunay triangulation," International Journal of Computer & Information Sciences, vol. 9, no. 3, pp. 219–242, 1980.

[100] 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, pp. 770–778, 2016.

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

[102] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in International Conference on Automatic Face and Gesture Recognition, 2018.

[103] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[104] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2921–2929, 2016.

[105] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," Journal of machine learning research, vol. 9, no. Nov, pp. 2579–2605, 2008.

[106] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," arXiv preprint arXiv:1412.6572, 2014.

[107] A. Kurakin, I. Goodfellow, S. Bengio, et al., "Adversarial examples in the physical world," 2016.

[108] M. Ngan, P. Grother, K. Hanaoka, and J. Kuo, "Face recognition vendor test (frvt) part 4: Morph performance of automated face morph detection," National Institute of Technology (NIST), Tech. Rep. NISTIR, vol. 8292, 2020.

[109] H. Fukui, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Attention branch network: Learning of attention mechanism for visual explanation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10705–10714, 2019.

[110] Y. Li, J. Zeng, S. Shan, and X. Chen, "Occlusion aware facial expression recognition using cnn with attention mechanism," IEEE Transactions on Image Processing, vol. 28, no. 5, pp. 2439–2450, 2018.

[111] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3156–3164, 2017.

[112] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499–1503, 2016.

[113] Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, "Thinking in frequency: Face forgery detection by mining frequency-aware clues," in European Conference on Computer Vision, pp. 86–103, Springer, 2020.

[114] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face x-ray for more general face forgery detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5001–5010, 2020.

[115] H. Zhao, W. Zhou, D. Chen, T. Wei, W. Zhang, and N. Yu, "Multi-attentional deepfake detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2185–2194, 2021.

[116] L. Li, J. Bao, H. Yang, D. Chen, and F. Wen, "Advancing high fidelity identity swapping for forgery detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5074–5083, 2020.

[117] T. Neubert, C. Kraetzer, and J. Dittmann, "A face morphing detection concept with a frequency and a spatial domain feature space for images on emrtd," in Proceedings of the ACM Workshop on Information Hiding and Multimedia Security, pp. 95–100, 2019.

[118] J. M. Singh, R. Ramachandra, K. B. Raja, and C. Busch, "Robust morph-detection at automated border control gate using deep decomposed 3d shape & diffuse reflectance," in 2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), pp. 106–112, IEEE, 2019.

[119] S. J. Nightingale, S. Agarwal, and H. Farid, "Perceptual and computational detection of face morphing," Journal of Vision, vol. 21, no. 3, pp. 4–4, 2021.

[120] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," IEEE transactions on pattern analysis and machine intelligence, vol. 35, no. 8, pp. 1798–1828, 2013.

[121] T. Hu, H. Qi, Q. Huang, and Y. Lu, "See better before looking closer: Weakly supervised data augmentation network for fine-grained visual classification," arXiv preprint arXiv:1901.09891, 2019.

[122] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in Proceedings of International Conference on Computer Vision (ICCV), December 2015.

[123] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne.," Journal of machine learning research, vol. 9, no. 11, 2008.

[124] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in Proceedings of the IEEE international conference on computer vision, pp. 618–626, 2017.

[125] P. Aghdaie, B. Chaudhary, S. Soleymani, J. Dawson, and N. M. Nasrabadi, "Detection of morphed face images using discriminative wavelet sub-bands," 2021.

[126] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," IEEE transactions on pattern analysis and machine intelligence, vol. 35, no. 11, pp. 2765–2781, 2013.

[127] Y. Li, S. Gu, C. Mayer, L. V. Gool, and R. Timofte, "Group sparsity: The hinge between filter pruning and decomposition for network compression," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8018–8027, 2020.

[128] X. Zhang, T. Wang, J. Qi, H. Lu, and G. Wang, "Progressive attention guided recurrent network for salient object detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 714–722, 2018.

[129] H. Zheng, J. Fu, Z.-J. Zha, J. Luo, and T. Mei, "Learning rich part hierarchies with progressive attention networks for fine-grained image recognition," IEEE Transactions on Image Processing, vol. 29, pp. 476–488, 2019.

[130] N. Tishby and N. Zaslavsky, "Deep learning and the information bottleneck principle," in 2015 IEEE Information Theory Workshop (ITW), pp. 1–5, IEEE, 2015.

[131] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4690–4699, 2019.

[132] `https://github.com/Jongchan/attention-module`. Accessed: Feb 10, 2022.

[133] `https://github.com/SaoYan/LearnToPayAttention`. Accessed: March 15, 2022.

[134] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in Proceedings of the 26th annual international conference on machine learning, pp. 41–48, 2009.

[135] G. Hacohen and D. Weinshall, "On the power of curriculum learning in training deep networks," in International Conference on Machine Learning, pp. 2535–2544, PMLR, 2019.

[136] A. Graves, M. G. Bellemare, J. Menick, R. Munos, and K. Kavukcuoglu, "Automated curriculum learning for neural networks," in international conference on machine learning, pp. 1311–1320, PMLR, 2017.

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

[138] `https://biic.wvu.edu/data-sets/identical-twin-face-morph-dataset`. Accessed: Sep 10, 2022.

[139] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," Image and vision computing, vol. 16, no. 5, pp. 295–306, 1998.

[140] L. DeBruine and B. Jones, "Face Research Lab London Set," May 2017.

[141] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the face recognition grand challenge," in 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol. 1, pp. 947–954, IEEE, 2005.

[142] "Face morph using openCV. http: //www.learnopencv.com/ face-morph-using-opencv-Cpp-Python/, 2017.,"

[143] M. Ferrara, A. Franco, and D. Maltoni, "Decoupling texture blending and shape warping in face morphing," in 2019 International Conference of the Biometrics Special Interest Group (BIOSIG), pp. 1–5, IEEE, 2019.

[144] "Satya mallick, "face morph using OpenCV — C++ / Python," march 2016,"

[145] "Alyssa quek, "FaceMorpher," january 2019,"

[146] "Lisa DeBruine, "debruine/webmorph: Beta release 2," jan. 2018.,"

[147] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5265–5274, 2018.

[148] "ISO/IEC 30107-3:2017." `https://www.iso.org/obp/ui/#iso:std:iso-iec:30107:-3:ed-1:v1:en`. Accessed: DEC 28, 2022.

[149] I. Medvedev, F. Shadmand, and N. Gonçalves, "Mordeephy: Face morphing detection via fused classification," arXiv preprint arXiv:2208.03110, 2022.

[150] Y. Lu, W. Zhang, C. Jin, and X. Xue, "Learning attention map from images," in 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1067–1074, IEEE, 2012.

[151] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in European conference on computer vision, pp. 818–833, Springer, 2014.

[152] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "SmoothGrad: removing noise by adding noise," arXiv preprint arXiv:1706.03825, 2017.