Face Image and Video Analysis in Biometrics and Health Applications

Na Zhang

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
Computer Science

Xin Li, Ph.D., Chair
Donald Adjeroh, Ph.D.
Matthew Valenti, Ph.D.
Yuxin Liu, Ph.D.
Shuo Wang, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2023

Keywords: Face Analysis; Face Morphing; Morphing Attack and Detection; Fingerprinting; Few-Shot Learning (FSL); Autism Spectrum Disorder (ASD); Facial Dynamics; Autism Trait; Facial Micro-Expression Spotting; Micro-Expression Analysis.

ABSTRACT

Face Image and Video Analysis in Biometrics and Health Applications

Na Zhang

Computer Vision (CV) enables computers and systems to derive meaningful information from acquired visual inputs, such as images and videos, and make decisions based on the extracted information. Its goal is to acquire, process, analyze, and understand the information by developing a theoretical and algorithmic model. Biometrics are distinctive and measurable human characteristics used to label or describe individuals by combining computer vision with knowledge of human physiology (e.g., face, iris, fingerprint) and behavior (e.g., gait, gaze, voice). Face is one of the most informative biometric traits. Many studies have investigated the human face from the perspectives of various different disciplines, ranging from computer vision, deep learning, to neuroscience and biometrics.

In this work, we analyze the face characteristics from digital images and videos in the areas of morphing attack and defense, and autism diagnosis. For face morphing attacks generation, we proposed a transformer based generative adversarial network to generate more visually realistic morphing attacks by combining different losses, such as face matching distance, facial landmark based loss, perceptual loss and pixel-wise mean square error. In face morphing attack detection study, we designed a fusion-based few-shot learning (FSL) method to learn discriminative features from face images for few-shot morphing attack detection (FS-MAD), and extend the current binary detection into multiclass classification, namely, few-shot morphing attack fingerprinting (FS-MAF). In the autism diagnosis study, we developed a discriminative few shot learning method to analyze hour-long video data and explored the fusion of facial dynamics for facial trait classification of autism spectrum disorder (ASD) in three severity levels. The results show outstanding performance of the proposed fusion-based few-shot framework on the dataset. Besides, we further explored the possibility of performing face micro-expression spotting and feature analysis on autism video data to classify ASD and control groups. The results indicate the effectiveness of subtle facial expression changes on autism diagnosis.

DEDICATION

I dedicate my dissertation work to my family. A special feeling of gratitude to my husband, Kewen Wang, who has been a constant source of support and encouragement during the challenges of graduate school and life. I couldn't finish my PhD study without you. My wonderful 3-year-old daughter, Jiayang, you have been my best cheerleader. I am truly thankful for having you in my life.

I also dedicate this dissertation to my many friends who have supported me throughout the process. I will always appreciate all they have done, especially Shan Jia for helping me a lot during my pregnancy and after giving birth. Your generous assistance and encouragement made me to bravely face the challenges during my doctorate program.

ACKNOWLEDGEMENTS

I would first like to express my deepest appreciation to my committee chair and advisor, Dr. Xin Li, for the invaluable guidance and generous support you gave me throughout my doctorate program. Your expertise was precious in formulating the research problems and methodology, and your insightful feedback pushed me to sharpen my thinking and brought the best out of my work. Your professionalism and patience helped me build up self-confidence and motivate me to find new insights on my research. I believe these experiences will benefit me a lot in my future life and career.

A special thank you also goes to my committee member, Dr. Shuo Wang, who has provided immense support in various forms. This work would not have been possible without the enormous effort and generous help you gave me every step of the way. Your rigorous and enthusiastic research attitude has always deeply guided and inspired me in my PhD study, and will motivate me in the future.

I would like to extend my gratitude to Dr. Donald Adjeroh, Dr. Matthew Valenti, and Dr. Yuxin Liu, for agreeing to serve on my committee. You gave me many valuable suggestions for my work and helped me a lot to improve this dissertation. You were more than generous with your expertise and precious time.

Also, I am thankful to all my lab mates, colleagues, faculty and staff members in Lane Department of Computer Science and Electrical Engineering, for their generous assistance and support, making the completion of this research an enjoyable experience.

Finally, I would like to thank all my friends and my family, especially my parents, for the constant support and unconditional love that they have been for me.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Artificial intelligence (AI) enables machines to learn, think, and make decisions like human, using computer science technology. One of the popular fields in artificial intelligence is computer vision (CV). It seeks to develop techniques to enable computer and systems to derive useful information from images, videos, or other visual data, process and analyze the information, and take actions or make decisions in the end. The techniques involve theoretical and algorithmic studies of capturing data from real world, processing raw data, extracting discriminative features and analyzing the features.

Computer vision holds a lot of similarities with human vision. Both aim at taking actions or make recommendation by learning experience. However, there exists significant differences between them. As the Figure 1.1 shows, human vision uses human eyes to capture data from the world and human brain to deal with the captured information, and get the result finally. Human vision is a complex process. How the human vision system perceives and interprets things captured from eyes is still not completely understood. Computer vision usually uses cameras or sensing devices to capture photos or videos of the real world, and simulates the abilities of human vision into computers by building models or algorithms. In other words, computer vision can be treated as a technological implementation of human vision.

Biometrics are human related personal characteristics used to label or distinguish individuals. In computer vision, biometrics are often adopted to measure human body characteristics. The basic principle of biometrics is the unique link between individuals and their biometric reference data. There are two types of human characteristics that are commonly used in computer vision area in the literature. The first type is human physiology. It describes the shape of human body, such as mouse movement, fingerprint, palm, face,

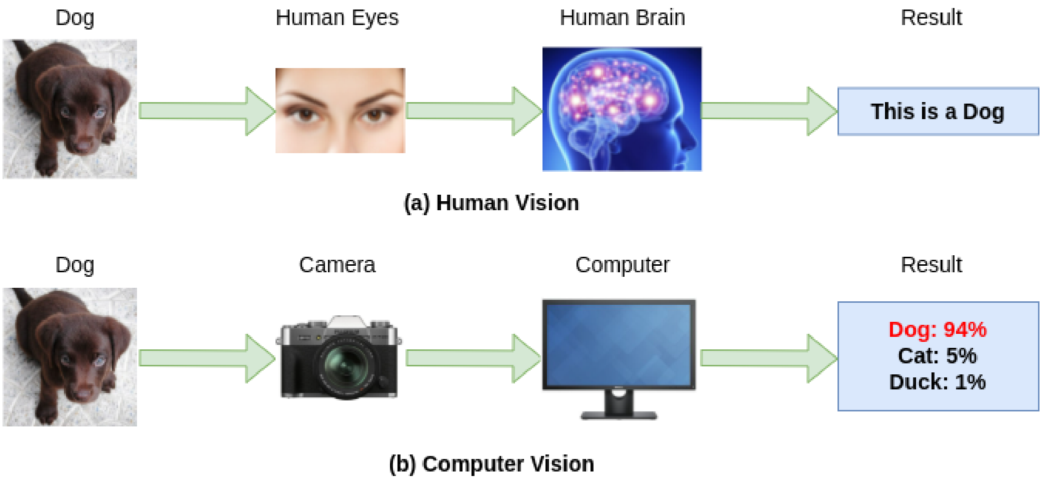


Figure 1.1: Illustration of (a) human vision and (b) computer vision.

.

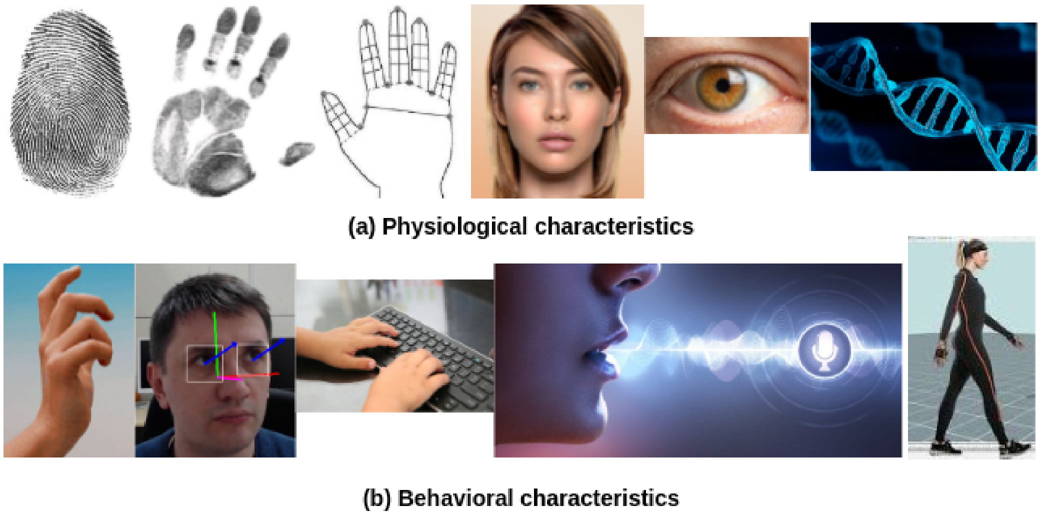

(a) Physiological characteristics
(b) Behavioral characteristics

Figure 1.2: Some examples of biometrics characteristics of human (a) physiology and (b) behavior. Top row from left to right: fingerprint, palm, hand geometry, face, eyes, DNA. Bottom row from left to right: hand gesture, gaze, typing rhythm, voice, gait.

.

DNA, hand geometry, iris, retina and odor or scent. The second type is human behaviour. This type of characteristics is to describe the pattern of behavior of a person. Commonly used characteristics contain but not limited to typing rhythm, gait pattern, gaze pattern, signature, voice, and mouse movement. Research in [1] coined the term ’behaviometrics’ to describe these behavioral characteristics. Fig. 1.2 shows some examples of both types

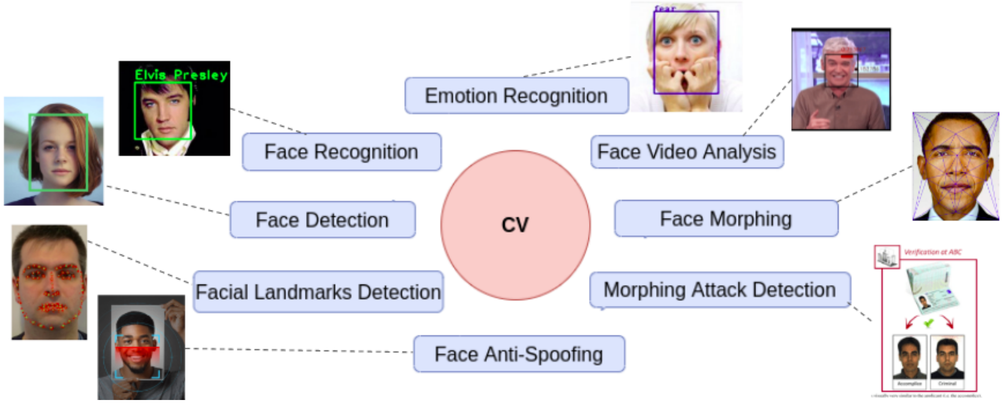


Figure 1.3: Some face related topics in computer vision.

.

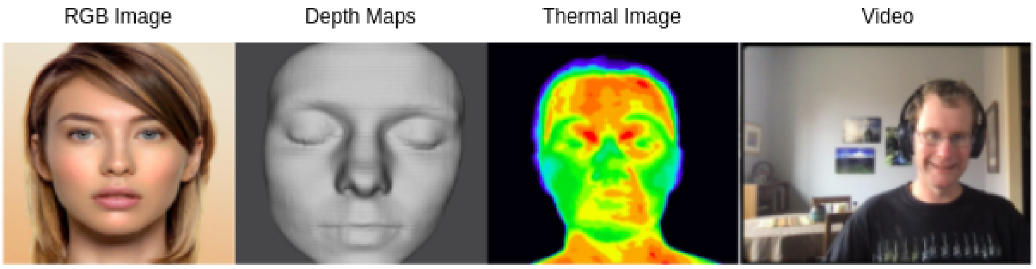


Figure 1.4: Commonly used input types for face analysis.

.

of characteristics.

Face is one of the most informative human biometrics characteristics, containing expressive information of person. We can know the person's gender, age, skin color, feelings just by seeing the face. Many studies have studied the human face from the perspectives of various different areas, ranging from computer vision and deep learning, to neuroscience and biometrics. Fig. 1.3 shows some face analysis related topics of interest in computer vision community, including but are not limited to Face Detection, Face Recognition, Facial Landmarks Detection, Head Pose Estimation, Emotion Recognition, Face Morphing, Face Morphing Attack Detection, Face Anti-spoofing, and Face Video Analysis. As shown in Fig. 1.4, the types of input of these models can be RGB images, depth maps, thermal images, or videos.

With the fast development of computer hardware, imaging technology and deep learn-

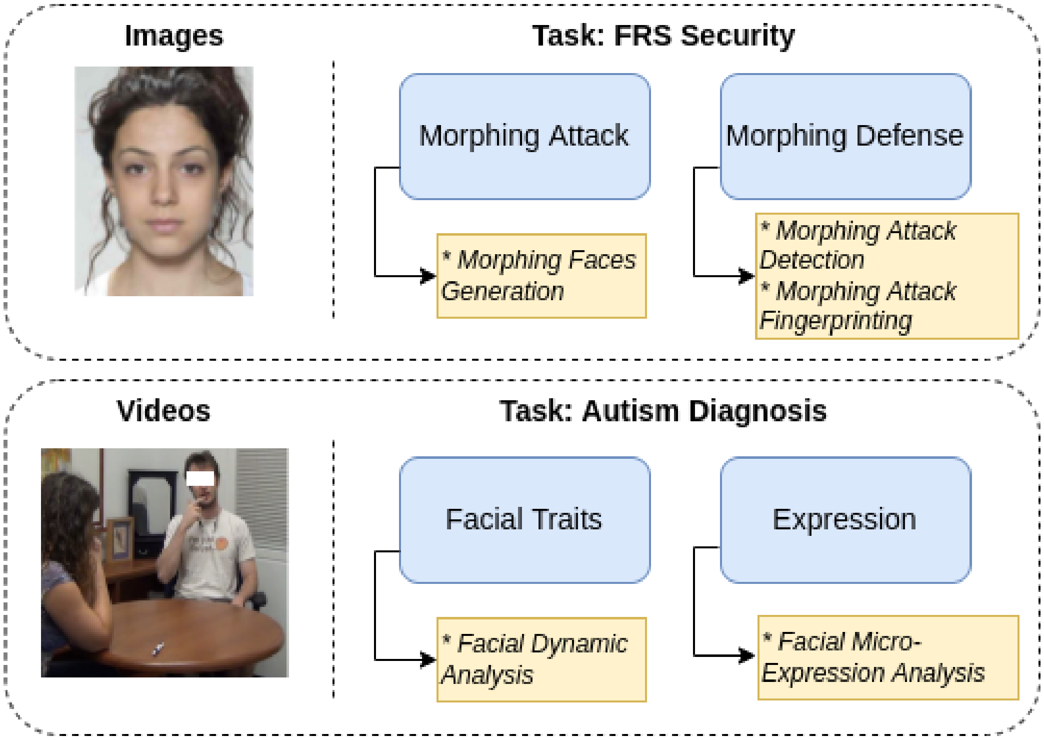

Images
Task: FRS Security
Morphing Attack
Morphing Defense
* Morphing Faces Generation
* Morphing Attack Detection
* Morphing Attack Fingerprinting
Videos
Task: Autism Diagnosis
Facial Traits
Expression
* Facial Dynamic Analysis
* Facial Micro-Expression Analysis

Figure 1.5: Main blocks of my work.

.

ing techniques, face related applications have been applied widely to daily lives, such as access control, video surveillance, etc. The demands of face analysis are also growing quickly in recent years. In the future, automatic face analysis will be one promising tool in many areas.

As shown in Fig. 1.5, this dissertation refers two types of input data: still face images and face videos. The first part focuses on image based face analysis in security of face recognition system (FRS), like morphing attack and defense. Face Morphing Attack aims at generating more realistic morphed faces. Morphing Defense aims at classifying bona fide faces and different types of morphed faces. The second part studies video based human health analysis. Two kinds of facial features are designed for autism diagnosis, i.e., facial dynamics trait feature and facial micro-expression feature.

