Quadrant-Based Bi-level Self-Attention Feature Extraction for Face Morphing Attack Detection

1. Introduction

Face morphing attack is a kind of security breach facing facial biometric systems. Face morphing is the creation of a blended image from two or more different people facial images such that any of the constituent faces can be verified with the blended morphed mage. If such blended images are successfully used for verification or authentication under a biometric verification system, then face morphing attack is regarded to have taken place [1], [2], [3] This attack can lead to serious security breach potentially exposing an entire country to unprecedented risks from criminals and terrorist groups. This threat arises if a criminal or terrorist manages to obtain an identity document featuring a morphed image that combines their facial features with those of a non-criminal accomplice. While the document, such as an e-passport, would display the accomplice's name, the morphed image could allow the criminal to authenticate their identity at an automated border control (ABC) system. This would grant them unauthorized access to the country, enabling them to carry out criminal activities [4], [5]. Due to the threat posed by this face morphing attack, researchers have been developing techniques for detecting the attack; with the detection technique framed as either a single-image [6] or differential-image [7] detection approaches. The former involves detecting whether a particular given image is morphed or not, while the latter is applicable in the context of face verification systems, where two images are used—one as a probe and the other as a reference image. The aim is to match the probe to the reference to determine how closely they correspond within a threshold for acceptance or rejection of the probe as a match to the reference. This method is often used to prevent the false verification of morphed images in the context of a face recognition system.

The two problem formulations have been addressed using approaches from classical computer vision and in recent time deep learning approaches due to their promising performance on various tasks. However, these morphing detection methods often rely on global features or uniform attention mechanisms, which may miss subtle artifacts localized in specific regions of the face. Thus, there is a critical need for detection models that can identify localized morphing artifacts effectively. Existing methods may not capture subtle regional anomalies or variations in face texture and features introduced by morphing.

We introduce Quadrant-Based Bi-level Self-Attention Feature Extraction (QBSAF), a novel model that applies quadrant-based attention to enhance face morphing detection. It divides the face image into quadrants, applies localized attention mechanisms to each quadrant, and integrates features across regions. This approach improves detection accuracy and provides better interpretability of model decisions by highlighting regions of interest in the face image. The contributions of this approach are:

1. **Localized Attention Mechanism**: Unlike traditional methods that apply global or uniform attention mechanisms, QBSAF divides the face image into quadrants and applies attention independently to each quadrant. This localized attention helps in detecting subtle morphing artifacts that might be missed by methods focusing on the entire image.
2. **Contextual Integration of Local and Global Features**: QBSAF integrates features from both local quadrants and the global context of the face. This dual approach allows the model to capture both regional and holistic information, improving the detection of morphing artifacts that may appear at various scales and locations.

3. Multi-Scale Attention Fusion: The model employs multi-scale attention mechanisms within each quadrant, allowing it to detect morphing artifacts at different resolutions. This capability enhances the model's sensitivity to fine-grained details and subtle variations introduced by face morphing.

2.0 Related Work

2.1 Face Morphing Techniques

Various morphing techniques exist for creating seamless morph of facial images smoothly. Point-based morphing establishes corresponding points on the source images and warps the images based on interpolated points [7], [8]. Triangle-based morphing subdivides the images into triangles, interpolates triangle vertices, and warps the images accordingly[9]. Mesh-based morphing divides the images into grids or meshes, interpolates vertices, and performs image warping. Optical flow-based morphing estimates dense motion between images and applies it for smooth transitions. Image-based morphing analyses and transforms image content using techniques like registration, blending, and texture synthesis. Model-based morphing employs predefined 3D models or shapes to gradually transform between images. The choice of morphing technique depends on factors such as image characteristics and desired control, and combinations or advanced algorithms can be used for more sophisticated results. Recently, generative adversarial network based morphing method has been proposed[10], [11]. All these techniques are either utilized in software tools such FantaMorph[12] WebMorph[13] among others or in library like Opencv[14]. The plethora of different morphing approaches makes the task of morph image detection challenging due to the variability in morphing techniques, the high quality and realism of morphed images, subtle differences between morphed and genuine images, and the need for models to generalize across diverse conditions and domains [7], [11], [15], [16].

2.2 Morphing Detection Methods

The morphing detection methods are broadly categorized under **single image detection** and **differential image morph detection**. Single image detection methods are akin to developing a binary image classification model that is trained to classify face images as either morphed or real. In contrast, differential morph detection is akin to face verification, recognition, or face matching, where a model compares a probe image to a reference image to determine whether the probe is a real match to the reference or a fake[16], [17]

All the techniques for morph detection techniques can be broadly classified into three: traditional computer vision approach, deep learning approach and hybrid approach.

2.2.1 Traditional Computer Vision Approaches

In the literature of image detection with traditional approaches researchers have used image feature extraction algorithms based on texture descriptors, keypoint descriptors, edge-gradient descriptors, frequency and wavelet descriptors, statistical descriptors among others. These techniques are either used individually or as a fusion of two or more.

The texture descriptors such as Local Binary Patterns (LBP), Local Phase Quantization (LPQ), Gabor Filters, Steerable Filters and BSIF (Binarized Statistical Image Features) are used to extract the surface properties or patterns. Authors in [18] utilized a steerable pyramid which decomposes an image into multiple scales to extract image features from their print scan dataset at different resolutions and allowing analysis on fine details to coarse structures. Kenneth *et al.* [3] utilized LBP feature descriptor with Decision Tree classifier while [19] also utilized a fusion of different configurations of multi-scale block local binary patterns. The same LBP features was

also evaluated as part of other methods like SIFT, SURF, LPQ, HOG and BSIF in [15], [18], [20]. Similarly, BSIF features with SVM classifier was primarily used by Raghavendra *et al.* [21].

Keypoint descriptors which are used for detecting and describing significant points in an image have also being used extensively, they include SIFT (Scale-Invariant Feature Transform), SURF (Speeded-Up Robust Features) , ORB (Oriented FAST and Rotated BRIEF), BRIEF (Binary Robust Independent Elementary Features), FAST (Features from Accelerated Segment Test). SURF and SIFT features descriptors in conjunction with SVM classifier were evaluated as part of other features techniques for morph detection in [15]. An ensemble of features across texture descriptors, Keypoint extractors and gradient estimators have also been proposed in [6], [22].

The key limitations of these traditional methods is that the features extracted are either from local regions or only global regions and they do not capture both local and global features at once. To achieve a model with local and global features systematic ensembles of different features is needed.

2.2.2 Deep learning Approaches

In the deep learning approach we identified the work earliest work of Seibold et al. [23] performed face morphing attacks using deep convolutional neural networks. Three CNN architectures were trained from scratch and with pre-trained weights. The pre-trained networks outperformed those trained from scratch in all cases. The use of deep learning feature embedding of images for images pairing selection for morphing and detection of resulted morphed images was investigated in [24]. Convolutional Neural network based demorphing network was employed in [25] to discover the initial pictures from two images that has been morphed. By unraveling the constituent's images in a morphed image the method was able to detect morphing attack at ABC gate scenario. In another research work presented in [26], the use proposed neural network training schemes, which are based on different alternations of the training data, to increase robustness and generality of morphing attack detection model. Siamese network based VGG-16 architecture was introduced in [27] while deep representation from a pretrained vanilla vision transformer neural network was used with SVM for single image morph detection in [28].

2.2.3 Hybrid and Other Approaches

Hybrid approaches include approaches that combined traditional methods with deep learning methods. Authors in [24], [25] combined traditional approach with deep earning approach by utilizing wavelet sub bands features computed on images as input to spatial attention mechanisms over convolutional and feed forward network to create a morphing detection system. Our approach compared with this approach is devoid of preliminary feature extraction with traditional methods which increases the processing steps and our attention mechanism has low resource overhead compared to the coupled attention mechanism used in [24], [25].

In another work [31], a method utilizing two-stream network with channel attention and residual of multiple color spaces is proposed for face morphing detection. The method first obtains H, S, V, Y, Cb, Cr six color channel image, then use the bilateral filter for filtering the six color channel to get the corresponding residual noise image, then the combined six channel image and the residual noise image as input to the two-stream network for training. Other approaches for morphed detection include those utilizing image forensics analyses as presented by Hildebrandt et al. [7] Nuebert et al. [8].

The key limitation of these existing approaches is their lack of explicit pipeline to handle localized morphing artifacts.

2.3 Attention Mechanisms

Attention mechanism in neural networks is a framework that allows the model to focus selectively on certain parts of the input data, assigning varying levels of importance to different tokens or elements during the learning process. This mechanism is particularly useful in sequence-based tasks like natural language processing (NLP) and computer vision, as it helps capture long-range dependencies by giving greater weight to more relevant parts of the input. It has become very prevalent in computer vision tasks, improving the performance of vision models.

The foundational work by Vaswani et al. [32] Introduced the self-attention mechanism, which allows models to consider the relationships between all elements in the input sequence. This innovation demonstrated that attention could effectively capture long-range dependencies in data, making it applicable not only in natural language processing but also in visual tasks. In vision, self-attention allows the model to evaluate the importance of each feature in relation to others, leading to a more nuanced understanding of the image [33], [34], [35], [36], [37].