1.1 Image based Face Morphing

1.1.1 Face Morphing Attacks

With the rapid development of deep learning technology, automatic face recognition (FR) has become a key method in security-sensitive applications of identity management (e.g. travel documents). However, the face recognition system is vulnerable to face morphing attacks [2], which aim to create facial images that can be successfully matched to more than one person. Existing face-morphing methods can be classified into two categories. One is performed on the image level via landmark interpolation, like OpenCV [3], FaceMorpher [4], LMA [5], WebMorph [6]. The other works are performed by manipulating latent codes of generative adversarial networks (GAN), such as MIPGAN-II [7], MorGAN [5], StyleGAN [8]. Both approaches have serious limitations. For landmark-based methods, as the morphing process translates landmarks and the associated texture, misaligned pixels tend to generate artifacts and ghost-like images, making the images unrealistic (i.e., easy for a human observer to detect). Similarly, for GAN-based methods, unpleasant visual artifacts, such as noticeable blurring and abnormal image patterns, often occur, often making morphed faces unnatural (see Fig. 1.6). It is natural to seek an alternative approach to face morphing attacks.

Transformer-based architectures have found successful applications in natural language processing [9, 10, 11], object detection [12], image restoration [13, 14], video inpainting [15, 16], image synthesis [17, 18, 19, 20, 21, 22], and so on. Inspired by the capability of exploiting the long-range dependency of GANformer [18], we propose to develop the GANformer-based morphing attack in a compositional latent space, as shown in Fig. **??** (b). The compositional latent space is composed of multiple latent components in local-style and one latent component in global-style, respectively. Such a compositional design allows us to have finer control of salient regions (e.g., face in the foreground) than the less important region (e.g., background). Meanwhile, MorphGANFormer is bidirectional,

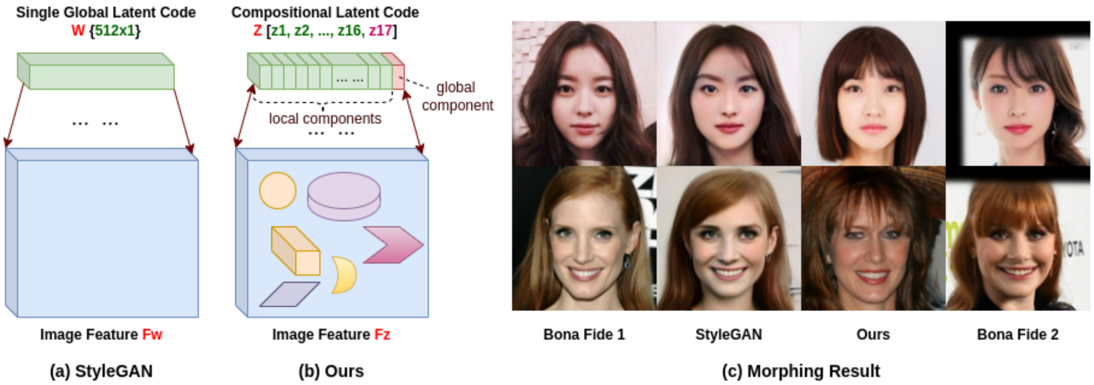


Figure 1.6: Illustration of latent code modulation of (a) StyleGAN2 and (b) Our MorphGANFormer. StyleGAN2 uses a single global-style latent code to modulate the whole scene uniformly in one direction. Ours is a compositional latent code with 16 local- and one-global-style components to impact different regions in the image allowing for spatially finer control over the generation process bidirectionally. Figure (c) shows some morphing results of StyleGAN2-based model and our MorphGANFormer (ours contain fewer visual artifacts).

allowing the propagation of information between latent codes and image features in both directions. In addition to long-range dependency, duplex attention on bipartite graphs facilitates the synthesis of high-resolution by keeping computation linear.

Under the transformer-based framework, we focus on the design of latent code in the compositional space. Unlike GANformer [18] which simply adopts the loss function of StyleGAN studies [23, 8], we have designed a class of loss functions specifically tailored for face morphing applications. Our design attempts to expedite the search for a suitable latent code by combining the strengths of both landmark-based and GAN-based approaches. Both facial landmarks and features (e.g., histogram of orientated gradients [24]) are included as content-related regularization terms. Style-related regularization consists of VGG-based perceptual loss and pixel-based MSE loss. The tradeoff between the style and context loss terms allows us to strike an improved balance between visual quality (i.e., fewer artifacts) and attack success (i.e., better matching).

Morphing and demorphing [25, 26] are two sides of the same coin, although relatively less attention has been paid to demorphing studies in the literature. The other contribution of this work is to conduct a dual study of demorphing in latent space, which complements our construction of MorphGANFormer. For the first time, we address a fundamental issue of vulnerability-detectability tradeoff for face morphing studies - i.e., what pair of images should be used in morphing study? A pair of similar images (e.g., doppelganger [27]) might be desirable from a recognition vulnerability perspective but suffers from being more easily detectable (i.e., higher APCER/BPCER rates). On the other hand, two random faces enjoy the advantage from the attack detectability perspective, but sacrifice the recognition vulnerability (i.e., lower MMPMR rate [28]). It is argued that neither the selection of doppelgangers nor random pairs is optimal and a Lagrangian multiplier-based approach should be used to achieve an improved trade-off between the recognition vulnerability and the detectability of the attack [5]. The main contributions of this paper [29] are summarized below.

- Design a transformer-based GAN model with a compositional latent space. It is made up of 16 local-style latent code components and one extra global-style component with 32×1 dimension for each. Different components can impact different regions in the image, allowing for spatially finer control over the generation process bidirectionally.

- Design special loss functions to improve the performance of the latent code optimization problem by maximizing the similarity between the generated face and the target face. Four types of loss function are adopted: biometric loss, landmark-based loss, perceptual loss, and pixel-wise mean square error (MSE).

- Extend the study of transformer-based face morphing to demorphing using the same generator. With the final morphed face and a given trusted live capture of one bona fide face, we have shown how to successfully restore the other bona fide face.

- Experimental results with both Doppelganger and random selection to demonstrate the trade-off between recognition vulnerability and attack detectability. We hope that this line of research will lead to a deeper understanding of adversarial attack and defense in the study of face morphing and demorphing.

1.1.2 Morphing Attacks Detection and Fingerprinting

Like other security systems, morphing attacks and defenses co-evolve in a never-ending race. The vulnerability of face recognition systems to morphing attacks has posed a severe security threat due to the wide adoption of face biometrics in the real world. Despite rapid progress, existing MAD methods are most constructed upon a small training dataset and single modality, making them lack of good generalization properties. The performance of existing MAD methods might be satisfactory for predefined morphing attack models, but degrades rapidly when deployed in the real-world facing newly evolved attacks. Although it is possible to alleviate such problem by fine-tuning the existing MAD model, the cost of collecting labeled data for every new morphing attack is often formidable. Moreover, we argue that MAD alone is not sufficient to meet the demand of increased security risk facing FRS. A more aggressive countermeasure than MAD to formulate the problem of morphing attack fingerprinting (MAF) - i.e., in addition to binary detection, we aim at a multiclass classification of morphing attack models.

Based on the above observations, we first propose to formulate MAD as a few-shot learning problem in this paper. Conventional few-shot learning (FSL) [30] learns the knowledge from a few examples of each class and predict the class label of new test samples. Similarly, we train the detector using data from both predefined models and new attack models (only a few samples are required) to predict unknown test samples. As illustrated in Figure 1.7, few-shot MAD learning aims at learning general discriminative features, which can be generalized from predefined to new attack models. In addition to binary MAD, we also propose to explore few-shot learning in a generalized multiclass MAF problem, aiming at classifying different attack models (a.k.a. model attribution [31]) from a few samples. The problem of few-shot MAF is closely related to camera identification (ID) [32], camera

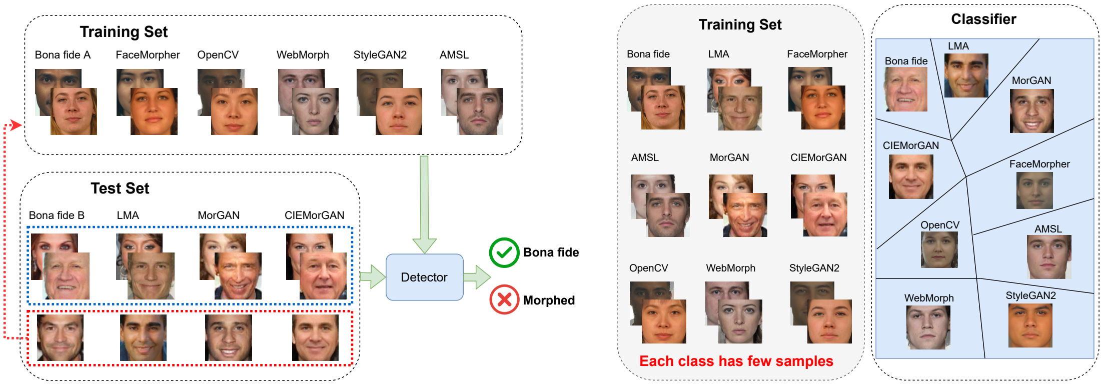


Figure 1.7: Illustration of few-shot MAD task (left) and MAF task (right). In few-shot MAD task, the training set contains bona fide faces and several predefined morphing types. The test set contains bona fide faces from other data sources and several faces of new types. Few-shot MAD is to train the detector on the training set and few collected faces (the red box) in the test set to train the detector, and test it on other faces of the test set. Few-shot learning for morphing attack fingerprinting (MAF), a multiclass extension of MAD. Each class (morphing attack model) of the training set contains a few examples. After training, the model can classify unseen test samples for each class.

model fingerprinting [33], and GAN fingerprinting (a.k.a. model attribution [31]) in the literature. The main contributions [34] are summarized as follows.

• Problem formulation of few-shot learning for MAD/MAF. We challenge the widely accepted assumptions by the MAD community, including NIST's FRVT MORPH competition. Generalization property of MAD/MAF methods will turn out as important as the optimization of recognition accuracy.

• Extend binary few-shot MAD into multiclass MAF in which both PRNU and Noiseprint have demonstrated promising performance for forensic applications. We believe such an extension will support other applications such as GAN model attribution and Deepfake fingerprinting.

• Design a fusion-based FSL method with adaptive posterior learning (APL) for MAD/MAF. By adaptively fusing the most surprising observations encountered by PRNU and Noiseprint, we can optimize the performance of FS-MAD and FS-MAF at the system level. Extensive experimental results have justified the superior generalization performance of the system to all other competing methods.

1.2 Video based Autism Diagnosis

1.2.1 Facial Dynamics Analysis

Autism is a prevalent neurodevelopmental disorder characterized by impairments in social and communicative behaviors. Faces provide nonverbal information that are important to social communication among typically developing persons. Studies have shown that more than a half visual-based nonverbal behaviors of people are around the facial region - e.g., facial expression changes, head movements, eye glances, brow raising, etc., in human communication activities [35, 36]. Some behaviors related to facial dynamics like gaze patterns have been explored in autism analysis and proved useful for autism detection. However, most works are based on facial images or short videos. There is no video data capturing the facial expression of ASD patients in a naturalistic setting. To fill in this gap, it is desirable to construct a video dataset collected from realistic interviews, such as autism diagnosis observation schedule (ADOS) [37]. Our collaboration with Caltech researchers has greatly facilitated the construction of such a database based on the ADOS interviews of nearly 50 patients from 2015 to 2017.

The patterns of nonverbal behavior of a person, such as the mobility, complexity, and dynamic activation, can be quantified to provide clues for behavior analysis [38]. Motivated by [39], we report a novel extension of previous studies into autism trait classification by facial expressions. More specifically, we will extract the facial dynamic nonverbal features of people with autism to automatically classify ASD persons with different severity from raw interaction video data called ADOS. As the golden standard for the research diagnosis of autism, each ADOS video contains 15 observation activities such as story telling and tooth brushing. During the interview, the examiner presents the participant being assessed with numerous opportunities to exhibit behaviors of interest in the diagnosis of autism through standard procedures for communication and social interaction. These videos are designed to capture the abnormal behaviors in people with ASD, and rich in terms of behaviors to analyze. All videos have been scored by ADOS-reliable clinical psychologists with consensus, and overall ratings are made at the end of the schedule. These ratings (i.e., autism, autism spectrum, and non-spectrum) can be used to formulate a diagnosis result through the use of a diagnostic algorithm. We aim at discovering whether an AI-enabled method can be developed to automatically evaluate ASD traits from the interview videos. To the best of our knowledge, this work is the first to examine a computational approach on ADOS interview videos for autism analysis and measure the severity of autism computationally.

The major contributions of this research [40] are summarized into the following four aspects:

- Construct the first dataset of long interview videos for ASD diagnosis with manually labelled scores and data sheet. Unlike popular short videos (e.g., TikTok), efficient and reliable analysis of hour-long video has remained an under-researched field in computer vision.

- Develop an ADOS video classification system capable of ASD trait classification by integrating spatio-temporal feature extraction and K-SVD sparse coding with marginal Fisher analysis (MFA).

- Propose a few-shot learning (FSL) extension of the developed system for ASD classification based on distribution calibration and adaptive posterior learning. When combined with feature-level fusion of each scene, our FSL system has reached the accuracy of 91.72% on the Caltech ADOS video data.

- Demonstrate the benefit of scene-level fusion, as well as unequal distribution of ASD-diagnostic information among different scenes. Our scene-level analysis results support the hypothesis that ASD is a complex condition beyond three-category classification, implying the necessity for further study on ASD phenotyping.

1.2.2 Facial Micro-Expression Analysis

Human face and its expressions can be the window to the soul. We can tell a lot about someone by their faces. Knowing how to read faces is one essential skills in social communication. Recently facial expression analysis has attracted great interest in wide application areas such as behavior analysis, video communication, e-health, human-machine interaction. For example, during online visual-conferences between several participants, facial expression analysis can strengthen social interaction between all participants. In electronic health applications, facial expressions analysis can help to better understand patients' minds and pain.

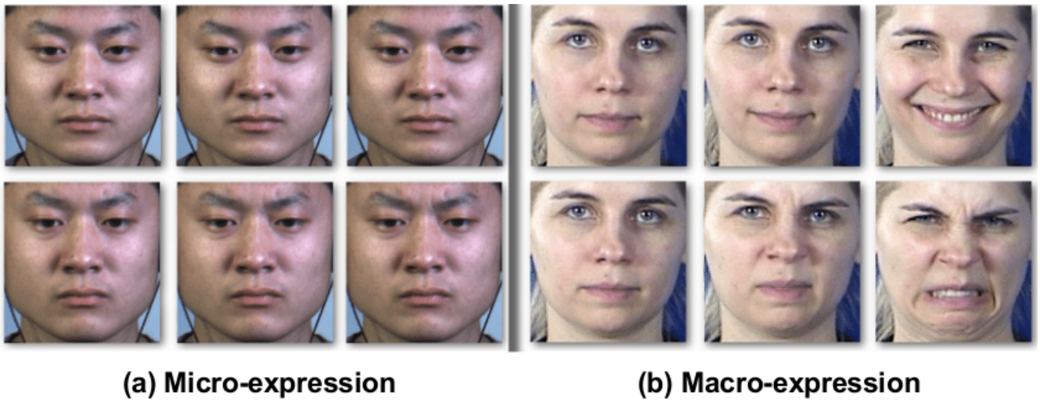


Figure 1.8: The samples of facial (a) micro- and (b) macro- expression of happiness (top) and disgust (bottom).

Basically, facial expression contains two types: micro- and macro- expression [41]. As shown in Fig. 1.8, macro-expression is easy to be perceived in daily interactions, since they are obvious, lasts long. However, micro-expression often occurs when people are trying to conceal or repress their true feelings. It is not easy to be noticed, and lasts short. The main difference between them are the duration and the intensity of expression. Since macro-expressions are often obvious, it can be analyzed based on a single image. But micro-expressions indicate brief and subtle facial movements, so it needs to be analyzed across an image sequence due to their low intensity. Micro-expressions often reveal true emotions that a person is attempting to suppress, hide, mask, or conceal. These expressions reflect a person's real emotional state. So, micro-expressions are especially important in high-risk situations like lie detection, criminal investigation, and clinical and diagnosis.

Individuals with ASD is widely known to have difficulties with socio-emotion interaction, suffering from communication disorder and emotional dysregulation, together with rigid and repetitive behaviors. These difficulties can lead to problems related to performance of expressive language, social, and emotional adaptive skills [42]. All individuals diagnosed with ASD, experience either one or more aforementioned difficulties regardless of the severity levels of diagnosis.