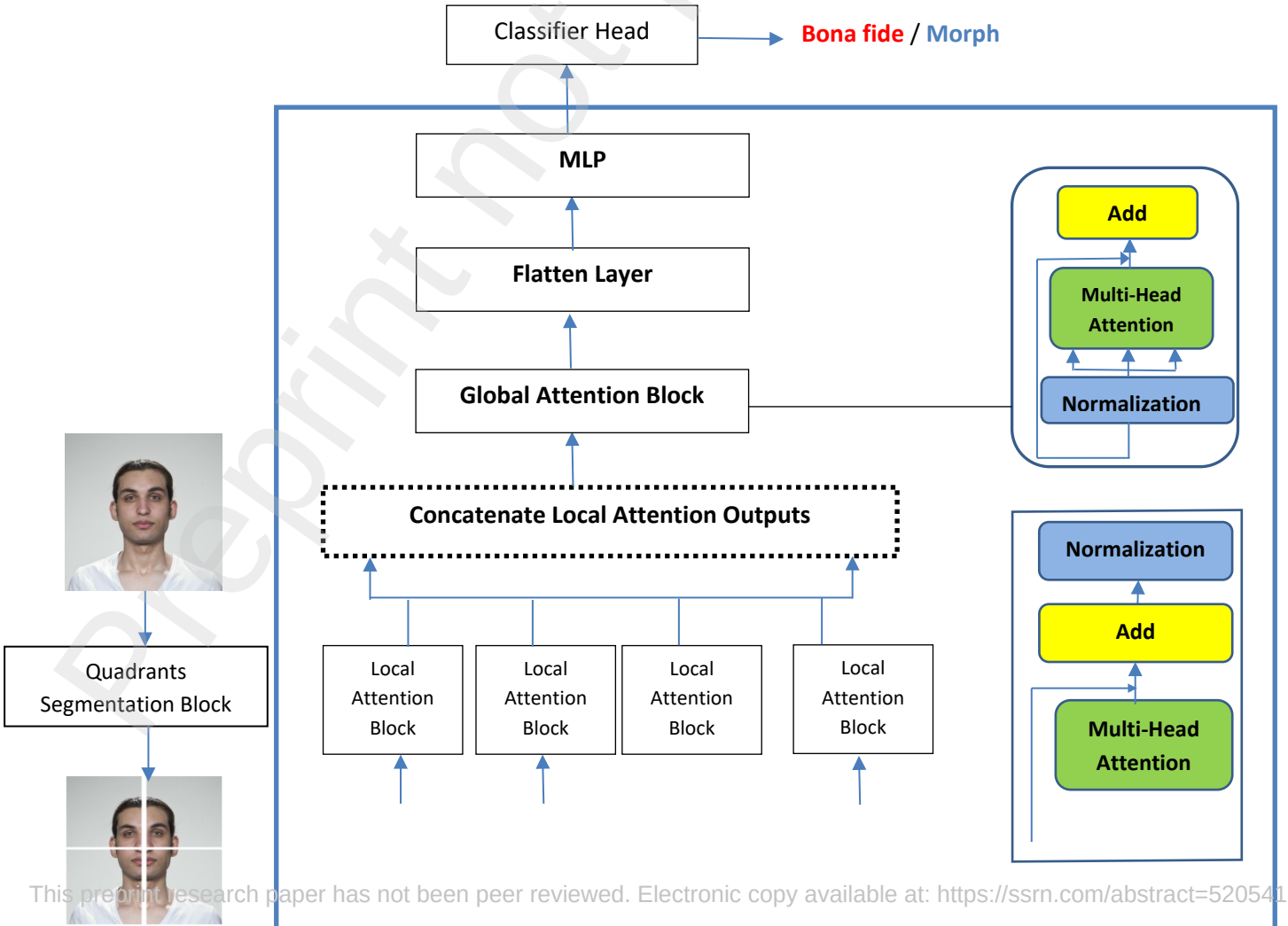
Attention mechanisms can be categorized into local and global attention. Local attention focuses on specific regions of the image, enhancing the model's ability to detect fine details and anomalies. In contrast, global attention assesses the entire image, capturing overall relationships and contextual information. Vision Transformers (ViTs) by Dosovitskiy et al. [38] showcases the power of global attention in visual tasks. By treating image patches as input tokens, ViTs utilize self-attention to capture contextual dependencies across the entire image. This approach has led to state-of-the-art performance in several benchmark tasks, underscoring the significance of attention in effectively modeling visual data. Similarly, attention mechanisms contribute to improved feature representation by emphasizing the most salient aspects of an image. The Convolutional Block Attention Module (CBAM) proposed by Woo et al.[39] incorporates both channel and spatial attention, allowing models to refine feature maps by focusing on important features while ignoring irrelevant information. This dual attention mechanism enhances the model's ability to detect critical features in images, significantly improving performance in tasks such as object detection and segmentation. Furthermore, attention mechanisms have the ability to enhance the interpretability of models. Attention weights can be visualized, providing insights into which areas of an image the model considers important for its predictions. This transparency is particularly valuable in applications such as face morphing and forensic analysis, where understanding the rationale behind model decisions is critical [40].

However, our proposed dual-stage self-attention framework for detecting morphing artifacts in images distinguishes itself from existing methods, such as Vision Transformers (ViTs) and Convolutional Block Attention Modules (CBAM). The proposed QBSAF method begins with a unique segmentation of images into four quadrants. This localized approach allows the framework to focus on specific regions of the image independently, capturing local contextual relationships before integrating information across quadrants. ViTs process the entire image as a whole, dividing it into non-overlapping patches. While they also capture contextual relationships, they do so without explicitly segmenting the image into regions with distinct attention mechanisms for local and global contexts. CBAM enhances feature maps through a sequential application of channel and spatial attention. It operates on the feature maps produced by convolutional layers, focusing on both channel-wise and spatial aspects without a defined segmentation approach.

The processing pipeline of our approach is distinct, with a structured approach where self-attention is first applied locally to quadrants, followed by a global attention. This stepwise methodology enhances both local and global feature representations for morphing detection. Also, the processing pipeline in ViTs is generally linear, with the entire image being processed in one step. This can result in the loss of local feature significance during global contextualization while CBAM integrates attention mechanisms within existing architectures, applying attention sequentially rather than employing a structured dual-stage analysis, which limits its ability to capture distinct local-global feature interactions.

3. Proposed Method

This section outlines the methodology employed in our dual-stage self-attention framework for detecting morphing artifacts in images. The approach consists of several key steps, including image quadrants segmentation, initial self-attention processing, combination of quadrant outputs, and a second application of self-attention on the combined representation. The overview of our proposed method is depicted in Figure 1. The proposed model framework composed of several key components that can be structurally grouped into a dual stage steps consisting of segmentation of input image into four segments, local attention computation on each segment separately at the low level pixels without need for further computational overhead of feature extraction. Performing feature extraction on the image will defeat the main role of deep learning which is meant to eliminate the need for feature engineering that is an ethos in the traditional machine learning. Thus, our model does not use feature engineering steps before passing the model to the local attention layer of the network. The second stage of the model entails concatenating the outputs of the four local attentions. Each of the output from the local attention is a transformed representation of the original input that captures the contextual information from other elements in the segment based on their importance or contribution to the meaning of that segment.



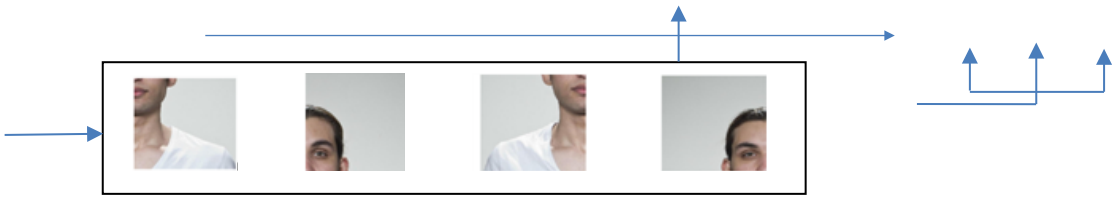


Figure 1 Overview of Proposed Quadrant-Based Bi-level Feature Extraction Model

The concatenated image restored the image to its orderly position eliminating the need of using positional encoding in the segmented images. After concatenation, another global attention computation is also performed on the concatenated image thereby capturing contextual information not just from each segment but from the entire unit, reflecting how each color interacts with every other color across the combined segments.

3.1 Image Quadrant Segmentation

Given an input face image I of size $H \times W \times C$, the image is segmented into four segments: I_1, I_2, I_3, I_4 *with each segment* I_i *is of size* $H/2 \times \mathrm{W}/2 \times \mathrm{C}$ where the H is the height of the image, W is the width and C is the number of channels. Our model directly processed the RGB channels images without any feature engineering step thereby reducing the computational overhead of the model. Since different parts of a face contain distinct and important features (e.g., eyes, nose, mouth), by dividing an image into quadrants, our model can focus on learning fine-grained, localized features specific to each part of the face leading to identification of subtle local morphing artefact.

3.2 Quadrant Local Self-Attention Block

For each quadrant I_i a multi-head self-attention mechanism is applied to capture local features. Each quadrant feature map $X_i \in R^{H/2 \times \mathrm{W}/2 \times \mathrm{C}}$ is transformed into Query (Q) , Key (K) and Value(V) matrices of the multi-head attention. In our model we used h=4 as one of the hyper parameters of the model. The matrices are obtained according to equations 1, 2,3.

$$\boldsymbol{Q_i^h} = \boldsymbol{X_i W_Q^h} \quad (1)$$

$$\boldsymbol{K_i^h} = \boldsymbol{X_i W_k^h} \quad (2)$$

$$\boldsymbol{V_i^h} = \boldsymbol{X_i W_v^h} \quad (3)$$

where $\boldsymbol{W_Q^h}$, $\boldsymbol{W_k^h}$, $\boldsymbol{W_v^h}$ are learnable weights for head h_k.