In the aspect of emotions, it is reported that individuals with ASD usually do not show the emotions in a way that normal people would able to recognize and understand. It is either they do not respond emotionally, or their emotional responses might sometimes seem over overreaction. There are many research that have been embarked around recognizing human emotions, particularly for autistic children and individuals. This study focuses on showing how to analyze human emotions felt by the autistic persons. The major contributions of this research are summarized into the following aspects:

- Utilize computer vision and machine learning methods to analyze face micro-expression movements of the participants in the hour-long ADOS video sequences for the diagnosis of ASD.

- Develop a ADOS video based binary classification system by spotting facial micro-expression movements and extracting subtle facial movements features with optical flow and local patches. When combined with decision-level fusion of different scenes, our system has reached the accuracy of 97.32% on ADOS video data.

- Demonstrate the necessity and effectiveness of combining micro-expression spotting and recognition tasks on scene-level fusion to handle with hour-long ADOS videos.

1.3 Organization

In the rest, we separately discuss the two parts of our works in detail. Chapter 2 gives a thorough description of related works on these fields. Chapter 3 describes a transformer based morphing attack generation model MorphGANFormer, and the few-shot single-image Morphing Attack Detection and Fingerprinting method. Chapter 4 presents the discriminative few-shot learning of Facial Dynamics feature and face micro-expression analysis in interview ADOS videos for autism classification. In Chapter 5, conclusions and future works are summarized.

CHAPTER 2
REVIEW OF RELATED RESEARCH

This chapter gives an overview of related works, containing face morphing and de-morphing, morphing detection, few-shot learning, transformer, sensor noise pattern, autism research on all kinds of behaviors, like gaze pattern, emotion, speech traits, body movements, etc., facial micro-expression, and so on.

2.1 Face Morphing

With the fast development of deep learning techniques, face recognition systems (FRS) [43, 44] have emerged as a popular technique for person identification and verification due to the ease of capturing face biometrics. In our daily lives, one of the most relevant applications of FRS is the Automatic Border Control (ABC) system, which can rapidly verify a person's identity with his electronic machine-readable travel document (eMRTD) [45] by comparing the face image of the traveler with a reference in the database.

Though FRS with high accuracy can effectively distinguish an individual from other subjects, it is also prone to be attacked to mislead or conceal the real identity. In past years, researchers have pointed out diverse potential vulnerabilities of biometric recognition systems [46], espacially with the development of image and video manipulation technique [47, 48]. In particular, as with all the applications, FRS has been found to be vulnerable to various attacks such as presentation attacks (a.k.a. spoofing attacks) [49, 50, 51, 52, 53, 54] with a goal to subvert the FRS by presenting an artifact using electronic display attack, print attack, replay attack, 3D face mask attacks, etc. Besides, recent research found that attacks based on morphed face images [55, 56], i.e., morphing attacks, pose a severe security risk across various applications.

Morphing attacks was first introduced in 2014 [55], and the authors showed that com-

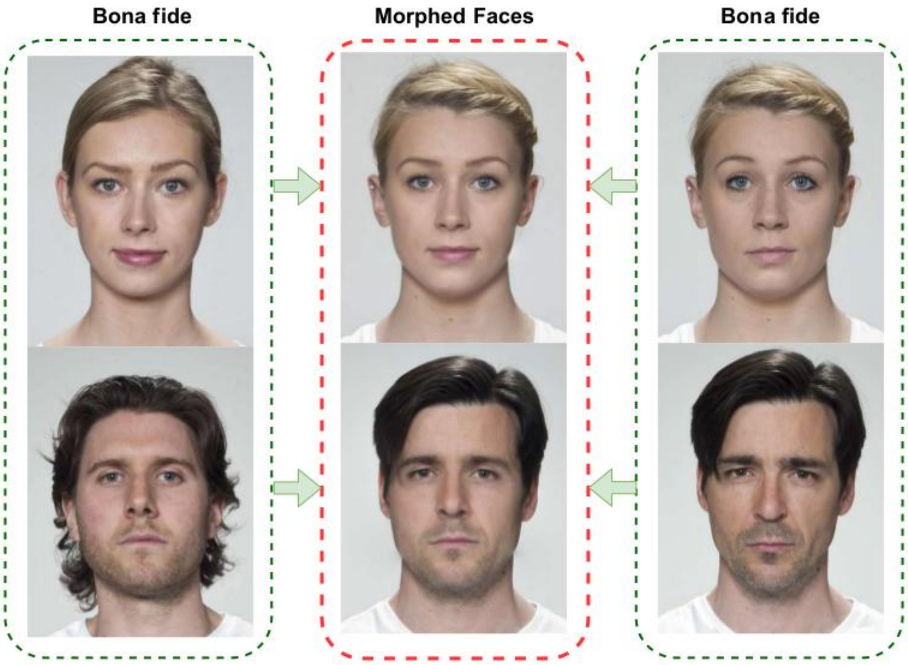


Figure 2.1: Some morphed face images generated using two bona fide faces.

.

mercial face recognition software tools are highly vulnerable to such attacks. In a further study [57], the authors showed that morphed face images are realistic enough to fool human examiners. Fig. 2.1 presents some morphed examples. The faces in the middle column are called morphed faces, which are generated by combining two bona fide faces in the first and third columns. One can see the morphed faces show strong visual resemblance to both bona fide faces. Face morphing can be treated as “a seamless transition of a facial image transforming a facial image into another” [58] in the context of biometrics. The morphed face image can be successfully verified against probe samples of both contributing subjects.

Presentation attacks try to interfere with the operation of the FRS [59] by presenting an attack instrument (e.g. print outs or electronic displays) to the face capture device. Morphing attacks represent a presentation attack at the time of enrollment. Fig. 2.2 gives an illustration of this attack. The basic idea is that: first, a morphed face image is generated by combining images of two real face including criminal’s and accomplice’s, which resembles the biometric information of the real individuals in image and feature domain; and then the

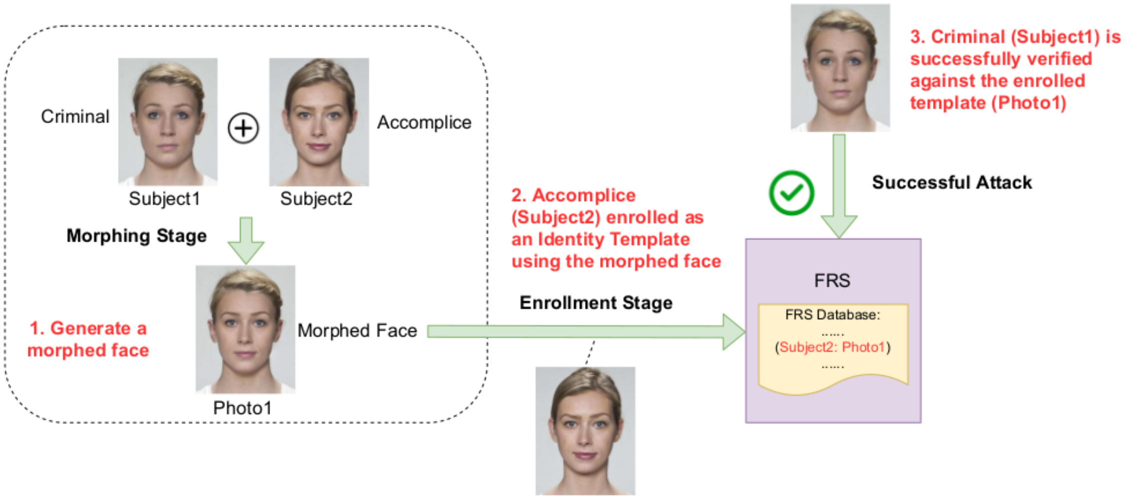

Criminal
Accomplice
Subject1
Subject2
Morphing Stage
1. Generate a morphed face
Morphed Face
Photo1
2. Accomplice (Subject2) enrolled as an Identity Template using the morphed face
Enrollment Stage
3. Criminal (Subject1) is successfully verified against the enrolled template (Photo1)
Successful Attack
FRS
FRS Database:
......
(Subject2: Photo1)
......

Figure 2.2: Illustration of face morphing attack procedure.

.

morphed image is enrolled as an identity template of the FRS by the accomplice; in a successful attack, the criminal contributing to the morphed image will be successfully verified against that single enrolled template. This means the basic principle of biometrics, the unique link between individuals and their biometric reference data, is violated. A wanted criminal is easy to obtain a legitimate eMRTD by morphing his/her facial image with an accomplice who will use the morphed face to apply for a passport.

2.1.1 Morphing Attacks Generation

With the emergence of face morphing generation techniques [60, 5, 61, 23], and numerous easy-to-use face morphing software, e.g., MorphThing [62], 3Dthis Face Morph [63], Face Swap Online [64], Abrosoft FantaMorph [65], FaceMorpher [66], the accomplice can easily submit the generated morphed image for system enrollment. As the morphed image's facial feature resembles the applicant's face, the system approves the application. Eventually, a malicious person can successfully pass the system's check. Existing generation methods can be roughly divided into two groups: landmark-based method and deep learning based method.

Landmark-based Generation. Morphed face is initially performed by detecting facial landmarks of two bona fide faces. The final morphed face is generated by landmark interpolation and texture blending. Most popular methods contain OpenCV [3], FaceMorpher [4], LMA [5], WebMorph [6], and so on.

In OpenCV [3] algorithm, face landmarks of bona fide faces are obtained by Dlib and then used to form Delaunay triangles [67], which are in-turn warped and alpha blended. Only the facial area is morphed and stitched into one of the original morphed images. FaceMorpher [4] is also an open-source tool similar to OpenCV, but with the STASM [68] landmark detector instead. Both algorithms create morphs with noticeable ghosting artifacts, as the region outside the area covered by these landmarks is simply averaged. WebMorph [6] is an online landmark-based morphing tool, which requires 189 landmarks, to generate morphed images with better alignment and of an overall higher visual quality. Ghosting artifacts are still visible and prominent around the hair and neck area, but are noticeably improved around the ears. Similar to OpenCV and FaceMorpher, LMA [5] is performed by detecting facial landmarks, the mean face points for each image are calculated and each image is then warped to sit on these coordinates after performing the Delaunay Triangulation, but it uses 194 points detected by an ensemble of randomized regression trees [69]. One special is a private Combined Morphs tool used in AMSL face morph image database [70]. This tool can generate very realistic morphs with virtually no ghosting artifacts, even around the hair and neck area, because of the additional poisson image editing.

GAN-based Generation. Taking advantage of the advanced GAN architectures and their ability to produce synthetic images, and to avoid the image-level interpolation, a few GAN-based morphing approach were proposed. StyleGAN2 [8] is a morphing algorithm which can generate high resolution realistic looking faces with no noticeable artifacts. The real images are projected into the latent space, and latent vectors are linearly interpolated to generate a new latent vector of the morphs, which are fed back into the generator. Based on StyleGAN [23], the MIPGAN-II [7] was designed to generate images with higher identity preservation by introducing a loss to optimize the identity preservation in the latent vector.

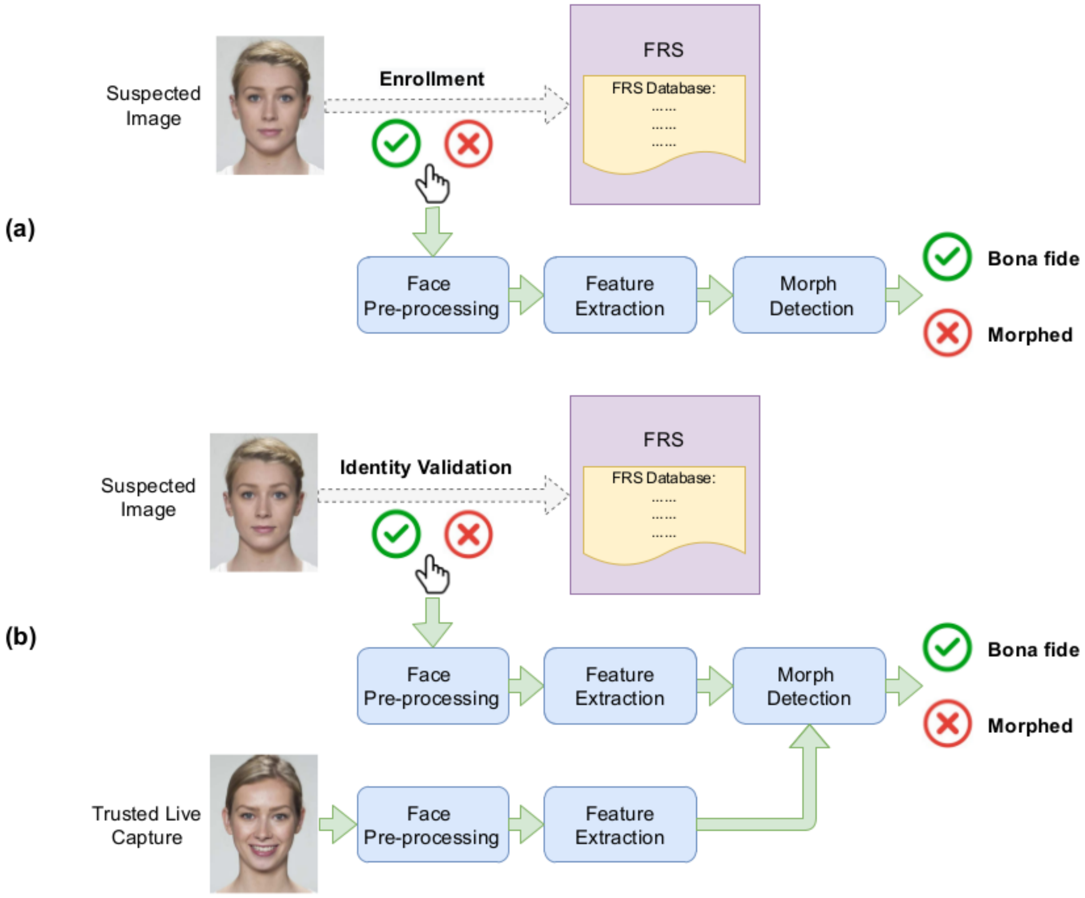


Figure 2.3: Illustration of two morphing attack detection scenarios: (a) single image based morphing attack detection, and (b) differential image based morphing attack detection.

.

MorGAN [5] is a face morphing attack approach, based on automatic image generation using a specially designed GAN. Based on the MorGAN, an enhanced version of data generated by CIEMorGAN [71] is released, which attempts to suppress the generation artifacts and increase the image resolution.

2.1.2 Morphing Attack Detection

There is an imminent need to protect the security of FRS by detecting morphing attacks. In order to protect the security of FRS, the detection of face morphing attack is becoming an urgent problem to be resolved. In the recent past, a number of morphing attack detection (MAD) approaches have been proposed. Focusing on the workflow (face pre-processing, feature extraction, and detection) of a generic biometric system, proposed approaches can be coarsely categorized in two types with respect to the considered morph detection scenario as shown in Fig. 2.3: (a) single image based MAD (S-MAD), i.e. no-reference morph detection; (b) differential image based MAD (D-MAD). Fig. 2.3 (a) is a basic S-MAD method focusing on a single potentially morphed image presented to the algorithm. The detection action occurs during enrollment, e.g. the passport application process. Fig. 2.3 (b) is a D-MAD method to distinguish morphed and bona fide face images with a corresponding face image captured in a trusted environment. The detection action occurs at the time of identity validation, e.g. passing through an Automated Border Control gates at borders.

Existing S-MAD methods can be further classified into two subtypes [72]: model-based (using handcraft features) and deep learning-based. Photo-Response Non-Uniformity (PRNU) noise based methods [73, 74, 75, 76] represents the former subtype for its popularity and outstanding performance. Originally proposed for camera identification, PRNU turns out to be useful for detecting the liveness of face photos. For the latter subtype, Noiseprint [33] used a CNN to learn salient features, aiming at improving the detection performance as well as supporting fingerprinting applications.

Model-based S-MAD. Residual noise feature-based methods are designed to analyze the pixel discontinuity that may be largely impacted by the morphing process. Generally, the noise patterns are extracted by subtracting the given image from a denoised version of the same image via different models, such as deep Multi-scale Context Aggregation Network (MS-CAN) [77]. The most popular one should be sensor noise patterns, such as PRNU. Both PRNU-based [76, 73, 74, 75] and scale-space ensemble approaches [78, 79] were studied recently.