Scaled dot-product attention for head h is calculated as:

$$\boldsymbol{Attention}(\boldsymbol{Q_i^h}, \boldsymbol{K_i^h}, \mathrm{V_i^h}) = \boldsymbol{softmax}\left(\frac{\boldsymbol{Q_i^h}(\boldsymbol{K_i^h})^T}{\sqrt{\boldsymbol{d_h}}}\right) \mathrm{V_i^h} \quad (4)$$

$$X_{i\_attended\_quandrant} = \boldsymbol{MultiHead\_Output} = \mathrm{Concat}(head_1, \ldots, head_H) W_o \quad (5)$$

For each output from each local self-attention block, we incorporate a skip connection, where the input of the layer is element-wise added to its output before proceeding to the normalization layer. This operation, defined as:

$$X_i = X_{i\_attended\_quandrant} + X_i \quad (6)$$

where $X_{i\_attended\_quandrant}$ represents the transformations obtained from the multi-head local self-attention block and the X_i is the input to the block. This enables the model to learn residual

mappings that ease optimization and improve generalization performance. After this layer to stabilize and improve training, we applied a normalization layer to the skip connection output, which ensures that the activations remain within a controlled range across training iterations. This addresses internal covariate shift, a situation whereby the distribution of activations changes as training progresses, which can hinder optimization. By normalizing the output, we enable the model to learn more robust representations by reducing sensitivity to variations in input distribution, leading to improved convergence and model performance.

3.2 Concatenation and Global Multi-Head Attention

The second stage of the dual stage process of our proposed method consists of concatenation of the output from the local attention block described in Section 3.1 into a single unit and the application of a global attention to the concatenated output. Each of the four local attention blocks transformed their input quadrant into a new representation capturing the local relationship among the features in that quadrant. The output from each local attention block is then combined into a single unit in such a way to have the same dimension as the original input image to the model. Quadrant outputs are concatenated to form $X_{combined}$ according to equation 7.

$$X_{combined} = \cup_{i=1}^{4}(X_i) \quad (7)$$

This combined single unit representation X_{comb} is then fed to the global attention block to gain global contextual insights and inter-quadrant relationships using the self-attention block. The self-attention operation is performed according to equation 8-11.

$$Q_{comb} = X_{comb} W_Q \quad (8)$$

$$K_{comb} = X_{comb} W_K \quad (9)$$

$$V_{comb} = X_{comb} W_V \quad (10)$$

WhereW_Q , W_K and W_V are learnable weights for to transform the X_{comb} into Query, Key and Value matrices: Q_{comb}, K_{comb} and V_{comb} respectively.
The scale-dot product attention is then computed according to equation 11.

$$Global\ Attention(Q_{comb}, K_{comb}, V_{comb}) = \boldsymbol{s}\text{oftmax}(\frac{Q_{comb}(K_{comb})^T}{\sqrt{\boldsymbol{d_h}}})\ V_{comb} \quad (11)$$

This global attention enables the vectors in the single unit images to incorporate knowledge from the entire four quadrants, understanding how each pixel relates to others across all quadrants, rather than just locally. It reflects relationships between pixels that were previously separated by quadrants. The complex patterns and features that emerged from the global attention on the entire unit allow for richer representations that facilitate the discerning of morphed images from unmorphed ones.

3.3 Flatten, MLP and Classification Layers

The output of the global attention block is passed to the flatten layer. The flatten operation is applied to the output of the global attention layer to enable compatibility with subsequent fully connected and classification layers. The global attention outputs a 3D tensor of shape (H,W,C) representing spatial dimensions and channels, which must be transformed into a 1D vector for subsequent processing. Flattening aggregates information across spatial and channel dimensions,

compressing it into a format optimal for higher-level decision-making and pattern recognition. Additionally, this operation reduces dimensional complexity, enhancing computational efficiency while preserving the salient features learned through attention.

Let the output of the global attention layer be a 3D tensor $X \in R^{H\times W\times C}$ where: H and W represent the spatial height and width, and is the number of channels. The flatten operation f reshapes X into a 1D vector $X_{flat} \in R^{H\cdot W\cdot C}$

$$X_{flat} = f(X) = reshape(X, (H \cdot W \cdot C,1)) \quad (12)$$

Here, $reshape$ is the operation that takes each element of X and arranges it into a vector of size H×W×C. This flattened form, X_{flat} , is then compatible with the subsequent fully connected layers for higher-level feature integration and classification.

After this step, a Multi-Layer Perceptron (MLP) network is applied to the flattened output of the global attention layer to introduce non-linear transformations that enhance feature representation and capture meaningful discernible patterns which is then fed to the classification unit head. The MLP layer is made of N dense layers having M units in each layer with each unit having a GELU activation function. Each layer is then followed with dropout layer to prevent overfitting by randomly deactivating a subset of neurons during training, which helps the model generalize better. The N and M are other hyper-parameters of our model.
After flattening the attention output $X \in R^{H\times W\times C}$ into a 1D vector denoted as serves as input to the MLP, the operation of the MLP is described according to the equations 13-

$$H^{(0)} = X_{flat} \quad \in R^{H\cdot W\cdot C} \quad (13)$$

For each dense layer l $(where\ l = 1,2,\ldots,N)$, the output is computed as in Eqn 14:

$$H^{(l)} = Dropout(\sigma(\ W^{(l)}H^{(l-1)} + b^{(l)})) \quad (14)$$

where:

$W^{(l)} \in R^{M\times d_{l-1}} is\ the\ weight\ matrix\ for\ the\ \ l\text{-}th\ layer$
$b^{(l)} \in R^{M}\ is\ the\ bias\ vector\ for\ the\ l\text{-}th\ layer$
$\sigma\ represents\ the\ GELU\ activation\ function.$

The dropout layer that follows each feed forward layer is computed as:

$$Dropout(H^{(l)}) = \begin{cases} \frac{H^{(l)}}{1-p} & with\ probability\ 1-\ p \\ 0 & with\ probability\ p \end{cases} \quad (15)$$

where p is the dropout rate.

The classification unit, positioned after the Multi-Layer Perceptron (MLP), is designed for binary classification to determine if an input image is morphed or authentic. The unit takes the output from the last MLP layer, denoted as $H^{(N)}$ which contains the refined feature representations. Afterwards, the classification is performed using a single dense layer with computation and sigmoid activation function in Equation 16 and 17 . The sigmoid activation function is applied to produce the output probability:

$$y_{logits} = W^{(c)}H^{(N)} + \ b_c \quad (16)$$

where $W^{(c)} \in$ is the weight vector of the classification layer and b_c is the bias. The activation function is applied to y_{logits} as:

$$y = \sigma(y_{logits}) = \frac{1}{1 + e^{-y_{logits}}} \quad (17)$$

The output y represents the probability that the input image is morphed, ranging from 0 to 1.

After the model is designed as described above, it then trained with the morphed datasets with the training configurations that include the loss function, number of epochs, batch size for the training and validation datasets among other hyperparameters. The details of this training and the experiments conducted on different datasets are presented in the next section.
The entire process in the model development is depicted in Algorithm QBSAF.

Algorithm QBSAF

Input: $D = \{ I\_x, \acute{y}\}$ *where* $I_x \in R^{H \times W \times C}$ $\acute{y} \in \{0,1\}$

Output: $f(D) \rightarrow \{0,1\}$ // a trained model

for each batch of image I in I_x {

$I_1 = I\left[0 : \frac{\mathrm{H}}{2}, 0 : \frac{\mathrm{W}}{2}, : \right]$

$I_2 = I\left[0 : \frac{\mathrm{H}}{2}, \frac{W}{2} : \mathrm{W}, : \right]$

$I_3 = I\left[\frac{\mathrm{H}}{2} : \mathrm{H}, 0 : \frac{\mathrm{W}}{2}, : \right]$

$I_4 = I\left[\frac{\mathrm{H}}{2} : \mathrm{H}, \frac{\mathrm{W}}{2} : \mathrm{W}, : \right]$

//Local Attention

$for\ each\ quandrant\ I_i with\ feature\ map\ X_i\ \ in\ [I_1, I_2, I_3, I_4]$ {

$\boldsymbol{Q_i^h = X_i W_Q^h}$

$\boldsymbol{K_i^h = X_i W_k^h}$

$\boldsymbol{V_i^h = X_i W_v^h}$

$\boldsymbol{Attention}(\boldsymbol{Q_i^h}, \boldsymbol{K_i^h}, \mathrm{V}_i^{\boldsymbol{h}}) = \boldsymbol{softmax}\left(\frac{\boldsymbol{Q_i^h (K_i^h)^T}}{\sqrt{\boldsymbol{d_h}}}\right) \mathrm{V}_i^{\boldsymbol{h}}$