Learning-based S-MAD. Along with rapid advance in deep learning, many methods have considered the extraction of deep learning feature for detection. The use of convolutional neural networks has reported promising results [80]. Most works are based on pretrained networks and transfer learning. The commonly adopted deep models contain AlexNet [81], VGG16 [82], VGG19 [82, 79], GoogleNet [83], ResNet [84], etc. Besides, a few self-design models were proposed too. More recently, a deep residual color noise pattern was proposed for MAD in [85]; and an attention-based deep neural network (DNN) [86] was studied focusing on the salient Regions of Interest (ROI) which have the most spatial support for morph detector decision function.

Differential Image based MAD (D-MAD). Existing methods mainly focus on feature difference and demorphing. In feature difference based methods, features, such as texture information [87], 3D information [88], gradient information, landmark points [89] and deep feature information [90], of the suspected image and the live image are subtracted and further classified. The idea behind demorphing methods is to invert the morphing procedure and reveal the component images that are used to generate the morphed image. The commonly used features contain landmark points [25] and deep learning feature [91]. Some GAN-based, like FD-GAN [92], cGAN [93], and Siamese architecture based [94] detection methods were proposed.

2.2 Face De-Morphing

The common definition of demorphing is that by using one bona fide identity as a reference image, the morphed face image can be reverted (or demorphed) to reveal the identity of the other bona fide subject. In [25], the authors reverse the morphing operation to find the second bona fide by exploiting the live image acquired from the first bona fide. In FD-GAN [92], the authors designed a symmetric dual network and adopted two layers of restoration losses to separate the second bona fide's face image. The basic idea is that it first restores the image of the second bona fide from the given morphed input using the first bona fide as a reference, and then tries to restore the first bona fide from the morphed image with the restored second bona fide as a reference. In [93], a conditional GAN is designed to disentangle identity from the morphed image using the pixel difference by minimizing conditional entropy. Recently, [95] proposed a method to recover both bona fide face images simultaneously from a single given morphed face image without reference image or prior knowledge. Such blind demorphing is conceptually similar to the unmixing of hyperspectral images.

In addition, some works have been proposed that treat face demorphing as a technique to detect reference-based morphing attacks [91, 96]. For example, in [96], the authors apply a fusion of two differential morphing attack detection methods, i.e., demorphing and deep-face representations, for detection. [26] focuses on the robustness of face demorphing and uses it as a technique to protect face recognition systems against the well-known threat of morphing.

2.3 Few-Shot Learning

Few-shot learning addresses the challenge with the generalization property of deep neural networks - i.e., how can a model quickly generalize after only seeing a few examples from each class? Early approaches include meta-learning models [97] and deep metric-learning techniques [30]. More recent advances have explored new directions such as relation network [98], meta-transfer learning [99], adaptive posterior learning (APL) [100], and Clustering-based Object Seeker with Shared Object Concentrator (COSOC) [101].

2.4 Camera and Manipulation Fingerprinting

PRNU, as a model-based device fingerprint, has been used to carry out multiple digital forensic tasks, such as device identification [102], device linking [103], forgery localization [104], detection of digital forgeries [105]. It can find any type of forgeries, irrespective of its nature, since the lack of PRNU is seen as a possible clue of manipulation. Moreover, the PRNU based MAD methods [73, 74, 75, 76] also confirm the usefulness of sensor fingerprint on MAD.

In recent years, PRNU has been successfully applied in MAD [74, 73, 75]. The method in [74] shows region-based spectral analysis of PRNU reliably detects morphed face images, while it fails if image post-processing is applied to generated morphs. Based on previous work, a PRNU variance analysis was performed in [73]. It focused on the local variations of face images, which can be useful as a reliable indicator for image morphing. The work in [75] proposed an improved version of the scheme based on previous PRNU variance analysis across image blocks.

2.5 Transformer in Computer Vision

The transformer that focuses on long-range relationships, is a highly successful model in natural language processing [9, 10, 11] which inspired the development of self-attention for image classification. The transformer-based architectures has been applied into many other vision tasks, such as detection [12], image restoration [13, 14], video inpainting [15, 16], etc, and various applications of daily lives, like illicit drug identification on Instagram [106, 107, 108], mapping of urban canyon geometry [109], and achieves state-of-the-art in most tasks.

The famous one that is worth mentioning is Vision Transformer (ViT) [110], one of the first research efforts at the intersection of Transformers and computer vision. It is becoming a more dominating technique in various vision tasks in comparison to Convolutional Neural Network (CNN). The main difference between transformer and CNN is that transformer splits an image into a sequence of patches and applies uses self-attention operations, not convolution operations. Another success is BERT (Bidirectional Encoder Representations from Transformers) [11], which is originally designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. It is quickly adopted by a self-supervised learning framework in computer vision, Bidirectional Encoder representation from Image Transformers (BEiT) [111].

Many studies have been proposed to achieve similar performance to CNN with the same amount of data. The appealing performance of transformer is generally attributed to the long-range modeling capacity. However, one of the challenges that vision transformers are faced with is the large number of visual tokens because the computational cost quadratically increases along with the token number.

2.6 Autism Diagnosis

Autism Spectrum Disorder (ASD) is a developmental disorder that affects communication and behavior [112]. Individuals with ASD often have difficulty interpreting and regulating their own emotions, as well as understanding the emotions expressed by others [113]. Studies on facial expression/emotion recognition and ASD have primarily used static images with posed expressions in the literature (e.g., [114] and [115]). Despite the extension into dynamic video with posed facial expression [116], there is still no automated and comprehensive analysis of facial expression in autism, especially in natural settings [117]. Existing research on computer vision for ASD diagnosis are mainly in eye-tracking data [118] and self-person perspective photos [119]. This is primarily due to the lack of video data collected from realistic interviews capturing the facial expression of ASD patients.

The rise of artificial intelligence (AI), including machine learning (ML) and computer vision (CV) technologies, has made impressive progress from face recognition and emotion analysis to action detection and speech recognition. Rapid advances in AI have also leveraged on the field of behavior imaging for understanding human behaviors [120, 121] and early diagnosis of autism [122]. ML technology has enhanced the diagnosis and intervention research in behavioral sciences such as depression diagnosis [39, 123] and stroke rehabilitation [124, 125]. In recent years, CV-based approaches have presented a class of quantitative and objective diagnosis tools for ASD by focusing on gaze patterns (e.g., eye movement [126], [127], visual attention [118], [128] or eye tracking [129], [130]) or body movements (e.g., gesture analysis [131], motor skills [132], and repetitive behaviors [133]). As of today, there still lacks a CV-based study about the feasibility of ASD classification based on facial expression.

Table 2.1: The ASD diagnosis methods using ML techniques. “-” means N/A.

Methods	Target Group	Characteristic	Data Description	Participants	Algorithm/Model
Jiang&Zhao [118]	adult	gaze pattern	image	20ASD+19TD	DNN
Eraslan et al. [134]	adult	gaze pattern	image	15ASD+15TD	Scanpath Trend Analysis
Ahuja et al. [135]	adult	gaze pattern	video	35ASD+25TD	Gaze features
Tao&Shyu [130]	child	gaze pattern	image	14ASD+14TD	CNN+LSTM
Duan et al. [129]	child	gaze pattern	image	13ASD	GAN
Dris et al. [127]	child	gaze pattern	image	-	Region of Interests
Wei et al. [128]	child	gaze pattern	image	-	CNN
Li et al. [136]	child	gaze pattern	video	53ASD+136TD	Displacement Feature
Wan et al. [137]	child	gaze pattern	image	37ASD+37TD	Areas of Interest
Fernández et al. [138]	child	gaze pattern	image	8ASD+23TD	CNN
Liu et al. [139]	child	gaze pattern on face	image	29ASD+2groups TD	K-means
Jiang et al. [140]	-	gaze pattern on face	image	23ASD+35TD	DNN
Kaliukhovich et al. [141]	child+adult	gaze pattern	images	94ASD+38TD	-
Leo et al. [142]	child	facial expression	image	17ASD+10TD	CNN
Beary et al. [143]	child	facial expression	image	1,507ASD+1,507TD	MobileNet
Akter et al. [144]	child	facial expression	image	1,468ASD+1,468TD	MobileNet-V1
Lu& Perkowski [145]	child	facial features	image	561ASD+561TD	VGG16
Kowallik et al. [146]	adult	facial expression	image	55ASD	Logistic Regression
Lecciso et al. [147]	child	facial expression	image	12ASD	-
Guo et al. [148]	child	facial expression	image	30ASD+30TD	-
Elshoky et al. [149]	child	facial features	image	2,936	A set of ML methods
Bangerter et al. [150]	child+adult	facial expression	video	124ASD+41NT	Gaussian Mixture Model
Banire et al. [151]	child	facial expression	video	20ASD+26TD	CNN
Zlibut et al. [152]	adult	facial expression	video	27ASD+57NT	K-means
Alvari et al. [153]	child	facial expression	video	18ASD+15TD	Openface
Zunino et al. [131]	child	grasping a bottle	video	20ASD+20TD	LSTM
Crippa et al. [132]	child	reach-to-drop task	video	15ASD+15TD	Kinematic Measures
Tian et al. [133]	child	repetitive behavior	video	-	CNN
Oller et al. [154]	child	speech	acoustic data	232	LDA
Ecker et al. [155]	adult	brain anatomy	-	-	Volumetric&Geometric feature of cortical surface
Sherkatghanad et al. [156]	adult	brain imaging	image	539ASD+573TD	CNN
Thabtah&Peebles [157]	adult	questionnaires	text	189ASD+515TD	Rule-based architecture
Devika&Chinnaiyan [158]	adult+toddler	questionnaires	text	-	A set of ML methods
Nasser et al. [159]	all	questionnaires	text	1100 instances	ANN
Raj&Masood [160]	all	questionnaires	text	1100 instances	A set of ML methods
Peral et al. [161]	all	questionnaires	text	1100 instances	A set of ML methods
Hossain et al. [162]	all	questionnaires	text	1100 instances	A set of ML methods
Küpper et al. [163]	adult	behavior	ADOS codes	385ASD+288TD	SVM
Ruan et al. [119]	adult	attention behavior	photos	16ASD+21TD	DNN
Subah et al. [164]	child+adult	fMRI	image	402ASD+464TD	DNN

The widely accepted behaviors that people with ASD are mainly indicated in gaze patterns, hand gestures, speech traits, and reciprocal social exchange. Due to the heterogeneity of ASD symptoms, participants may only demonstrate abnormalities in a subset of these categories. Recently, machine learning methods have been applied to autism analysis and diagnosis with various modalities, such as atypical visual scanning patterns during face and emotion perception, abnormal hand gestures and body behaviors, strange speech traits (e.g., loud volume, limited vocal variation, abnormal speech speed), etc. Here, major related works are discussed briefly.

2.6.1 Gaze Pattern

People with ASD have atypical attention towards visual stimuli such as human faces [165]. Lots of research on the gaze pattern has been done in autism and developmental disorders [166, 167, 168, 169, 170, 171, 172, 173, 174, 135, 138, 134]. ML-based methods have been widely used and achieved good performance in various applications of gaze pattern analysis. In [118], the authors concentrated on the analysis of differences in eye movement patterns between healthy people and those with ASD via a Deep Neural Network (DNN), and a Fisher-score-based image selection method was adopted to learn more discriminative features for efficient and accurate diagnosis. In [129, 138, 128] proposed saliency prediction models based on deep neural networks- e.g., Generative Adversarial Network (GAN), Convolutional Neural Network (CNN), to predict atypical visual attention of children with ASD. Most recently, SP-ASDNet [130] is a framework which uses both CNN and Long Short-Term Memory (LSTM) networks to classify whether an observer is typically developed (TD) or has ASD based on the gaze's scan path.

Traditional ML methods like Support Vector Machine (SVM) have also been used for gaze pattern analysis [135, 127, 136, 137]. In [135], five gaze features (standard deviation of gaze points, standard deviation of difference in gaze points, standard deviation between the gaze and annotated object of interest, RMSE between the gaze and annotated object of interest, and delay in looking at the object of interest) were calculated for binary classification using SVM. In [136], a method to automatically recognize ASD children was proposed for raw video data via analyzing the trajectory of eye movement and used the SVM for classification. In [137], the fixation times of children with ASD was investigated for classification and demonstrated that a short video clip may provide enough information to distinguish ASD from TD children. In [126], both electroencephalography (EEG) and eye movements were considered for ASD diagnosis. They have used several methods like SVM, logistic regression, deep neural network, and Gaussian Naive Bayes for classification. Recent work [134] used scan-path trend analysis (STA) to identify the trending path of users on a web page based on their eye movements. [141] applied eye-tracking in children and adults to assess the allocation of visual attention in a dynamic social orientation paradigm, and found qualitative differences across ages in ASD.

2.6.2 Interpretation of Facial Expressions

The human faces are among the most important visual stimuli for social interactions. Failure to accurately interpret facial expressions (i.e., happiness, surprise, fear, anger, disgust, sadness) [175, 176, 177, 178] and face-processing [179, 180, 181, 38, 182, 183, 184] is one of the key impairments in ASD. Recent works also indicate that observers with ASD have difficulties using patterns of facial motion to judge identity and gender, and may be less able to derive global motion perception [165, 185, 186, 187]. Machine learning methods have been used in this kind of analysis. In [139], a machine learning method is proposed to classify children with ASD and two groups of matched controls by analyzing the gaze patterns in a face recognition task. Despite their prominent accuracy, the face stimuli and the structured recognition task are highly dependent on the existing knowledge about ASD, limiting their generalizability to other clinical populations or young children who may fail to understand or comply with the task instruction. In [140], a Dynamic Affect Recognition Evaluation (DARE) task was adopted to distinguish between ASD and TD. Participants were asked to recognize one of six emotions while observing a slowly transitioning face video, and their response time and eye movements were recorded.

Besides, analyzing the facial expressions of participants to distinguish the differences between ASD and TD is also a good point. Many works have been proposed in recent years. In [144], a transfer-learning-based autism face recognition framework is proposed to identify kids with ASD in the early stages of face images collected from the Kaggle data repository. In [143], it introduced a deep learning model based on MobileNet to classify children with ASD. In [148], a facial expression analysis system is proposed to evaluate the differences of empathy ability between ASD and TD children by analyzing the real-time facial expressions of children. In [145], a VGG16 transfer learning-based ASD screening solution is designed to detect ASD using facial images on a unique ASD dataset of clinically diagnosed children. In [149], the authors used various machine learning methods (SVM, Random Forest, deep learning, etc.) to predict ASD in children using facial images. In [147], several computer-based interventions are designed to help children with an autism spectrum condition to improve the emotional competencies.

In addition, video data can provide more information for analysis. In [150], it used automated facial analysis software to investigate the differences between ASD and TD groups of children and adults with short clips of video. The authors in [151] designed two face-based attention recognition models to detect and classify children with ASD. One is based on geometric feature transformation and the other is based on the transformation of time-domain spatial features to 2D spatial images. The author in [153] investigated the facial expressions of infants to analyze facial micromovements in videos to extract the subtle dynamics of Social Smiles. In [142], it analyzed how ASD and TD children produce facial expressions by monitoring facial muscle movements, and the output is then fused to model the individual ability to produce facial expressions. In [152], facial expression coding and clustering approach are applied to find differences between autistic and neurotypical adults. In [146], it applied a baseline-intervention-retest design to investigate the impact of imitation on facial emotion recognition with six basic emotional expressions.

2.6.3 Body Movements

The underlying rationale of using body movements for ASD detection comes from psychological and neuroscience studies, claiming that the executions of simple motor acts are different between pathological and healthy subjects, and this can be used to discriminate between them. In this category, the behaviors of subjects are mostly recorded by video cameras [131, 133, 188, 189]. In [131], a simple gesture of grasping a bottle by patients and healthy children was recorded and processed by a Recurrent Deep Neural

Network (RNN) for classification. [133] introduced an end-to-end deep architecture, the one glimpse early ASD detection (O-GAD) network, for video-based early ASD detection via taking arbitrary-length videos as input. The network can detect ASD typical actions and determine if repetitive behaviors appeared only at one glimpse. [132] developed a supervised ML method to determine whether a simple upper-limb movement (reach-to-drop task) could be useful to accurately classify low-functioning children with ASD aged 2 to 4. This work offered insight into a possible motor signature of ASD that may be potentially useful in identifying a well-defined subset of patients, reducing the clinical heterogeneity within the broad behavioral phenotype.

2.6.4 Speech Traits

For generations, the vocal study and its role in language have been conducted laboriously, with human transcribers and analysts coding and taking measurements from small recorded samples. Large-scale statistical analysis of strategically selected acoustic parameters on the development of infant control over infrastructural characteristics of speech is not only able to track children's development of acoustic parameters known to play key roles in speech, but also is able to differentiate vocalizations from typically developing children and children with autism or language delay. [154] adopted this analysis method to show the potential to fundamentally enhance research in vocal development and to add a fully objective measure to detect speech-related disorders, such as autism, in early childhood.

2.6.5 Questionnaires

There are some datasets published in the form of questionnaires [157, 190, 158]. A set of questions are designed for diagnosing ASD. The participants are asked to answer the questions for data collecting. For example, [159, 160, 161, 162] used the dataset collected from the UCI Repository, which are ASD Screening Data for Adult [191], Children [192], and Adolescent [193]. These datasets have 20 common attributes that are used for prediction, such as age, sex, behavioral features, jaundice attributes.

2.6.6 Others

Other methods such as brain imaging are also used for ASD detection -e.g., a multiparameter classification approach was developed in [155] to characterize the complex and subtle structural pattern of gray matter anatomy implicated in adults with ASD and discriminate between ASD and TD control by SVM. Resting-state functional magnetic resonance imaging (fMRI) data from a multisite dataset named the Autism Brain Imaging Exchange (ABIDE) were used in [156] for ASD detection. A deep neural network (DNN) classifier was proposed in [164] proposed to detect ASD using functional connectivity features of resting-state fMRI data such as ABIDE dataset. Most recently, photos taken by ASD are shown to have different characteristics from controls in [119, 168]. A summary of existing autism diagnosis methods using ML techniques is shown in Table 2.1. More survey works can be referred to in papers [194, 195, 196].

2.7 Emotion Recognition for ASD

There are a lot research on emotion recognition for autism spectrum disorder diagnosis. The commonly used stimuli to invoke emotions of participants contains picture, video and task. Picture, is a series of images that invoke the desired emotions. Video, is a bi-sensory stimulation that combines audio and visual stimulus. Task, is a directive real-life situation that elicit the desired emotions. The evoked emotions includes some basic emotions, like fear, anger, happiness, sadness, disgust, surprise, etc. An effective method for the paradigm of elicitation is to watch an emotional picture/video or do some tasks while maintaining a neutral expression (i.e. suppressing emotions). A common approach adopted in the published literature consists of presenting participants with emotional content which is expected to rouse their emotions, while at the same time asking them to disguise their emotions and maintain a neutral facial expression [197]. And for the types of extracted feature about emotions includes optical flow, spatio-temporal, thermal intensity value(TIV) from thermal images, EEG, texture, etc. Commonly used types of classifiers have K-NN, K-means, SVM, Decision Tree, GMM, CNN, etc.

2.8 Facial Micro-Expression Analysis

Micro-expression analysis contains two basic tasks: spotting and recognition. Spotting is to locate the time interval where micro-expressions are detected in the video sequence. Recognition is to classify micro-expressions. Spotting is a vital step, which is a prerequisite for further recognition. With the development of deep learning, the recognition task gained a widespread popularity, while the spotting task, especially on long videos still remain subdued [198, 199].

2.8.1 Micro-Expression Spotting

Generally, facial expressions usually undergo three distinct phases: onset, apex, and offset. In [200], it describes that onset occurs when facial muscles begin contracting, apex is the phase where the facial action is at its peak intensity, and offset occurs when the facial muscles return back to neutral state. The task of spotting is to locate the duration segments from onset to offset. The major challenges of spotting contains: (1) relies on setting the optimal thresholds to detect micro-expression for any given feature; (2) different people may perform different extra facial actions, such as some people blink habitually, while other people sniff more frequently; (3) when recording videos, many comprehensive factors may significantly influence the micro-expression spotting, like head movement, physical activity, recording environment, lighting condition.

A discriminative feature should be able to capture the difference in both spatial and temporal domains, and should be able to capture the micro-level differences. The published studies on micro-expression spotting can be classified as appearance-based, motion-based, and general methods. For appearance-based methods, most commonly adopted feature contains local binary pattern (LBP) [201, 202, 203], histogram of oriented gradients [204, 203] for feature difference analysis between two frames in a fixed duration. Some works adopt established pre-processing techniques involving landmark detection, region masking, and region of interest (ROI) selection. The micro-expression is determined if the frame's feature vector is above the threshold set for peak detection. Motion-based approaches focus on characterizing the subtle facial movements. Most commonly used motion feature is optical flow. In [205], the authors first introduced optical strain (a derivative of optical flow) to analyze subtle motion changes based on the elastic deformation of facial skin tissue by considering the amount of strain observed across time at different facial regions. To encourage researchers towards spotting micro-expressions in long videos, the micro-expression community has also organized the 3rd MEGC Workshop (MEGC2020) [206].

2.8.2 Micro-Expression Recognition

Micro-Expression recognition is to attempts to categorize a video of micro-expressions into one of the expression classes, such as happiness, sadness, disgust, fear, contempt, anger, surprise, etc. Various inputs are used in recognition task, such as apex frame, onset to apex, onset and apex, image with dynamic information, optical flow and combinations [207].

In recent years, the development of micro-expression has been very rapid. Early study on micro-expression recognition focused on extracting handcrafted features. For example, in [208], they use local binary pattern histograms from three orthogonal planes (LBP-TOP) to describe the spatio-temporal local textures. In [209], they use bi-weighted oriented optical flow (Bi-WOOF) to encode essential expressiveness. And in [210], the sparse part of robust PCA is used to extract the subtle motion information. Besides, motion magnification [203] and hierarchical spatial division scheme [211] are also proposed.

With the popularity of deep learning methods, most studies on recognition has begun to design deep learning based frameworks for micro-expression recognition. The possible first work uses transfer learning from objects and facial expression-based CNN models

[212]. In the following years, different strategies were proposed, such as 3D flow-based [213], facial appearance and geometry [214], 2D landmark feature map [215], incorporating unique gender characteristics [216], etc. Most popular is optical-flow-based [217, 218]. Efficient neural networks, like CNN, LSTM, GCN [219], etc., and their variant structures or combinations are designed to complete the feature learning task. Besides, there are some strategies used for recognition, such as capsule networks [220], and knowledge distillation [221].

2.9 Optical Flow Estimation

Optical flow is used to estimate pixel motion between video frames. In early stage, it is considered as an energy minimization problem aiming to obtain an ideal trade-off between feature similarities and motion smoothness [222, 223]. Major improvements in this era came from better designs of similarity and regularization terms. The problem is that it is hard to get precise flow fields by hand-crafted features and optimization objectives. The rise of deep neural networks significantly advanced this field. With the development of deep learning, researchers tried to avoid the optimization step and directly estimate optical flow [224] or learn more robust data terms [225].

FlowNet [226] was the first end-to-end convolutional network for optical flow estimation. Its extended work, FlowNet2.0 [227], adopted a stacked architecture with warping operation. To improve results on optical flow, many recent works introduce stronger learning paradigms, such as the coarse-to-fine method [228, 229], iterative refinement method [230, 231, 232], explicit pixel-wise-relation modeling, and joint representation learning with other tasks [233]. Based on RAFT [230] architecture, many works [234, 224, 235] were proposed to either reduce the computational costs or improve the flow accuracy. Some works used transformer architecture, like FlowFormer [236]. Recently, optical flow was extended to more challenging settings, such as low-light [237], foggy [238], and lighting variations [239].

CHAPTER 3

FACE MORPHING ATTACKS AND DEFENSE

In this chapter, face morphing attacks and defense work are discussed in detail. First, data used in this part is introduced in Section 3.1. Second, methodology and experimental result of face morphing attacks are described in Sections 3.2 and 3.3, respectively. And then methodology and experimental result of morphing attacks detection are presented in Sections 3.4 and 3.5. Finally, discussion and limitation are given in Section 3.6.

3.1 Database Construction

To simulate the data amount and distribution in real-world applications, we have combined five datasets to build a large-scale evaluation benchmark for few-shot morphing attack detection and fingerprinting. It contains four publicly available datasets, namely, FERET-Morphs [240, 241], FRGC-Morphs [242, 241], FRLL-Morphs [243, 244, 241], and CelebA-Morphs [245, 5, 71]. We also generated a new dataset with high-resolution faces collected from the web, named Doppelgänger Morphs, which contains morphing attacks from three algorithms and satisfies the so-called Doppelgänger constraint [27] (i.e., look-alike faces without biological connections). A total of over 20,000 images (6,869 bona fide faces and 15,764 morphed faces) have been collected. Eight morphing algorithms are involved, including five landmark based methods, OpenCV [3], FaceMorpher [4], LMA [5], WebMorph [6], and AMSL [244], and three generative adversarial networks based, including MorGAN [5], CIEMorGAN [71] and StyleGAN2 [8].

Table 3.1 shows the detailed information about these datasets in terms of morphing method, data size, and image resolution. Existing morphing methods used in the database can be classified into two categories: landmark-based and Generative Adversarial Networks (GAN) based. Fig. 3.1 provides some cropped face samples with real faces and morphed

Table 3.1: The hybrid face morphing database consists of five image sources and 3-6 different morphing methods for each.

Database	Subset	#Images	Resolution
FERET-Morphs	bona fide [240]	576	512x768
	FaceMorpher [241]	529	512x768
	OpenCV [241]	529	512x768
	StyleGAN2 [241]	529	1024x1024
FRGC-Morphs	bona fide [242]	964	1704x2272
	FaceMorpher [241]	964	512x768
	OpenCV [241]	964	512x768
	StyleGAN2 [241]	964	1024x1024
FRLL-Morphs	bona fide [243]	102+1932	413x531
	AMSL [244]	2175	413x531
	FaceMorpher [241]	1222	431x513
	OpenCV [241]	1221	431x513
	LMA	768	413x531
	WebMorph [241]	1221	413x531
	StyleGAN2 [241]	1222	1024x1024
CelebA-Morphs*	bona fide [245]	2989	128x128
	MorGAN [5]	1000	64x64
	CIEMorGAN [71]	1000	128x128
	LMA [5]	1000	128x128
Doppelgänger	bona fide	306	1024x1024
	FaceMorpher	150	1024x1024
	OpenCV	153	1024x1024
	StyleGAN2	153	1024x1024

* means only the cropped faces from raw images are used; no facial cropping is used for other datasets. The raw number of bona fide images in FRLL-Morphs is 102. Based on the raw faces, data augmentation is implemented to obtain extra 1932 images.

faces from different morphing algorithms in these five datasets. To the best of our knowledge, this is one of the largest and most diverse face morphing benchmarks, which can be used for both MAD and MAF evaluation.

FERET-Morphs and FRGC-Morphs are released by [241]. Both datasets are created by selecting similar looking pairs of people. A total of 529 face pairs from FERET [240] and 964 from FRGC v2.0 [242] are chosen finally. Two landmark-based morphing methods (OpenCV, FaceMorpher), and one Generative Adversarial Networks based method, StyleGAN2, are applied to generate morphs.

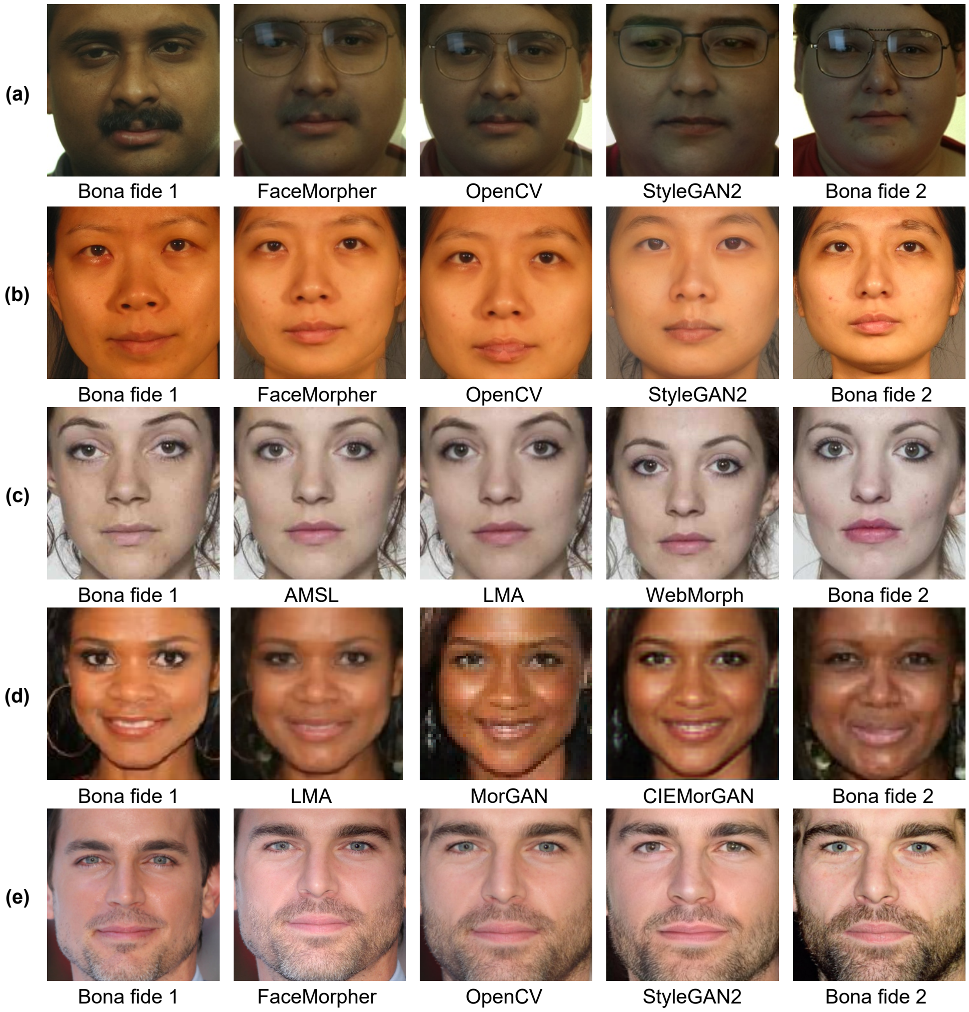


Figure 3.1: Face samples in five merged datasets. (a) FERET-Morphs (bona fide faces come from FERET [240]), (b) FRGC-Morphs (bona fide faces come from FRGC V2.0 [242]), (c) FRLL-Morphs (bona fide faces come from Face Research Lab London Set (FRLL) [243]), (d) CelebA-Morphs (bona fide faces come from CelebA [245]), and (e) Doppelgänger Morphs (bona fide faces come from the web collection).

FRLL-Morphs are generated based on the face images of FRLL [243]. The FRLL dataset is a great choice to use for creating morphing attacks, because it contains close-up frontal face images of high visual quality shot under uniform illumination with a large variety of ethnicity. The bona fide set contains 102 genuine neutral faces of 102 subjects (49 females, 53 males; 13 black people, 21 Asian, and 68 whites). The released version in [241] applied four types of morphing methods (OpenCV, FaceMorpher, WebMorph, StyleGAN2) to each preselected pair of faces.

AMSL is generated by Neubert et al. [244]. It contains 2,175 morphing faces based

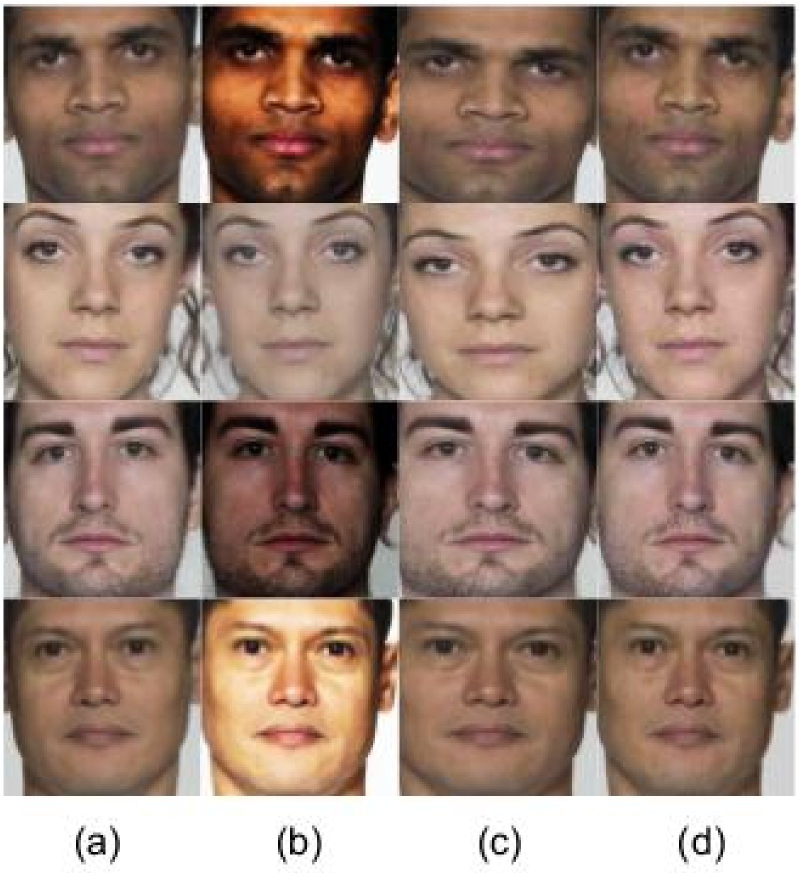


Figure 3.2: Some examples of augmented face images by (a) flipping horizontally, (b) changing brightness, (c) resizing, (d) JPEG compression.

on the selected pairs of face images in the genuine neutral face set. Besides, based on the FRLL bona fide faces, we generate a morphing subset using a morphing method LMA [5]. The numbers of morphing face in FRLL-Morphs is bigger than the original real faces (102). In order to get a balanced data set, several data augmentation strategies, like flipping horizontally, changing brightness with different parameters (linear, gamma, sigmoid), resizing to different sizes, JPEG compression using various factors, are taken to enlarge the scale of the genuine neutral faces. Finally, we get a total of 2,034 bona fide faces. Figure 3.2 shows several augmented face examples.