$\boldsymbol{MultiHead\_Output} = \mathrm{Concat}(head_1, \ldots, head_H) W_o$

}

//Quadrant outputs are concatenated to form $\boldsymbol{X_{comb}}$

$X_{comb} = \cup_{i=1}^{4} (X_i)$

$Q_{comb} = X_{comb} W_Q$

$K_{comb} = X_{comb} W_K$

$V_{comb} = X_{comb} W_V$

//Global Attention

$X = Global\ Attention(Q_{comd}, K_{comb}, V_{comb}) = \boldsymbol{s}\mathrm{oftmax}\left(\frac{Q_{comb} (K_{comb})^T}{\sqrt{\boldsymbol{d_h}}}\right) V_{comb}$

//Flatten , MLP and Classification Layer

$X_{flat} = f(X) = reshape(X, (H \cdot W \cdot C, 1))$

$H^{(0)} = X_{flat} \quad \in R^{H \cdot W \cdot C}$

$for\ l = 1\ to\ N$ // N number layers in MLP

$H^{(l)} = Dropout(\sigma(W^{(l)} H^{(l-1)} + b^{(l)}))$

$y_{logits} = W^{(c)} H^{(N)} + b_c$

$y = \sigma(y_{logits}) = \frac{1}{1 + e^{-y_{logits}}}$

}

$f(D) = Loss\ (y, \acute{y})$

$retrun\ f(D)$

4. Evaluation

4.1 Datasets

Two categories of datasets were employed in the experiments. The first were the real bonafide un-morphed image datasets. These dataset include the Utrecht [41] , Face lab London [42], Basel **Face Database (BFD)** [43] and **Chicago Face Database (CFD)**[44] The second categories of datasets were morphed images which are AMSL Face Morph Image Dataset [7] and FRLL morph dataset [10]. Since our face morphing formulation was based on single image morphed detection, we setup the experiments in such a way that any image can be used as real images and morphed images. This setup makes the detection to be more generalized and work under different scenarios.

Specifically, the bona fide images are collected from the following sources:

- **Basel Face Database (BFD)** : This dataset consist of 40 face images of 18 male and 22 female. The individuals look directly towards the camera with a neutral, relaxed facial expression (Details | Basel Face Database)
- **Chicago Face Database (CFD) :** 826 neutral face images and an additional 150 happy, closed-mouth expressions (totalling 980 images) of 597 unique individuals including self-identified Asian, Black, Latino, and White female and male models (CFD | Chicago Face Database)[42].
- **Face Research Lab London Set:** Images are of 102 adult faces 1350x1350 pixels in full colour. The faces were natural non-smiling faces.
- **Utrecht ECVP**: Subset of the Psychological Image Collection at Stirling (PICS) image datasets[41]. It consists of 131 images, 49 men, 20 women, usually a neutral and smile of each, collected at the European Conference on Visual Perception in Utrecht, 2008

The morphed datasets were the following:

- **FRLL-Morphs** is a dataset of morphed faces based on images selected from the publicly available Face Research London Lab dataset. The dataset is made of four different morphed datasets made from using different morphing tools.

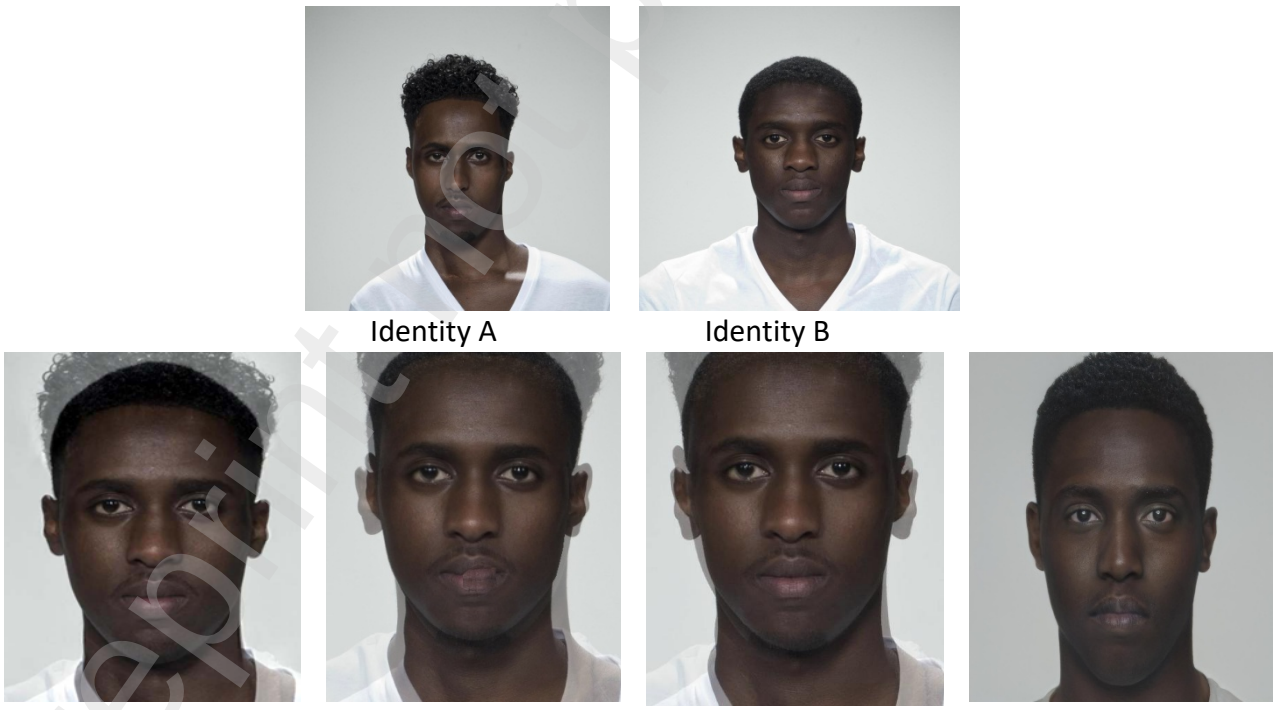


Figure 2: Morphing Data from Four Different Morphing Tools for Identity A and B

- **AMSL Face Morph Image** Data Set was created based on images from the Face Research Lab London Set. The morphed dataset was created with post-processed operations.

Dataset Name	Type	Sample Size
Basel Face Database (BFD	Bonafide	40
Chicago Face Database (CFD)	Bonafide	826
Face Research Lab London Set	Bonafide	102
Utrecht ECVP	Bonafide	131
FRLL-Morphs- OpenCV	Morphed	1221
FRLL-Morphs- FaceMorpher	Morphed	1222
FRLL-Morphs- StyleGAN	Morphed	1222
FRLL-Morphs- WebMorpher	Morphed	1221
AMSL Face Morph	Morphed	2175

4.2 Evaluation Metrics

In this study, eight performance metrics were used to comprehensively evaluate the model's performance. The key performance metrics include precision, recall, F1-score, accuracy, area under the curve (AUC), and attack-specific metrics: APCER (Attack Presentation Classification Error Rate), BPCER (Bona Fide Presentation Classification Error Rate), and EER (Equal Error Rate).

1. **Key Performance metrics:**
   - **Precision:** Reflects the proportion of predicted attacks that are actually correct.
   - **Recall:** Measures the model's ability to detect actual attack samples among all real attacks.
   - **F1-Score:** Combines precision and recall into a single metric, balancing their trade-offs.
   - **Accuracy:** Measures the model's overall ability to correctly predict classes.
   - **AUC (Area Under the ROC Curve):** Indicates the model's ability to distinguish between attack and bona fide samples across all threshold values.
2. **Attack-Specific Metrics**
   - **APCER (Attack Presentation Classification Error Rate):** Proportion of attack samples incorrectly classified as bona fide.
   - **BPCER (Bona Fide Presentation Classification Error Rate):** Proportion of bona fide samples misclassified as attacks.
   - **DEER (Detection Equal Error Rate):** The rate where the proportion of attacks misclassified as bona fide equals the proportion of bona fide samples misclassified as attacks.

4.3 Model Training and Testing

4.3.1 Within Dataset Testing

Four different models were trained using four different morph datasets- Facemorpher, OpenCV Morphs, StyleGAN Morphs, and WebMorpher from the FRLL morph dataset. The dataset contains morphed images created using four different morphing tools and approaches. Each model was trained by **balancing** the bona fide FRLL neutral images using an augmentation technique to create more realistic neutral images. At the end, the dataset was **split** into **60, 20, and 20** for training, validation, and test sets, respectively. To ensure the model is **generalizable** despite the limited dataset, an augmentation pipeline was added to the training loop to **ensure** different perspectives of the data were presented to the models. The performance of the four models on their respective test sets is presented in Table 2.

Table 1**: Performance of the trained models on held-out test sets of the four datasets**.

	Accuracy(%)		Performance on Test Set								
	Train	**Validation**	**Acc. (%)**	**Precision**	**Recall**	**AUC**	**F1-Score**	**APCER/ FPR**	**BPCER/ FNR**	**TPR**	**TNR**
Facemorpher	100.00	100.00	100.0	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
FRLL-Morphs-OpenCV	99.97	99.15	100.0	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
FRLL-Morphs-StyleGAN	99.83	99.79	99.80	1.00	0.99	1.00	0.99	0.00	0.00	1.00	1.00
FRLL-Morphs-WebMorpher	99.79	99.58	98.42	0.97	0.99	0.99	0.98	0.03	0.00	1.00	0.97

Table 2 presents the performance analysis of models trained on four datasets—Facemorpher, OpenCV-Morphs, StyleGAN-Morphs, and WebMorpher—reveals key differences in morph detection effectiveness. Models trained on Facemorpher and OpenCV-Morphs achieved 100% accuracy across training, validation, and test sets, with perfect precision, recall, F1-score, and AUC. The absence of false positives (APCER = 0.00) and false negatives (BPCER = 0.00) suggests that these datasets contain distinguishable artifacts, making morph detection straightforward.

The StyleGAN-Morphs model performed similarly well, attaining 99.80% test accuracy with a perfect precision of 1.00 but a slightly reduced recall of 0.99. This indicates that while most morphs were detected, a few were misclassified as real faces despite StyleGAN morphs having more visual quality compared to the other landmark based morphs. In contrast, the WebMorpher model had the lowest test accuracy (98.42%) and a higher false positive rate (APCER = 0.03), suggesting that some real faces were misclassified as morphs. The slightly lower true negative rate (TNR = 0.97) further supports this observation.

The results highlight the influence of morph generation techniques on model performance. While Facemorpher and OpenCV-Morphs were easily classified, the WebMorpher morphs made differentiation more challenging. The WebMorpher model's higher false positive rate suggests that real images in this dataset share more visual pixel features with morphs images that were retained during the landmark blending of thee source images. This performance of the model across the four datasets is still impeccable despite the slight tolerable error level.

A key takeaway from these findings is the need for further evaluation across diverse datasets to assess how well these models generalize beyond their training data distribution. Therefore, we explore cross-dataset testing to evaluate how the model performance will be impacted by testing on dataset different from their training data distribution. This is presented in next section.

4.3.2 Cross-Dataset Testing

The previous evaluation of with test data from each dataset shows impressive model performance. To further test the model for generalizability to unseen data outside the distribution of its training and test

sets, a cross dataset evaluation is performed. Subsequently, each trained model with a dataset is evaluated over the other datasets as test sets to show the performance of the model on different morphing dataset created with entirely different tools with the training data.

The performance of the each model across different datasets as test set with the model default threshold settings is presented first, follow by the results for DEER of the model and for varying the model APCER at different rate to obtain the corresponding APCER.

Table 2. Results of the FaceMorpher trained Model with other datasets as test sets with threshold at default value

	Accuracy (%)	Precision	Recall	AUC	F1-Scoe	APCER / FPR	BPCER / FNR	TPR	TNR
Basel Face Database (BFD + OpenCV Morph	100.00	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
Chicago Face Database (CFD)+ OpenCV Morph	100.00	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
Utrecht ECVP) + OpenCV Morph	100.00	1.00	0.98	1.00	0.99	0.02	0.00	1.00	0.98
OpenCV Morph	100.00	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
StyleGAN Morph	94.78	0.90	1.00	0.98	0.95	0.10	0.00	1.00	0.90
WebMorpher Morph	90.26	0.83	1.00	0.96	0.91	0.19	0.00	1.00	0.81
AMSL Face Morph	**74.04**	**0.05**	**0.28**	**0.51**	**0.09**	**0.24**	**0.72**	**0.28**	**0.76**

Table 3. Results of the FaceMorpher trained Model with other datasets as test sets at Varying APCER

Dataset Name	D-EER %	APCER(FPR)@5%			APCER(FPR)@10%			APCER(FPR)@30%		
		BPCER (FNR)	TNR	TPR	BPCER (FNR)	TNR	TPR	BPCER (FNR)	TNR	TPR
Basel Face Database (BFD + OpenCV Morph	0.00	0.00	100.00	100.00	12.5	100.00	87.50	12.5	100.00	87.50
Chicago Face Database (CFD)+ OpenCV Morph	**0.00**	**0.10**	100.00	99.90	0.10	100.00	99.90	0.10	100.00	99.90
Utrecht ECVP) + OpenCV Morph	**0.00**	1.53	100.00	98.47	16.03	100.00	83.97	18.32	100.00	81.68
OpenCV Morph	0.00	0.68	100.00	99.32	0.68	100.00	99.32	0.68	100.00	99.32
StyleGAN Morph	1.86	0.68	96.97	99.32	0.68	96.97	99.32	0.68	96.97	99.32
WebMorpher Morph	4.03	0.68	92.63	99.32	0.68	92.63	99.32	0.68	92.63	99.32
AMSL Face Morph	52.07	4.90	5.20	95.10	9.80	8.23	90.20	30.39	33.01	69.61

Table 2 and Table 3 show the performance evaluation result of the FaceMorpher-trained model across multiple datasets, focusing on its generalizability when applied to unseen morph datasets generated with different tools, as well as new bonafide datasets. Notably, the Basel Face Database (BFD), Chicago Face Database (CFD) and Utrecht ECVP are newly introduced bonafide datasets, with OpenCV morphing artificially added to create a morphed class. This is to ensure that evaluation metrics can be computed without division by zero while also allowing an assessment of the model's ability to distinguish unseen bonafide samples from morphs. AMSL

morph dataset The AMSL Face Morph dataset, which consists of both bonafide (positive) and morphed (negative) images created from the same FRLL dataset were modified to comply with the ICAO portrait quality standard, involving steps like cropping, down-scaling, and JPEG compression.

At the default threshold setting (Table 4), the model achieves 100% accuracy, precision, and recall on Opencv morphed paired BFD, CFD and UtrechtECVP bonafide datasets, demonstrating accurate detection of both bonafide and morph images. This result confirms the model's ability to correctly classify new, unseen bonafide images—a key requirement for real-world deployment. Since these bonafide datasets were not part of the training process, the model's performance suggests that it effectively generalizes to real faces beyond its original dataset, rather than overfitting to specific training samples.

Similarly, the model also gave accurate detection of both bonafide and morph images in the OpenCV morphed paired with its bonafide FRLL face data attaining 100% accuracy, precision, and recall. For StyleGAN morphs, the accuracy drops to 94.78%, with a precision of 0.90 and TNR of 0.90, indicating that majority of the morphed were correcty detected save 10% of them that were falsely detected as bonafide giving a FPR of 0.1. However the model still attains a perfect TPR of 1. This suggests that StyleGAN morphing artifacts distribution vary slightly from those learned from Facemorpher morph during training. The WebMorpher dataset shows a further decline in accuracy to 90.26%, with an APCER (false positive rate) of 0.19, meaning the model misclassifies some morphs data as bonafide giving a FPR of 0.19. The lowest performance is observed with the AMSL Face Morph dataset, where accuracy drops drastically to 74.04%, precision plummets to 0.05, and recall to 0.28. These results highlight the major shifts in the FaceMorpher model training data and completely modified AMSL data that had undergone post processing operations of cropping, down-scaling, and JPEG compression. But despite this, the model stlll attain a TNR of 0.76 showing models ability to discern morphing artifacts even from in data with completely distribution shifts.

The results in Table 3, which analyze performance at varying APCER levels, further emphasize these trends. At 5% APCER, the model maintains strong performance on BFD, CFD, UtrechtECVP and OpenCV-morphed based datasets, but its detection capability decreases for StyleGAN and WebMorpher morphs. The AMSL Face Morph dataset shows the most significant deterioration, with a D-EER of 52.07%, indicating a poor tradeoff between false positives and false negatives. As the APCER threshold is increased to 10% and 30%, the true positive rate (TPR) for AMSL Face Morph improves slightly, but at the cost of increased false negatives (BPCER). This suggests that the model struggles to generalize to datasets with highly realistic morphs, and adjustments in threshold settings do not significantly compensate for this limitation.
The results demonstrate that the FaceMorpher-trained model generalizes well to its training datasets as well cross datasets with exception of completely different morphing and bonafide data distribution like the AMSL. However this limitation can be ameliorated by training on a more diverse dataset incorporating multiple morphing techniques to optimize detection across different datasets.

Furthermore, Tables 4 and 5 present the performance evaluation of the OpenCV Morph-trained model across multiple test datasets, assessing its ability to generalize to unseen bonafide samples and different morphing techniques. As shown in Table 4, the model maintains near-perfect detection for most datasets, achieving 100% accuracy on FaceMorpher and 99.50%+ on WebMorpher, CFD, and BFD datasets. However, for Utrecht ECVP, recall drops to 0.56, indicating that almost half of the bonafide

images are misclassified as morphs, yielding a BPCER of 0.44. This suggests a distribution shift in Utrecht ECVP bonafide images, affecting the model's ability to generalize effectively.

StyleGAN and WebMorpher morph datasets exhibit slightly lower accuracy than OpenCV morphs but still perform well, with StyleGAN achieving 97.70% accuracy, 0.96 precision, and 1.00 recall. However, the TNR of 0.95 for StyleGAN implies that a small fraction (FPR=5%) of morphs are misclassified as bonafide.

AMSL Face Morph dataset shows the most significant performance drop, with accuracy at 93.63%, recall at 0.07, and precision at 0.12. The BPCER of 0.93 highlights severe difficulties in detecting bonafide faces, meaning a high false rejection rate. This aligns with previous observations that AMSL data underwent post-processing operations such as cropping, down-scaling, and JPEG compression, altering morphing artifacts and affecting detection performance.