CelebA-Morphs is a morphing results collection generated from CelebA database [245]. CelebA contains ten thousand identities, enabling the selection of similar faces from a wide range of identities. The morphing results are created by LMA, MorGAN, and CIEMor-

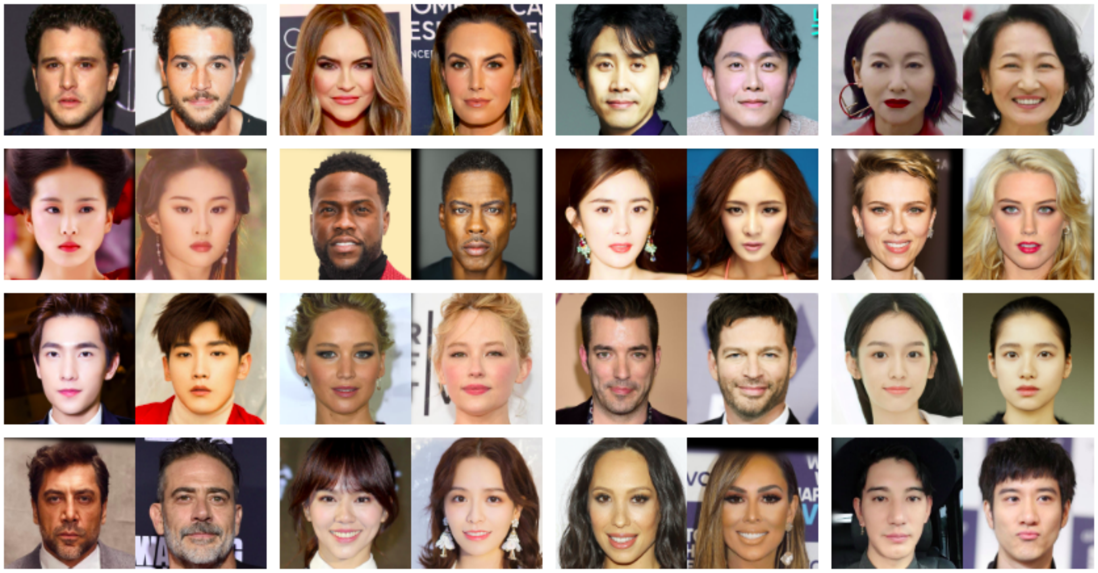


Figure 3.3: Some sample pairs of bona fide face images of Doppelgänger dataset.

.

GAN.

Doppelgänger Morphs is a morphing face dataset generated by the images we collect from the web. A name pair list is created to gather the faces of celebrities that appear similar, with same gender and ethnicity. Some samples are shown in Fig. 3.3. All faces are rotated to align the eyes on a horizontal line. Only one image per person is considered. Finally, we obtained 153 pairs. Three morphing methods (OpenCV, FaceMorpher, and StyleGAN2) are applied to generate morphed faces.

3.2 Morphing Attacks Methodology

This section introduces the transformer-based face image morphing pipeline and the technical details of our adaption to this pipeline.

3.2.1 Transformer-based GAN

Most existing GAN-based models adopt CNN as the basic architecture and rarely consider self-attention constructions. In this work, we have designed a transformer-based GAN model aiming to eliminate the blending artifacts, as well as, eliminate the manipulation in the latent space, resulting in more visibly realistic morphed faces. We applied the Generative Adversarial Transformer (GANformer) [18] as our backbone to generate high-quality morphing face images with 1024×1024 resolution by linearly interpolating the latent codes of the two input bona fide faces. The latent code is generated by improving the similarity between the input bona fide image and the embedded image created using a latent vector. In our work, we call the MorphGANFormer morphing model.

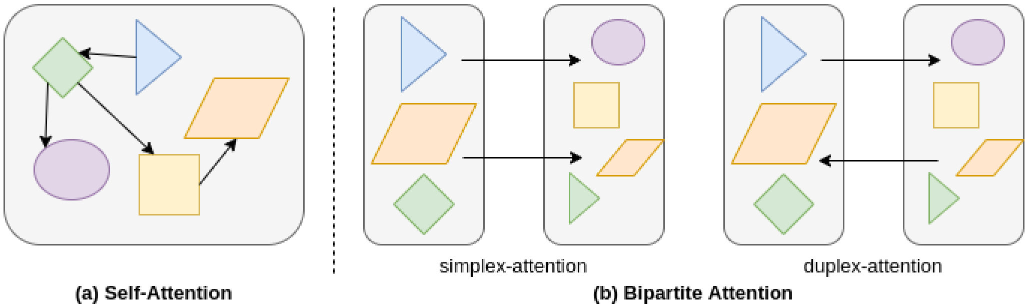


Figure 3.4: (a) Self-Attention and (b) Bipartite Attention. In comparison to self-attention, bipartite attention allows long-range interactions, and evades the quadratic complexity which self-attention suffers from.

MorphGANFormer contains a generator (G) that maps a sample from the latent space to an image, and a discriminator (D) that seeks to discern between real and fake images [246]. G and D compete with each other through a minimax game until they reach equilibrium [18]. The generator employs a bipartite structure, called bipartite transformer. Traditional transformer uses self-attention with pairwise connectivity, as shown in Fig. 3.4 (a). It is a highly-adaptive architecture centered around relational attention and dynamic interaction. However, the dense and potentially excessive pairwise connectivity causes quadratic mode of operation making it difficult to be extended to high-resolution input image. Bipartite transformer adopts a point-to-point mapping between individual latent components and different regions of evolving visual features, which can enable long-range interactions across the image and maintain the computation of linear efficiency, making scaling to high-resolution synthesis easy. Main idea is to iteratively propagate information from a set of latent variables to the evolving visual features and vice versa to support the refinement of each in light of the other.

Fig. 3.4 (b) shows two types of attention operations over the bipartite graph: simplex and duplex. Simplex attention permits communication in one direction, from the latents to the image features, while duplex attention enables both top-down and bottom up connections between latents and image features. In generateor, it iteratively propagates information between latent components and the image features bidirectionally, to support finer refinement. It can maintain computation of linear efficiency, making scaling to high-resolution synthesis is easy.

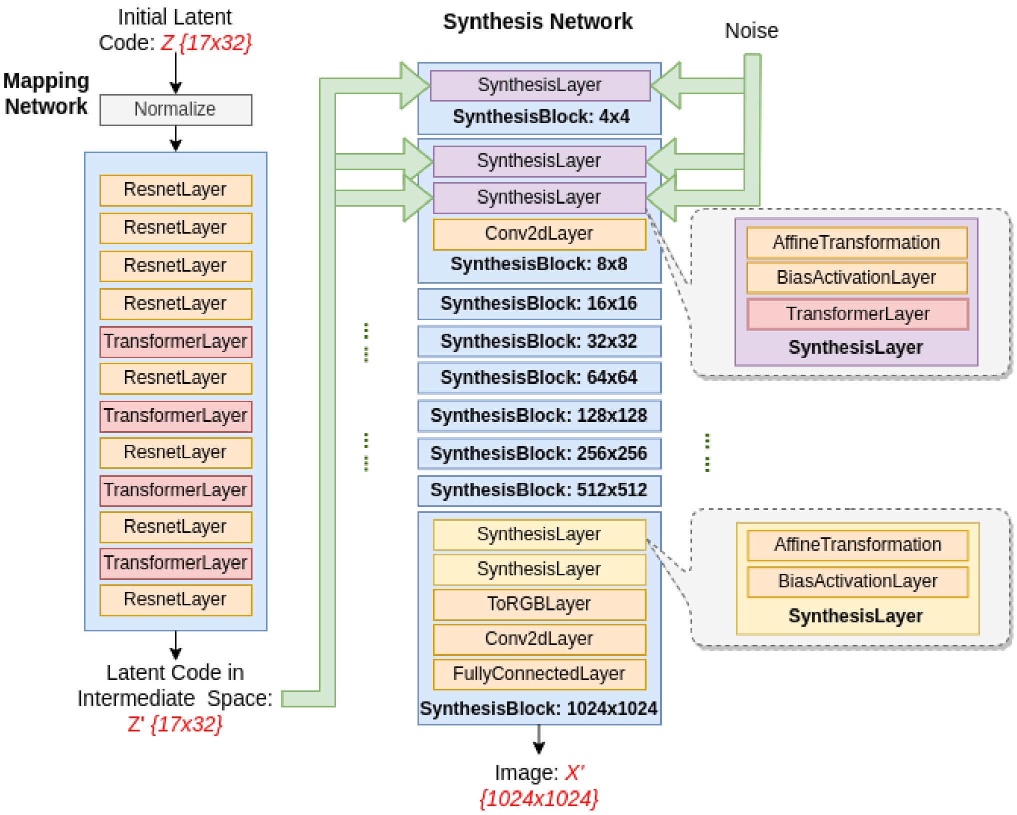


Figure 3.5: The architecture of generator G in MorphGANFormer, which contains a mapping network that maps a randomly sampled vector into a intermediate space and a synthesis network that generates a image based on the latent code.

The architecture of MorphGANFormer generator is illustrated in Fig. 3.5. It contains two parts: mapping network and synthesis network. The mapping network is composed of several feed-forward layers that receive a randomly sampled vector Z and output an intermediate vector Z', which in turn interacts directly with each transformer layer through the synthesis network with added noise to modulate the features of the evolving image. Finally, the intermediate vector Z' is transformed into an image X' as the output of the synthesis network.

The latent code Z, has the dimension of 17×32, denoted as [z1, z2, ..., z16, z17], in which [z1, ..., z16] are 16 components of the local-style latent code that are used to interact with the feature of the image through spatial attention, and z17 is a global-style component to globally modulate the feature of the image. The dimension of each component is 32×1. Figs. **??** (a) and (b) show the main difference in latent space between StyleGAN and MorphGANFormer. StyleGAN uses one global monolithic latent to impact the evolving image features of the whole scene uniformly, but in our work, we design a compositional latent space making the latent and image features attend to each other to capture the scene structure.

The synthesis network contains nine stacked synthesis blocks starting from a 4×4 grid and up to produce a final high-resolution image with 1024×1024 resolution. In a synthesis block, the bipartite (duplex) attention operation propagates information from the latent space to the image grid, followed by convolution and upsampling. Gaussian noise is added to each of the activation maps before the attention operations. A different sample of noise is generated for each block and interpreted on the basis of the scaling factors of that layer. The most important part of the synthesis block is the Synthesis Layer. For the first 8 blocks, the Synthesis Layer contains an affine transformation layer (translation, resizing, and rotation), a bias activation layer, and a transformer layer with bipartite attention operation. The blocks 16×16 to 512×512 have the same architecture as the block 8×8 which contains two Synthesis and one Conv2d layer. The Conv2d layer is the convolution layer with optional up-sampling or down-sampling. The last block removes the attention operation and adds an RGB layer to map the dense image features to RGB images.

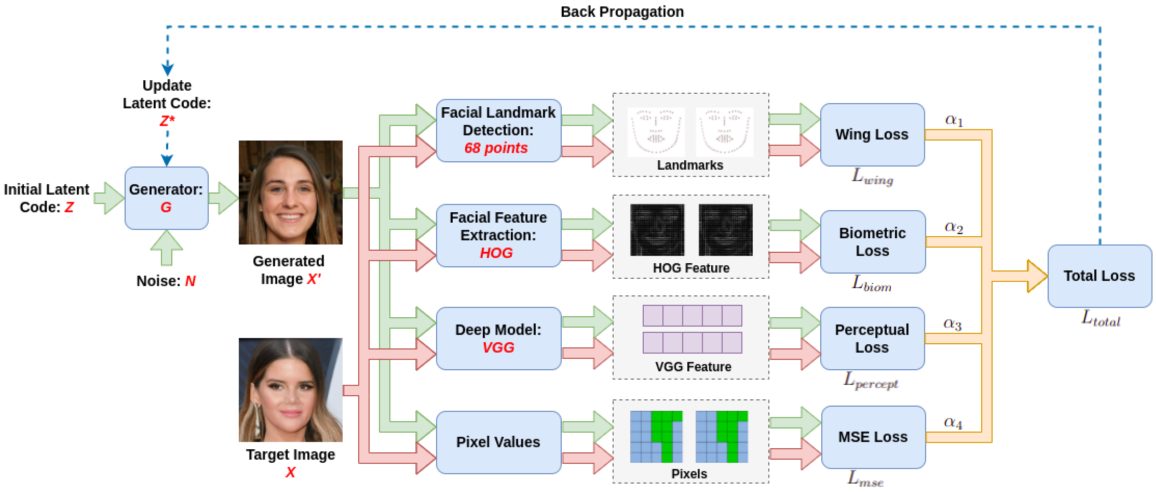


Figure 3.6: The pipeline of optimizing the latent code of the given face image.

3.2.2 Latent Code Learning

In StyleGAN [23, 8], it uses a latent code to control the style of all features globally. Although it can successfully disentangle global properties, it is more limited in its ability to perform spatial decomposition, as it does not provide a direct means to control the style of localized regions within the generated image. Luckily, the bipartite transformer offers a solution to meet this goal. Instead of controlling the style of all features globally, this attention layer can perform region-wise adaptive modulation. This approach achieves layer-wise decomposition of visual properties, allowing the model to control global aspects of the picture, such as pose, lighting conditions, or color schemes, in a coherent manner over the entire image.

In our method, we use the MorphGANFormer generator that is well trained in a large FFHQ face database [23] with a resolution of 1024×1024 as a basic module to obtain the latent code of the input image. The pipeline is shown in Fig. 3.6. The pipeline follows a pretty straightforward optimization framework used in [247, 248]. The bipartite attention operation can propagate information from the latent to the image grid, followed by convolution and upsampling. These are stacked multiple times starting from a 4×4 grid and up to 1024×1024 high-resolution images.

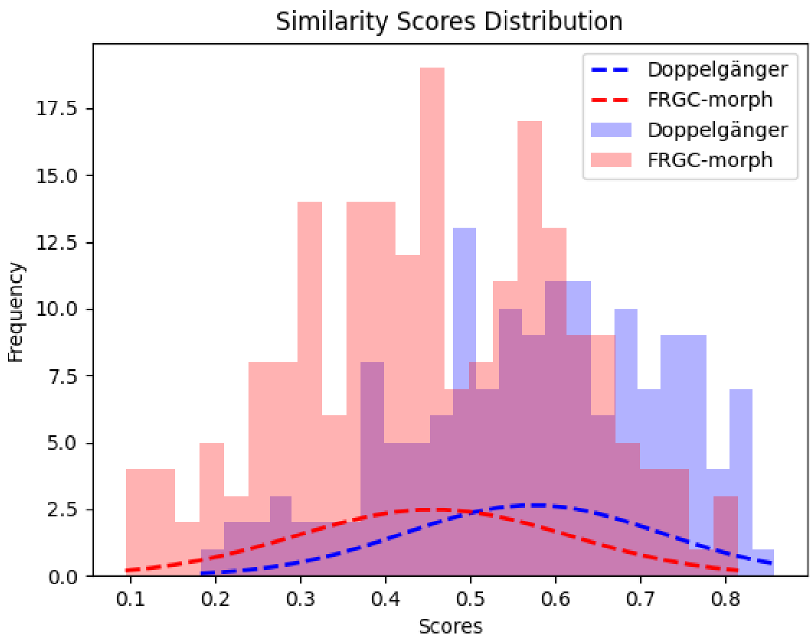


Figure 3.7: Similarity score distribution of bona fide pairs on Doppelgänger and FRGC-morph datasets.

3.2.3 Loss Function

To measure the similarity between the input image X and the generated image $G(Z)$ (X') using the learned latent code during optimization, we employ a loss function that is a weighted combination of the Wing Loss [249] based on facial landmarks, the biometric loss based on the distance of matching two faces, VGG-16 perceptual loss [250], and pixel-wise mean square error (MSE):

$$L_{total} = \alpha_1 L_{wing} + \alpha_2 L_{biom} + \alpha_3 L_{percept} + \alpha_4 L_{mse} \tag{3.1}$$

where α_1, α_2, α_3 and α_4 are weights.

We employ two loss functions related to face content. One is Wing Loss [249], which was originally proposed for facial landmark localization to improve deep neural network training ability for small and medium range errors in sample landmarks. The formula is defined as follows:

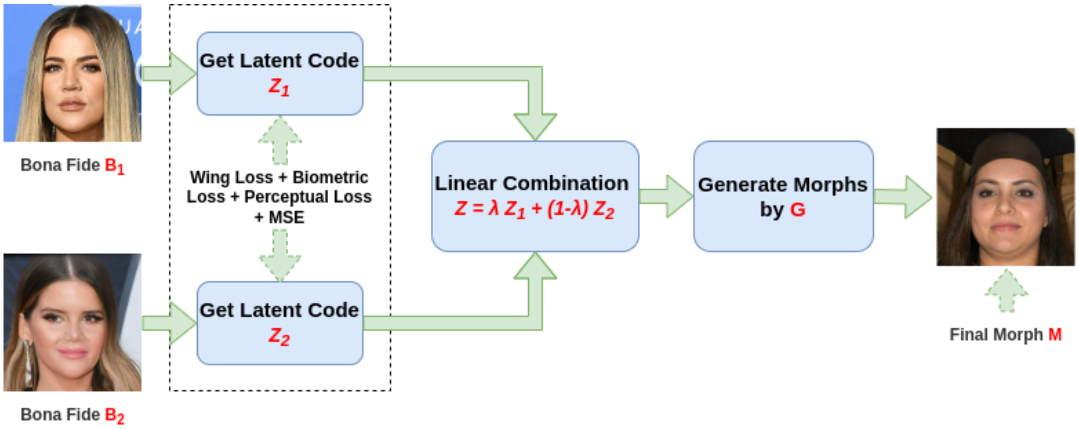


Figure 3.8: The pipeline of face morphing.

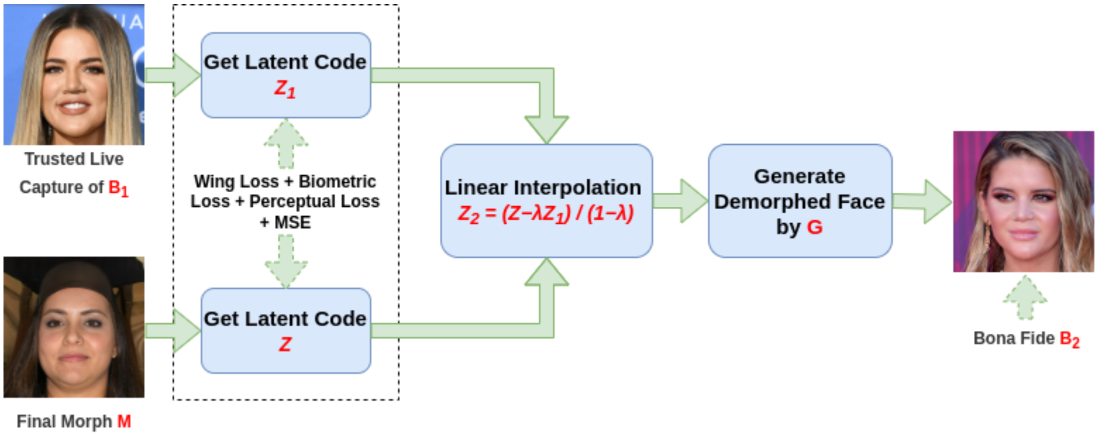

Trusted Live Capture of B1
Final Morph M
Get Latent Code
Z1
Wing Loss + Biometric Loss + Perceptual Loss + MSE
Get Latent Code
Z
Linear Interpolation
Z2 = (Z−λZ1) / (1−λ)
Generate Demorphed Face by G
Bona Fide B2

Figure 3.9: The pipeline of face demorphing.

$$L_{wing} = \begin{cases} \beta ln(1 + |x|/\epsilon) & if|x| < \beta \\ |x| - C & otherwise \end{cases} \tag{3.2}$$

where the nonnegative factor β sets the range of the nonlinear part to $(-\beta, \beta)$, ϵ limits the curvature of the nonlinear region, $|x|$ means the magnitude of the gradients between the landmark points of $G(Z)$ and X. C = $\beta - \beta ln(1 + \beta/\epsilon)$ is a constant that smoothly links the linear and nonlinear parts defined in part.

The other is biometric loss by calculating the matching distance of the faces. This loss is used to handle the biometric aspect of morphing and to make sure that the morphed faces are related to the original bona fide faces. The matching distance can induce a penalty for the generator during the latent code optimization process if the morphed outputs are not comparable to the original images in terms of biometric utility. The distance between two faces is calculated using the cosine similarity score based on the histogram of oriented gradients (HOG) [24] features, which can be defined as:

$$L_{biom} = 1 - \frac{HOG_{G(Z)} \cdot HOG_X}{\|HOG_{G(Z)}\| \|HOG_X\|}. \tag{3.3}$$

The study [251, 252] found that the learned filters of the VGG image classification model [253] are excellent general-purpose feature extractors, so they are used to measure the high-level similarity between images perceptually by the covariance statistics of the extracted features, which is formalized as perceptual loss [250]. For the perceptual loss term $L_{percept}$ in Eq. 3.1, we define it as:

$$L_{percept}(G(Z), X) = \sum_{j=1}^{4} \frac{\lambda_j}{N_j} \|F_j(G(Z)) - F_j(X)\|_2^2 \tag{3.4}$$

where $G(\cdot)$ is the well trained MorphGANFormer generator, Z is the latent code to optimize, $G(Z)$ is the embedded image, $X \in R^{n \times n \times 3}$ is the target image, N is the number of scalars in the image (i.e., $N = n \times n \times 3$), F_j is the output of the features of the VGG-16 layers conv1_1, conv1_2, conv3_2, and conv4_2, respectively, N_j is the number of scalars in the output of the j-th layer, λ_j is a factor. For the pixel-wise MSE loss term L_{mse}, it is defined as:

$$L_{mse}(G(Z), X) = \frac{1}{N} \|G(Z) - X\|_2^2. \tag{3.5}$$

The reason for choosing perceptual loss together with pixel-wise MSE loss is that pixel-wise MSE loss alone cannot easily find a high-quality latent vector. Perceptual loss can guide optimization to the right region of the latent space acting as a regularizer.

Given two face images B_1 and B_2, with their respective latent vectors Z_1 and Z_2, face morphing is calculated by linear interpolation:

$$Z = \lambda Z_1 + (1 - \lambda) Z_2, \lambda \in (0, 1) \tag{3.6}$$

and the final morphing result is generated from the generator G using the latent code Z. The commonly used λ is 0.5.

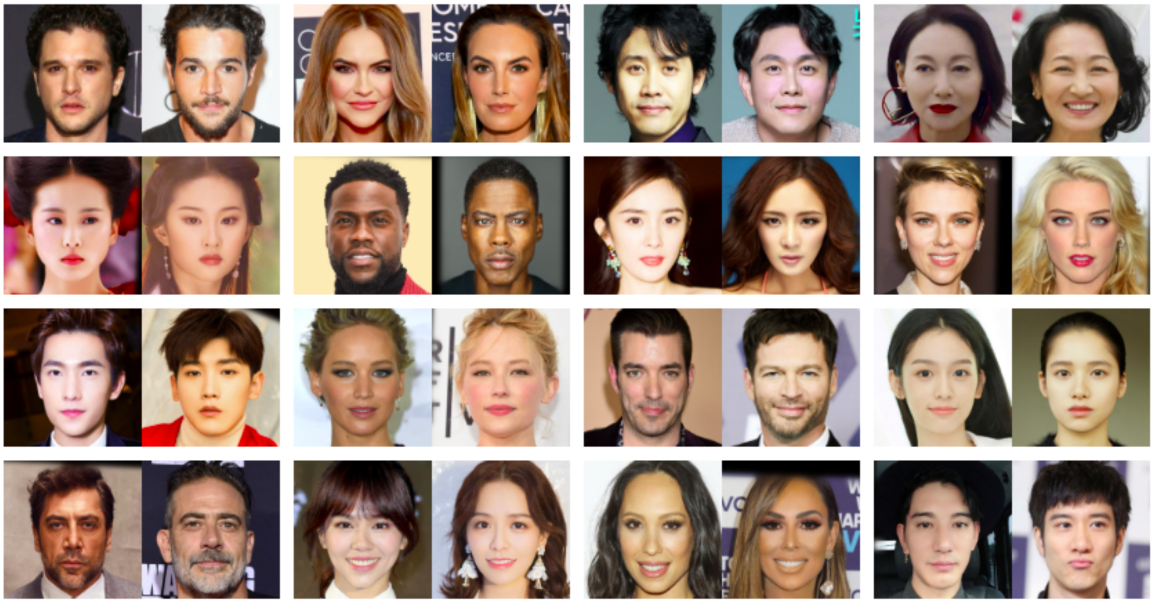


Figure 3.10: Some sample pairs of bona-fide face images from the Doppelgänger dataset (note that these look-alike pairs do not have biological connections).

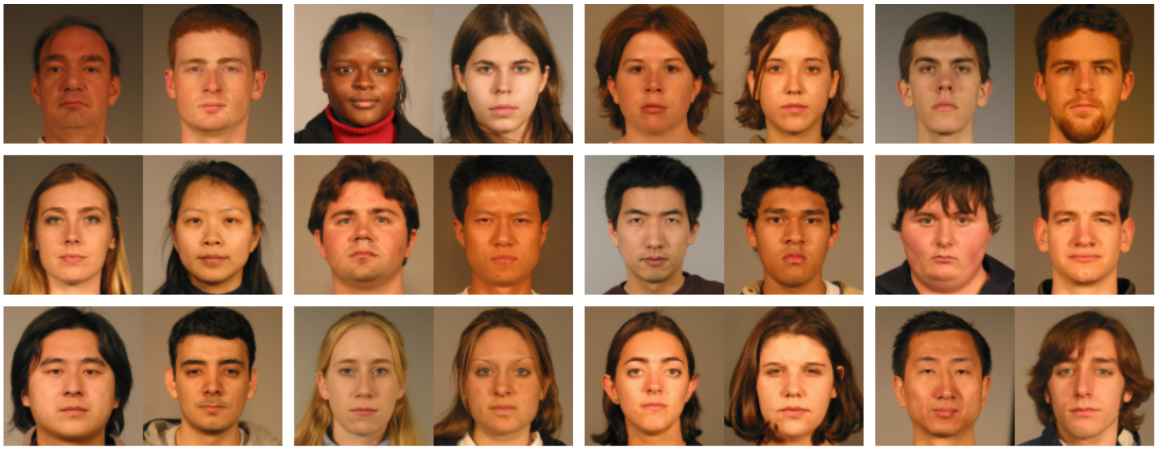


Figure 3.11: Some sample pairs of bona-fide face images from the FRGC-morph dataset.

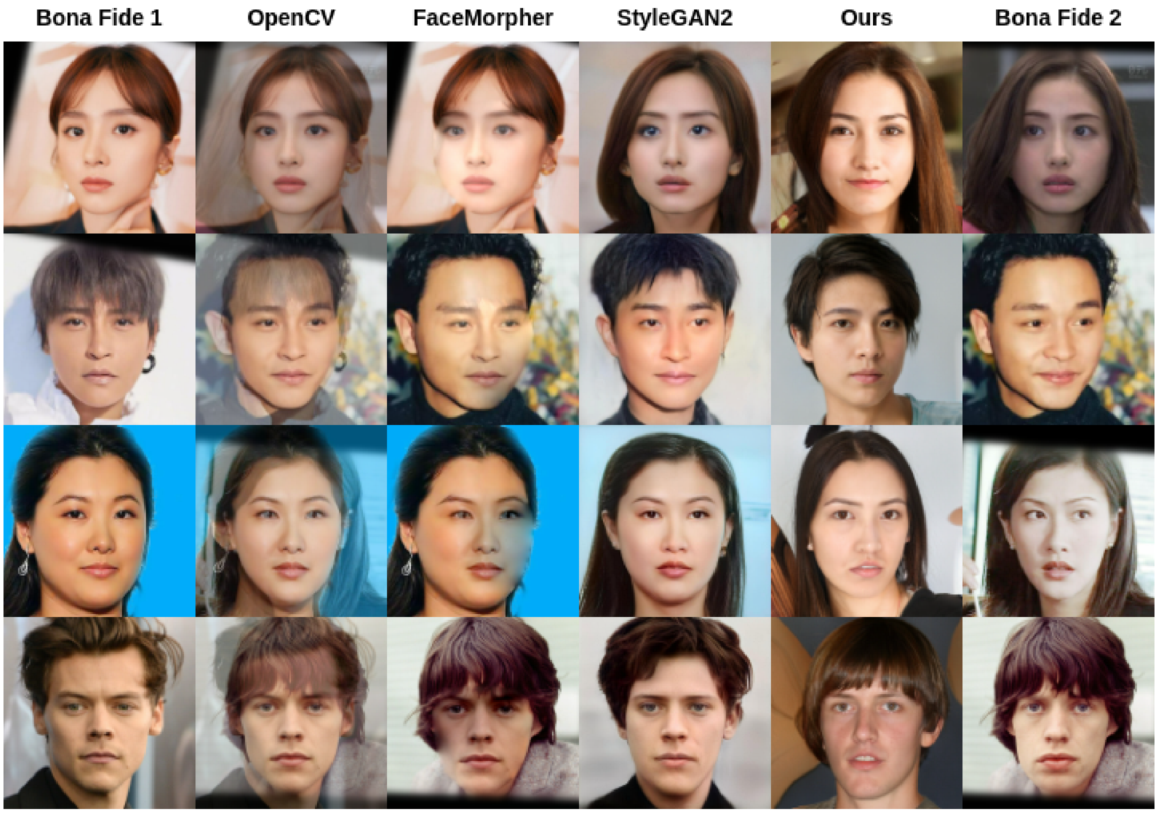


Figure 3.12: Face morphing results in the Doppelgänger Morphs database without any post-processing.

3.2.4 Face Morphing and De-Morphing

Figs. 3.8 and 3.9 show the main pipelines of face morphing and demorphing, respectively.

The basic idea of embedding a given image onto the manifold of the pre-trained generator is the following. With an initial latent code Z as the starting point, the model tries to find an optimized latent code Z^* that minimizes the loss function defined to measure the similarity between the target image and the image generated using Z^*. For the initialization of latent codes, we use the mean $\overline{Z}$ of 10,000 latent vectors that are randomly sampled from a uniform distribution of [-1,1], and we expect the optimization to converge to a vector Z^* so that the generated image X' has high similarity to the target image X. We also consider noise-space optimization [254] to complement latent-space embedding, which further improves quality.

The basic idea of demorphing [25] is to try to reverse the morphing process. In the morphing attack, a morphed image can be treated as a linear combination $M = B_1 + B_2$, where B_1 and B_2 are the bona fide faces of two subjects. In a general face verification process without a morphing attack, M can be treated as a combination of two identical face images of one person. In the morphing attack situation, during the face verification process, the system receives $\hat{B_1}$, a live captured variant of B_1, and the demorphing task is to calculate the demorphed image $\hat{B}2$ by removing $\hat{B}1$ from M, which is $\hat{B}2 = M - \hat{B}1$.

Given the live trusted capture of one bona fide face image B_1 and the morphed face image M, with their respective latent vectors Z_1 and Z, face demorphing is calculated in latent space by:

$$Z_2 = \frac{Z - \lambda Z_1}{(1 - \lambda)}, \lambda \in (0, 1) \tag{3.7}$$

and final demorphing result is generated from the generator G using the latent code Z_2.