Table 4. Results of the OpenCV Morph-Trained Model with Other Datasets as Test Sets (Threshold at Default Value)

	Accuracy (%)	Precision	Recall	AUC	F1-Scoe	APCER /FPR	BPCER /FNR	TPR	TNR
Basel Face Database (BFD + OpenCV Morph	99.52	1.00	0.85	1.00	0.92	0.00	0.15	0.85	1.00
Chicago Face Database (CFD)+ OpenCV Morph	99.68	1.00	0.99	1.00	1.00	0.00	0.01	0.99	1.00
Utrecht ECVP) + OpenCV Morph	95.71	1.00	0.56	1.00	0.72	0.00	0.44	0.56	1.00
FaceMorpher	100.00	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
StyleGAN Morph	97.70	0.96	1.00	1.00	0.98	0.05	0.00	1.00	0.95
WebMorp Morph	99.50	0.99	1.00	1.00	0.99	0.01	0.00	1.00	0.99
AMSL Face Morph	93.63	0.12	0.07	0.50	0.09	0.02	0.93	**0.07**	0.98

When evaluated at varying APCER levels as shoen in Table 5, the model maintains strong detection capabilities for BFD, CFD, and WebMorpher morphs, with minimal impact on TPR even as APCER increases. However, for Utrecht ECVP, the TPR drops from 95.42% at 5% APCER to 70.23% at 30% APCER, confirming the dataset's impact on the model's generalization ability.

The most pronounced deterioration is observed in AMSL Face Morph, where D-EER reaches 53.11%, significantly reducing the trade-off between false positives and false negatives. As APCER increases, TPR improves slightly, but BPCER remains high, reinforcing that the model struggles with AMSL morph variations.

Overall, while the OpenCV Morph-trained model demonstrates strong performance on many test datasets, its generalization to highly altered bonafide and morph datasets like Utrecht ECVP and AMSL remains a challenge. Enhancing training data diversity may improve its robustness across different morphing scenarios.

Table 5. : Results of the OpenCV Morph-Trained Model with Other Datasets as Test Sets at Varying APCER Levels

Dataset Name	**D-EER %**	**APCER(FPR)@5%**			**APCER(FPR)@10%**			**APCER(FPR)@30%**		
		BPCER (FNR)	**TNR**	**TPR**	**BPCER (FNR)**	**TNR**	**TPR**	**BPCER (FNR)**	**TNR**	**TPR**

Basel Face Database (BFD + OpenCV Morph	**0.08**	**2.50**	100.-00	97.50	**2.50**	100.-00	97.50	**2.50**	100.-00	97.50
Chicago Face Database (CFD)+ OpenCV Morph	**0.18**	**1.74**	100.00	98.26	**1.74**	100.00	98.26	**1.74**	100.00	98.26
Utrecht ECVP) + OpenCV Morph	1.62	4.58	99.18	95.42	12.98	99.26	87.02	29.77	99.92	70.23
Facemorpher	0.00	0.34	100.00	99.66	0.34	100.00	99.66	0.34	100.00	99.66
StyleGAN Morph	0.17	0.34	99.84	99.66	0.34	99.84	99.66	0.34	99.84	99.66
WebMorpher Morph	0.21	0.34	100.00	99.66	+0.34	100.00	99.66	0.34	100.00	99.66
AMSL Face Morph	53.11	4.90	6.94	95.10	9.80	10.80	90.20	30.39	26.53	69.61

The StyleGAN morphs trained model also exhibits ability to generalize to other datasets, as shown in Table 6. The accuracy is exceptionally high for FaceMorpher (99.75% accuracy, 1.00 precision, and 1.00 recall) and WebMorpher (98.79% accuracy, 0.99 F1-score), indicating that the model effectively detects morphing artifacts generated by similar techniques. However, its performance on other datasets, particularly those with non-StyleGAN morphing methods, deteriorates significantly.

For datasets such as Basel Face Database (BFD) and Utrecht ECVP, the model struggles with distinguishing bonafide faces from morphs. The recall for BFD is only 0.03, and for Utrecht ECVP, it drops to 0.26, leading to a BPCER of 0.74, meaning 74% of bonafide images are misclassified as morphs. Similarly, AMSL Face Morph achieves only 0.08 recall, reinforcing that the model struggles with datasets that employ significantly different morphing techniques or image post-processing. Chicago Face Database (CFD) performs moderately well, achieving 92% accuracy and a recall of 0.88, but the false rejection rate (BPCER = 0.12) suggests the model still misclassifies bonafide faces at a non-trivial rate. This indicates that while CFD morphs share some similarity with StyleGAN morphs, subtle differences still lead to misclassification.

Table 6. Results of the StyleGAN Morph-Trained Model with Other Datasets as Test Sets (Threshold at Default Value)

	Accuracy (%)	Precision	Recall	AUC	F1-Score	APCER /FPR	BPCER /FNR	TPR	TNR
Basel Face Database (BFD + OpenCV Morph	93.66	0.02	0.03	0.74	0.02	0.03	0.97	0.03	0.97
Chicago Face Database (CFD)+ OpenCV Morph	92.00	0.95	0.88	0.98	0.92	0.03	0.12	0.88	0.97
Utrecht ECVP) + OpenCV Morph	89.79	0.45	0.26	0.89	0.33	0.03	0.74	0.26	0.97
FaceMorpher	99.75	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
Opencv Morph	98.24	0.97	1.00	1.00	0.98	0.03	0.00	1.00	0.97
WebMorph	98.79	0.98	1.00	1.00	0.99	0.02	0.00	1.00	0.98
AMSL Face Morph	92.75	0.10	0.08	0.52	0.52	0.03	0.92	**0.08**	0.97

At varying APCER levels showing in Table 7, the model shows drastic performance fluctuations. For FaceMorpher and WebMorpher, performance remains robust across all APCER levels, with TPR consistently at 99.74% and TNR above 99.34%, reinforcing the model's proficiency in detecting morphs similar to its training data. However, for BFD, CFD, and Utrecht ECVP, increasing APCER results in a significant performance declined. At 30% APCER, TPR for Utrecht

ECVP drops to 70.23%, while for BFD, it falls to 70.00%, confirming that the model struggles with unseen bonafide images.
However, the performance on AMSL Face Morphs, indicates D-EER to be 47.41%. Even with increased APCER, BPCER remains at 30.39%, and TNR drops to 28.37%, indicating severe wide differences from the attributes of the trained data and this data.

Table 7. Results of the StyleGAN Morph-Trained Model other datasets as test sets at Varying APCER

Dataset Name	D-EER %	APCER(FPR)@5%			APCER(FPR)@10%			APCER(FPR)@30%		
		BPCER (FNR)	TNR	TPR	BPCER (FNR)	TNR	TPR	BPCER (FNR)	TNR	TPR
Basel Face Database (BFD + OpenCV Morph	30.44	**5.00**	55.45	90.00	10.00	56.67	90.00	30.00	69.12	70.00
Chicago Face Database (CFD)+ OpenCV Morph	**7.41**	**5.02**	88.70	94.98	10.04	88.70	94.98	20.90	99.34	79.10
Utrecht ECVP) + OpenCV Morph	19.23	5.34	73.55	94.66	9.25	76.00	90.08	29.77	83.95	70.23
Facemorpher	0.46	0.26	99.34	99.74	0.26	99.34	99.74	0.26	99.34	99.74
Opencv Morph	0.46	0.26	99.34	99.74	0.26	99.34	99.74	0.26	99.34	99.74
WebMorpher Morph	0.33	0.26	99.59	99.74	0.26	99.59	99.74	0.26	99.59	99.74
AMSL Face Morph	47.41	0.00	0.00	100.00	0.00	0.00	100.00	30.39	28.37	69.61

The WebMorph-trained model demonstrates a strong ability to detect WebMorph-style morphs, but its generalization to other datasets is inconsistent, as seen in Table 8.
For FaceMorpher, OpenCV Morph, and StyleGAN Morph, the model achieves near-perfect scores (99.75% - 99.96% accuracy, 1.00 recall and F1-score). This suggests that the model is highly specialized in detecting synthetic morphing artifacts, particularly those that share visual characteristics with WebMorph morphs.
However, the model's performance declines for datasets such as Basel Face Database (BFD), Utrecht ECVP, and AMSL Face Morph. Similar to the StyleGAN-trained model, the recall for BFD and Utrecht ECVP is particularly low (0.03 and 0.26, respectively), leading to a high BPCER (up to 0.74). This means the model often misclassifies bonafide faces as morphs, indicating an inability to generalize to unseen face distributions.
The Chicago Face Database (CFD) + OpenCV Morph dataset performs slightly better, with 92.90% accuracy and 0.88 recall, similar to its performance in Table 6. However, the BPCER remains at 0.12, indicating that a notable number of bonafide faces are still being misclassified.

Table 8. Results of the Webmorph trained model with other datasets as test sets with threshold at default value

	Accuracy (%)	Precision	Recall	AUC	F1-Scoe	APCER /FPR	BPCER /FNR	TPR	TNR
Basel Face Database (BFD + OpenCV Morph	93.66	0.02	0.03	0.74	0.02	0.03	0.97	0.03	0.97

Chicago Face Database (CFD)+ OpenCV Morph	92.90	0.95	0.88	0.98	0.92	0.03	0.12	0.88	0.97
Utrecht ECVP) + OpenCV Morph	88.79	0.45	0.26	0.89	0.33	0.03	0.74	0.26	0.97
FaceMorpher	99.75	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
Opencv Morph	98.24	0.97	1.00	1.00	0.98	0.03	0.00	1.00	0.97
StyleGAN Morph	99.96	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
AMSL Face Morph	92.75	0.10	0.08	0.52	0.09	0.03	0.92	**0.08**	0.97

At varying APCER thresholds, FaceMorpher, OpenCV Morph, and StyleGAN Morph maintain near-perfect performance, with TPR consistently at 99.74% - 100%, meaning the model is highly effective in detecting these morphs.

However, for Basel Face Database (BFD) and Utrecht ECVP, APCER at 30% reduces TPR significantly (70.00% and 70.23%, respectively), showing that the model fails to reliably detect morphs when tested against diverse datasets. The D-EER for AMSL Face Morph is alarmingly high (47.41%), indicating random-level performance for this dataset, reinforcing that WebMorph-trained models struggle to generalize to certain morphing techniques.

The WebMorph-trained model performs exceptionally well on morphs resembling its training data but struggles with different morphing techniques and unseen datasets. To improve robustness, the model can incorporate a broader range of morphing methods, especially those found in AMSL and Utrecht ECVP, to enhance generalization across diverse datasets.

Table 9. Results of the Webmorph trained model with other datasets as test sets at Varying APCER

Dataset Name	**D-EER %**	**APCER(FPR)@5%**			**APCER(FPR)@10%**			**APCER(FPR)@30%**		
		BPCER (FNR)	**TNR**	**TPR**	**BPCER (FNR)**	**TNR**	**TPR**	**BPCER (FNR)**	**TNR**	**TPR**
Basel Face Database (BFD + OpenCV Morph	30.44	**5.00**	55.45	95.00	10.00	56.67	90.00	30.00	69.12	70.00
Chicago Face Database (CFD)+ OpenCV Morph	**7.41**	**5.02**	88.70	94.98	10.04	95.25	89.96	20.90	99.34	79.10
Utrecht ECVP) + OpenCV Morph	19.29	5.34	73.55	94.66	9.92	76.00	90.08	29.77	83.95	70.23
Facemorpher	0.08	0.26	100.00	99.74	0.26	100.00	99.74	0.26	100.00	99.74
Opencv Morph	0.46	0.26	99.34	99.74	0.26	99.34	99.74	0.26	99.34	99.74
StyleGAN Morph	0.00	0.26	100.00	99.74	0.26	100.00	99.74	0.26	100.00	99.74
AMSL Face Morph	47.41	0.00	0.00	1.00	0.00	0.00	1.00	30.39	28.37	69.61

4.4 Comparison with state of the art model

Comparative evaluation with of our model with a state of the art method proposed by Aghaide et al. [29] shows that our model performance compete favorably with such related approach as shown in Table 10. The table provides a comparative evaluation of the proposed Quadrant-Based Bi-level Self-Attention Feature Extraction (QBSAF) against the Attention-Augmented methods (Twin-

LandMark + AttI and Twin-LandMark + AttII) across different morph datasets. The metrics analyzed include D-EER, BPCER at APCER levels of 5%, 10%, and 30%.

Table 10: Comparing QBSAF with Attention Auggumented method [29] D-EER%, BPCER@APCER=5%, BPCER@APCER=10%, and BPCER@APCER=30%.

Train Dataset	**FaceMorpher + QBSAF**				**Twin-LandMark + AttI@conv2d-3b**				**Twin-LandMark + AttII @conv2d-3b**			
Test Dataset	**DEER**	**5%**	**10%**	**30%**	**DEER**	**5%**	**10%**	**30%**	**DEER**	**5%**	**10%**	**30%**
Opencv Morph	0.00	0.68	0.68	0.68	2.53	1.14	0.24	0.24	7.20	9.41	5.48	1.80
StyleGAN Morph	1.86	0.68	0.68	0.68	5.20	5.97	3.00	1.90	20.21	38.95	29.62	15.95
Webmorph	4.03	0.68	0.68	0.68	40.1	67.07	58.50	45.40	21.53	53.23	40.54	17.36

The results show that QBSAF significantly outperforms both attention-augmented baselines, achieving the lowest D-EER across all test datasets. Notably, for OpenCV morphs, QBSAF attains a perfect 0.00% D-EER with consistently low BPCER values across varying APCER levels, whereas Twin-LandMark-based approaches exhibit higher error rates, particularly with AttII, which records a 7.20% D-EER. Similarly, for StyleGAN morphs, QBSAF remains robust with a D-EER of 1.86%, compared to 5.20% and 20.21% for AttI and AttII, respectively, confirming its superior generalization to complex morphing patterns.

A more pronounced performance gap is observed on the WebMorph dataset, where QBSAF maintains a relatively low D-EER of 4.03%, while Twin-LandMark + AttI degrades significantly to 40.1% and AttII further declines to 21.53%. This trend is consistent across BPCER values at all APCER thresholds, with QBSAF showing exceptional resilience against adversarial morphing techniques. The results highlight the ability of QBSAF to effectively capture localized morphing artifacts while ensuring robust feature representation, making it a superior approach for face morphing attack detection.

5.0 Conclusion

In this paper, we presented a novel Quadrant-Based Bi-level Self-Attention Feature Extraction (QBSAF) framework for Face Morphing Attack Detection (MAD). Our method introduces a two-stage self-attention approach, which first applies self-attention within individual quadrants of an image to capture localized morphing artifacts, followed by a second self-attention stage that integrates global contextual relationships across the quadrants. This hierarchical approach enhances the model's ability to detect subtle morphing inconsistencies, leading to improved accuracy and robustness.

Experimental evaluations on multiple morphing datasets, including StyleGAN, WebMorph, OpenCV Morph, and FaceMorpher, demonstrate the effectiveness of QBSAF. Our model achieves high detection rates, maintaining superior performance in terms of precision, recall, and overall classification accuracy. In particular, our approach significantly reduces false acceptance and false rejection rates, ensuring a more reliable detection of morphed face images. Additionally, results at varying Attack Presentation Classification Error Rate (APCER) levels further confirm the adaptability of our method across different attack scenarios.

Despite its strong performance, some challenges remain. Our analysis shows that while QBSAF performs exceptionally well on synthetic morphing techniques, its effectiveness could be further improved when applied to unseen datasets with highly complex morphing processes, such as AMSL Face Morph. Future work will focus on expanding the model's adaptability by incorporating a broader range of morphing techniques and augmenting the training data with

more diverse facial variations. Additionally, integrating QBSAF with real-time biometric verification systems could further enhance its practical applicability in security-critical environments.

Future Work

The proposed QBSAF method demonstrates strong performance across its training and test datasets. However, it struggles to generalize effectively to datasets with significant distribution shifts and high variance relative to its training data. This limitation is particularly evident with the AMSL dataset, which undergoes extensive post-processing operations such as cropping, resizing, and compression. These transformations obscure critical morphing artifacts and degrade the quality of bona fide images, making detection more challenging.
Future work will focus on expanding the training dataset to include more diverse samples, ensuring better generalization to unseen morphing techniques and variations in image quality. Additionally, we will explore more advanced feature extraction techniques leveraging state-of-the-art transformer architectures. Further improvements will be pursued through hybrid approaches, integrating landmark-based feature extraction with our self-attention framework to enhance detection robustness across multiple datasets.

Acknowledgement
This work was supported by funding provided by the Tertiary Education Trust Fund (TetFund) under the Nigeria Ministry of Education through the National Research Fund (NRF 2021) with research grant no: TETFUND/ES/DR-CE/NRF2021/SETI/ICT/00011/01

References

[1] M. O. Kenneth and B. A. Sulaimon, “Averaging Dimensionality Reduction and Feature Level Fusion for Post-Processed Morphed Face Image Attack Detection,” in *Illumination of Artificial Intelligence in Cybersecurity and Forensics*, Springer, 2022, pp. 173–195.

[2] M. O. Kenneth, B. A. Sulaimon, S. M. Abdulhamid, and L. C. Ochei, “A Systematic Literature Review on Face Morphing Attack Detection (MAD),” in *Illumination of Artificial Intelligence in Cybersecurity and Forensics*, S. Misra and C. Arumugam, Eds., Cham: Springer International Publishing, 2022, pp. 139–172. doi: 10.1007/978-3-030-93453-8_7.

[3] O. M. Kenneth, S. A. Bashir, O. A. Abisoye, and A. D. Mohammed, “Face morphing attack detection in the presence of post-processed image sources using neighborhood component analysis and decision tree classifier,” in *Information and Communication Technology and Applications: Third International Conference, ICTA 2020, Minna, Nigeria, November 24–27, 2020, Revised Selected Papers 3*, Springer, 2021, pp. 340–354.

[4] C. Rathgeb, K. Pöppelmann, and C. Busch, “Face Morphing Attacks: A Threat to eLearning?,” in *2021 IEEE Global Engineering Education Conference (EDUCON)*, 2021, pp. 1149–1154. doi: 10.1109/EDUCON46332.2021.9454128.

[5] E.-V. Pikoulis, Z.-M. Ioannou, M. Paschou, and E. Sakkopoulos, “Face Morphing, a Modern Threat to Border Security: Recent Advances and Open Challenges,” *Appl. Sci.*, vol. 11, no. 7, Art. no. 7, Jan. 2021, doi: 10.3390/app11073207.

[6] 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)*, 2020, pp. 1–6. doi: 10.23919/FUSION45008.2020.9190629.

[7] T. Neubert, A. Makrushin, M. Hildebrandt, C. Kraetzer, and J. Dittmann, “Extended StirTrace benchmarking of biometric and forensic qualities of morphed face images,” *IET Biom.*, vol. 7, no. 4, pp. 325–332, 2018, doi: 10.1049/iet-bmt.2017.0147.

[8] M. Hildebrandt, T. Neubert, A. Makrushin, and J. Dittmann, “Benchmarking face morphing forgery detection: Application of stirtrace for impact simulation of different processing steps,” in *2017 5th International Workshop on Biometrics and Forensics (IWBF)*, Apr. 2017, pp. 1–6. doi: 10.1109/IWBF.2017.7935087.

[9] L. Spreeuwers, M. Schils, and R. Veldhuis, “Towards Robust Evaluation of Face Morphing Detection,” in *2018 26th European Signal Processing Conference (EUSIPCO)*, Rome: IEEE, Sep. 2018, pp. 1027–1031. doi: 10.23919/EUSIPCO.2018.8553018.

[10] E. Sarkar, P. Korshunov, L. Colbois, and S. Marcel, “Are GAN-based morphs threatening face recognition?,” in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2022, pp. 2959–2963.

[11] 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,” Jul. 07, 2020, *arXiv*: arXiv:2007.03621. doi: 10.48550/arXiv.2007.03621.

[12] “Abrosoft FantaMorph - Photo Morphing Software for Creating Morphing Photos and Animations.” Accessed: Sep. 05, 2024. [Online]. Available: https://www.fantamorph.com/overview.html

[13] L. DeBruine, “debruine/webmorph: Beta release 2.” Zenodo, Jan. 2018. doi: 10.5281/zenodo.1162670.

[14] S. Mallick, “Face Morph Using OpenCV — C++ / Python \textbar LearnOpenCV #.” Mar. 2016. Accessed: Sep. 21, 2023. [Online]. Available: https://learnopencv.com/face-morph-using-opencv-cpp-python/

[15] 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)*, IEEE, 2018, pp. 1–6.

[16] U. Scherhag, C. Rathgeb, J. Merkle, and C. Busch, “Deep Face Representations for Differential Morphing Attack Detection,” *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 3625–3639, 2020, doi: 10.1109/TIFS.2020.2994750.

[17] B. Chaudhary, P. Aghdaie, S. Soleymani, J. Dawson, and N. M. Nasrabadi, “Differential Morph Face Detection using Discriminative Wavelet Sub-bands,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021, pp. 1425–1434. doi: 10.1109/CVPRW53098.2021.00158.

[18] R. Ramachandra, S. Venkatesh, K. Raja, and C. Busch, “Detecting Face Morphing Attacks with Collaborative Representation of Steerable Features,” in *Proceedings of 3rd International Conference on Computer Vision and Image Processing*, B. B. Chaudhuri, M. Nakagawa, P. Khanna, and S. Kumar, Eds., Singapore: Springer Singapore, 2020, pp. 255–265.

[19] 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 Biom.*, vol. 9, no. 6, pp. 278–289, 2020, doi: 10.1049/iet-bmt.2019.0206.

[20] L. Wandzik, G. Kaeding, and R. V. Garcia, “Morphing Detection Using a General- Purpose Face Recognition System,” in *2018 26th European Signal Processing Conference (EUSIPCO)*, Rome, Italy: IEEE, Sep. 2018, pp. 1012–1016. doi: 10.23919/EUSIPCO.2018.8553375.

[21] 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)*, Niagara Falls, NY, USA: IEEE, Sep. 2016, pp. 1–7. doi: 10.1109/BTAS.2016.7791169.

[22] 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*, in ICBEA ’18. New York, NY, USA: Association for Computing Machinery, May 2018, pp. 6–12. doi: 10.1145/3230820.3230822.

[23] C. Seibold, W. Samek, A. Hilsmann, and P. Eisert, “Detection of Face Morphing Attacks by Deep Learning,” in *Digital Forensics and Watermarking*, C. Kraetzer, Y.-Q. Shi, J. Dittmann, and H. J. Kim, Eds., Cham: Springer International Publishing, 2017, pp. 107–120.

[24] R. Kessler, K. Raja, J. Tapia, and C. Busch, “Towards minimizing efforts for Morphing Attacks—Deep embeddings for morphing pair selection and improved Morphing Attack Detection”, doi: 10.1371/journal.pone.0304610.

[25] D. Ortega-Delcampo, C. Conde, D. Palacios-Alonso, and E. Cabello, “Border Control Morphing Attack Detection With a Convolutional Neural Network De-Morphing Approach,” *IEEE Access*, vol. 8, pp. 92301–92313, 2020, doi: 10.1109/ACCESS.2020.2994112.

[26] C. Seibold, W. Samek, A. Hilsmann, and P. Eisert, “Accurate and robust neural networks for face morphing attack detection,” *J. Inf. Secur. Appl.*, vol. 53, p. 102526, Aug. 2020, doi: 10.1016/j.jisa.2020.102526.

[27] S. Soleymani, B. Chaudhary, A. Dabouei, J. Dawson, and N. M. Nasrabadi, “Differential Morphed Face Detection Using Deep Siamese Networks,” in *Pattern Recognition. ICPR International Workshops and Challenges*, A. Del Bimbo, R. Cucchiara, S. Sclaroff, G. M. Farinella, T. Mei, M. Bertini, H. J. Escalante, and R. Vezzani, Eds., Cham: Springer International Publishing, 2021, pp. 560–572. doi: 10.1007/978-3-030-68780-9_44.

[28] H. Zhang, R. Ramachandra, K. Raja, and C. Busch, “Generalized Single-Image-Based Morphing Attack Detection Using Deep Representations from Vision Transformer,” in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Seattle, WA, USA: IEEE, Jun. 2024, pp. 1510–1518. doi: 10.1109/CVPRW63382.2024.00158.

[29] P. Aghdaie, S. Soleymani, N. M. Nasrabadi, and J. Dawson, “Attention Augmented Face Morph Detection,” *IEEE Access*, vol. 11, pp. 24281–24298, 2023, doi: 10.1109/ACCESS.2023.3254539.

[30] 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)*, 2021, pp. 1–8. doi: 10.1109/IJCB52358.2021.9484398.

[31] M. Long, C. Jia, and F. Peng, “Face Morphing Detection Based on a Two-Stream Network with Channel Attention and Residual of Multiple Color Spaces,” in *Machine Learning for Cyber Security*, Y. Xu, H. Yan, H. Teng, J. Cai, and J. Li, Eds., Cham: Springer Nature Switzerland, 2023, pp. 439–454.

[32] A. Vaswani *et al.*, “Attention is all you need. Advances in neural information processing systems,” *Adv. Neural Inf. Process. Syst.*, vol. 30, no. 2017, 2017.

[33] H. Bao, L. Dong, S. Piao, and F. Wei, “BEiT: BERT Pre-Training of Image Transformers,” Sep. 03, 2022, *arXiv*: arXiv:2106.08254. doi: 10.48550/arXiv.2106.08254.

[34] M. Chen *et al.*, “Searching the Search Space of Vision Transformer,” Nov. 29, 2021, *arXiv*: arXiv:2111.14725. doi: 10.48550/arXiv.2111.14725.

[35] Z. Chen, L. Xie, J. Niu, X. Liu, L. Wei, and Q. Tian, “Visformer: The Vision-friendly Transformer,” Dec. 18, 2021, *arXiv*: arXiv:2104.12533. doi: 10.48550/arXiv.2104.12533.

[36] L. Chi, Z. Yuan, Y. Mu, and C. Wang, “Non-Local Neural Networks With Grouped Bilinear Attentional Transforms,” presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11804–11813. Accessed: Dec. 04, 2023. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/html/Chi_Non-Local_Neural_Networks_With_Grouped_Bilinear_Attentional_Transforms_CVPR_2020_paper.html

[37] X. Chu *et al.*, “Twins: Revisiting the Design of Spatial Attention in Vision Transformers,” Sep. 29, 2021, *arXiv*: arXiv:2104.13840. doi: 10.48550/arXiv.2104.13840.

[38] A. Dosovitskiy *et al.*, “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale,” in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[39] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “CBAM: Convolutional Block Attention Module,” *CoRR*, vol. abs/1807.06521, 2018, [Online]. Available: http://arxiv.org/abs/1807.06521

[40] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, and A. Torralba, “Learning Deep Features for Discriminative Localization,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, IEEE Computer Society, 2016, pp. 2921–2929. doi: 10.1109/CVPR.2016.319.

[41] “Psychological Image Collection at Stirling (PICS).” Accessed: Jan. 06, 2025. [Online]. Available: https://pics.stir.ac.uk/

[42] L. DeBruine and B. Jones, “Face Research Lab London Set,” May 2017, doi: 10.6084/m9.figshare.5047666.v5.

[43] M. Walker, S. Schönborn, R. Greifeneder, and T. Vetter, “The Basel Face Database: A validated set of photographs reflecting systematic differences in Big Two and Big Five personality dimensions”, doi: 10.1371/journal.pone.0193190.

[44] D. S. Ma, J. Correll, and B. Wittenbrink, “The Chicago face database: A free stimulus set of faces and norming data,” *Behav. Res. Methods*, vol. 47, pp. 1122–1135, 2015.