3.3 Morphing Attacks Experiments

3.3.1 Database Description

Table 3.2 presents the database used in our experiment: the newly constructed Doppelgänger face morphing database and reconstructed FRGC-morph dataset. Both are composed of bona fide faces, corresponding trusted live captures, four types of morphing results via OpenCV, FaceMorpher, StyleGAN2 and our MorphGANFormer.

Figs. 3.10 and 3.11 shows some pairs of bona fide face images from Doppelgänger and FRGC-morph dataset. Note that for the former we are guaranteed that the pair will look similar; for the latter, we have adopted a strategy of random pairing so the likelihood of obtaining two similar bona fide images is low.

We use the real images in two databases as bona fide faces. The first is the Doppelgänger dataset in which a name-pair list is created to gather the faces of celebrities that look alike, with the same gender and ethnicity. All faces are rotated to align the eyes on

Table 3.2: The data used in our experiment. One is the newly constructed Doppelgänger face morphing database and the other one is reconstructed FRGC-morph dataset.

Database	Subset	#Number	Resolution
Doppelgänger	bona fide	153 pairs	1024x1024
	trusted live captures	306	1024x1024
	FaceMorpher	150	1024x1024
	OpenCV	153	1024x1024
	StyleGAN2	153	1024x1024
	MorphGANFormer	153	1024x1024
FRGC-morph	bona fide	204 pairs	1024x1024
	trusted live captures	408	1024x1024
	FaceMorpher	204	1024x1024
	OpenCV	204	1024x1024
	StyleGAN2	204	1024x1024
	MorphGANFormer	204	1024x1024

a horizontal line. Only one image per identity is considered. Finally, we obtained 153 pairs (95 female; 58 male) with the size of 1024×1024 resolution. The second dataset is constructed from FRGC [242]. All faces are cropped, aligned, and resized to 1024×1024 resolution. Subjects with the same gender are randomly selected to compose bona fide pairs for face morphing. Each subject is selected only once. Finally, we get 204 pairs (112 male and 92 female). For both datasets, we obtain one extra image for each subject as a trusted live capture for de-morphing task. Fig. 3.7 illustrates the different distributions of similarity scores between two bona fide faces per pair in the Doppelgänger and FRGC-morph datasets using FaceNet [255] feature, which shows that the Doppelgänger pairs have higher similarity scores than the FRGC-morph.

3.3.2 Experimental Setup

For the latent code initialization, we use the mean $\overline{Z}$ of 10,000 latent vectors that are randomly sampled from a uniform distribution of [-1,1]. For perceptual loss, we choose pretrained VGG-16 as the backbone network to extract image feature. For Wing loss, we use dlib toolbox [256] to detect 68 facial points for calculation. For the distance between the

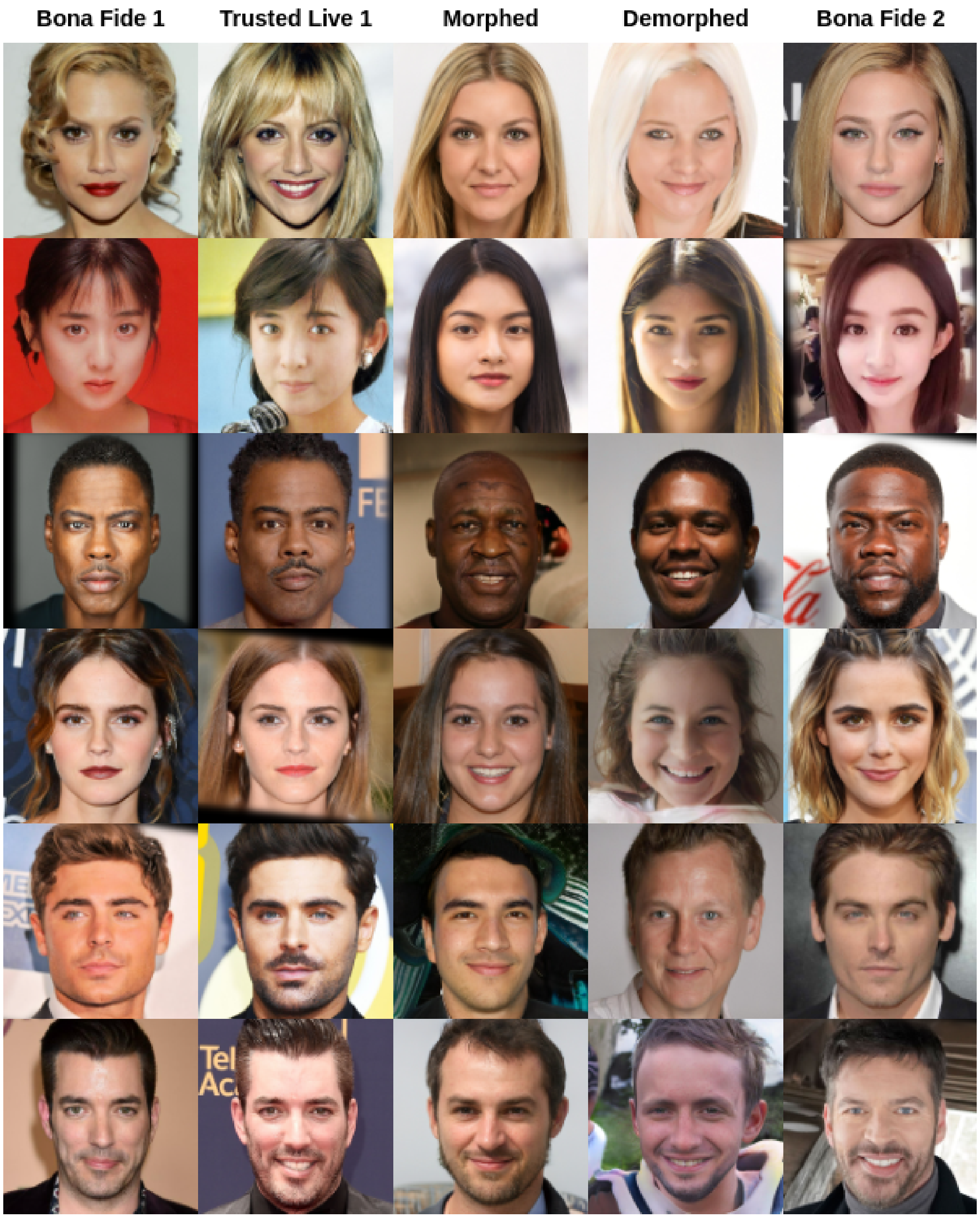


Figure 3.13: Some demorphed results on Doppelgänger dataset.

two faces, we use HOG feature [24] of the faces to calculate the matching score. We use Adam optimizer with a learning rate of 0.01 to optimize the latent code learning procedure with α_1=0.02, α_2=1.0, α_3=1.0, and α_4=1.0 for loss functions. We set 1,500 gradient descent steps for the optimization, and keep the latent code with the lowest loss value for generation.

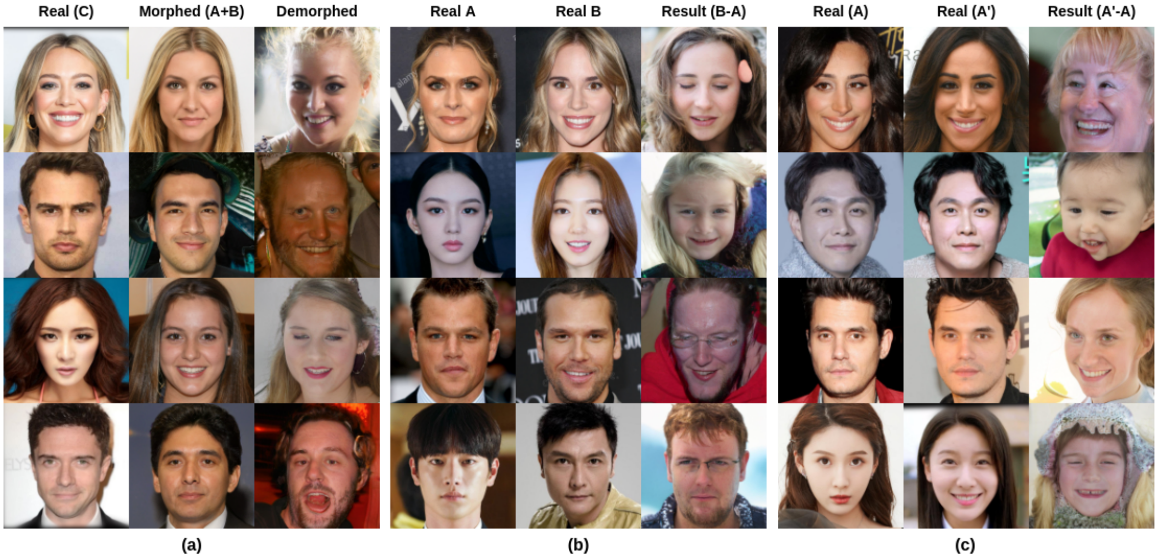


Figure 3.14: Some demorphing results using different inputs on Doppelgänger dataset. (a) The inputs are morphed faces combined by identity A and B, and trusted live captures of identity C; (b) The inputs are real faces of identity B as morphed images, and real faces of identity A as trusted live captures; (c) The inputs are real faces A' as morphed images, and the other real faces A of the same identity as trusted live captures.

Table 3.3: MMPMR (%) on Doppelgänger and FRGC-morph database.

Dataset	Morph Type	ArcFace	FaceNet	LBP
Doppelgänger	OpenCV [3]	94.73	82.23	87.50
	FaceMorpher [4]	81.21	73.83	87.92
	StyleGAN2 [8]	84.21	70.65	85.52
	MorphGANFormer	90.08	70.92	89.77
FRGC-morph	OpenCV [3]	87.75	74.51	94.61
	FaceMorpher [4]	80.39	72.06	85.78
	StyleGAN2 [8]	38.73	35.78	78.43
	MorphGANFormer	48.04	42.65	84.80

3.3.3 Vulnerability Test

We evaluate the vulnerability of three face recognition models to the morphing attacks created by our morphing framework. ArcFace [259] introduced Additive Angular Margin loss to improve the discriminative ability of the face recognition model. It scored state-of-the-art performance on several face recognition evaluation benchmarks such as Labeled Faces in the Wild (LFW) [262] 99.83% and YouTube Face (YTF) [263] 98.02%. We use an Arc-

Table 3.4: MMPMR (%) with ablation study on Doppelgänger database.

Loss	ArcFace	FaceNet	LBP
$Biom_{FaceNet}$	56.58	50.53	82.11
$Biom_{ArcFace}$	53.29	47.24	80.79
$Biom_{LBP}$	50.66	43.95	90.00
$Biom_{HOG}$	77.63	45.92	86.71
Percept	53.29	43.95	78.82
Percept+Wing	82.24	59.08	88.68
Percept+Wing+MSE	84.87	62.37	89.34
$Biom_{HOG}$+Percept	86.18	59.74	88.03
$Biom_{HOG}$+Percept+Wing	85.53	61.05	88.03
$Biom_{HOG}$+Percept+Wing+MSE	90.08	70.92	89.77

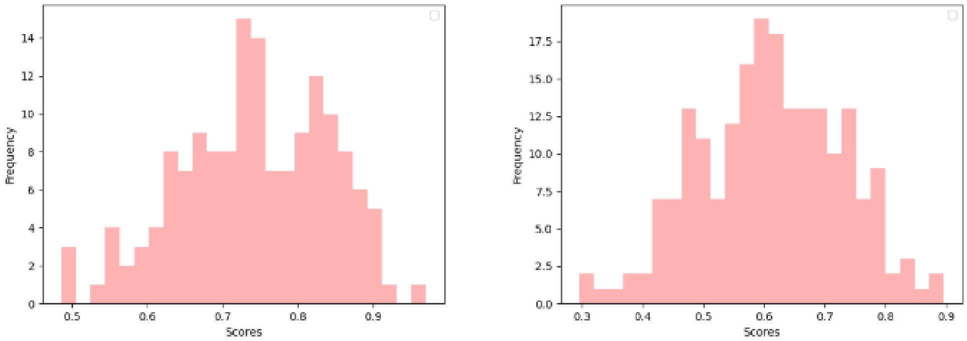
Table 3.5: Performance (%) comparison of MAD on OpenCV, FaceMorpher, StyleGAN2, and Our Method. Accu. - Accuracy.

		OpenCV [3]			FaceMorpher [4]			StyleGAN2 [8]			**MorphGANFormer**		
Dataset	MAD Method	Accu.	D-EER	ACER	Accu.	D-EER	ACER	Accu.	D-EER	ACER	Accu.	D-EER	ACER
Doppelgänger	MobileNetV2 [257]	66.45	36.18	49.50	66.00	42.36	50.82	66.45	37.50	49.51	65.57	59.87	50.82
	NasNetMobile [258]	68.64	35.53	43.42	65.12	45.02	49.26	62.50	45.56	52.63	61.84	65.13	53.62
	ArcFace [259]	66.23	40.13	40.79	62.91	46.35	46.11	59.43	46.88	50.99	58.77	51.97	51.97
	MB-LBP [260]	66.67	44.24	47.53	67.99	43.02	46.09	67.11	45.39	46.88	64.47	51.32	50.82
	FS-SPN [76]	48.68	44.74	43.59	45.47	47.67	48.15	50.00	42.11	41.61	44.96	50.66	49.18
	MixFaceNet-MAD [261]	67.76	34.21	33.55	63.36	39.54	40.31	57.02	50.66	49.67	57.89	46.71	48.36
FRGC-morph	MobileNetV2 [257]	44.28	28.43	42.16	44.12	36.27	42.40	44.77	18.26	41.42	33.33	57.35	58.58
	NasNetMobile [258]	71.57	29.53	32.60	69.93	32.84	35.05	68.46	33.82	37.25	59.48	49.02	50.74
	ArcFace [259]	66.34	43.63	46.94	65.36	44.73	48.41	66.67	37.25	46.45	68.79	38.73	43.26
	MB-LBP [260]	67.16	43.75	46.57	66.67	42.65	47.30	63.73	51.72	54.90	66.50	49.02	47.55
	FS-SPN [76]	55.72	46.57	47.43	54.90	47.06	48.65	72.06	24.02	22.92	58.33	45.59	43.50
	MixFaceNet-MAD [261]	67.48	33.33	39.71	65.52	40.20	42.65	62.91	44.12	46.57	61.11	49.02	49.26

Face model based on ReseNet-100 [84] architecture pre-trained on a refined version of the MS-Celeb-1M dataset (MS1MV2) [264] to extract face features. FaceNet [255] directly learns an embedding mapped from input to an Euclidean space in which the Euclidean distance indicates the similarity of the face. It uses triplets of tightly cropped face patches generated by an online triplet mining method to train the network, and its output is a compact 128-D embedding. Local Binary Pattern (LBP) [265] is a hand-crafted feature that describes the texture characteristics of surfaces. By applying LBP, the probability of the texture pattern can be summarized into a histogram. It is a commonly used feature in face recognition domain.

Dlib face detector [256] is used to segment the face region. The cropped face is normalized according to the eye coordinates and resized to a fixed size of 224×224 pixels.

The single feature extraction (ArcFace, FaceNet, and LBP) procedure is performed on the processed faces. Ideally, a strong morphing attack will have a similar and high similarity score to the target identities. We present the vulnerability results in a quantifiable manner by giving the Mated Morphed Presentation Match Rate (MMPMR) [28] based on the decision threshold at the false match rate (FMR) of 0.1%. Note that all vulnerability results are presented on the testing data.



(a) Doppelgänger

(b) FRGC-morph

Figure 3.15: Similarity score distribution between restored faces and real faces of the bona fide on (a) Doppelgänger and (b) FRGC-morph datasets based on FaceNet feature.

Table 3.3 shows the MMPMR (%) values of different morphing methods using ArcFace, FaceNet and LBP features. And Fig. 3.12 shows some morphing samples in the Doppelgänger database. We can see that for landmark-based morphing attacks, like OpenCV and FaceMorpher, it has high MMPMR values, indicating it highly preserves the characteristic of both bona fide identities, but the image artifacts caused by blending on image level are obvious too. In contrast, GAN-based morphing methods improve the visual quality of morphed images. However, synthetic-like generation artifacts, as shown in the StyleGAN2 attack, make morphing faces less realistic and natural. Our model has the same or even better ability to preserve the facial identities as landmark-based models and can also generate visually realistic and natural faces.

We also did an ablation study with different loss functions on Doppelgänger dataset as shown in Table 3.4. The first part shows some results using different facial features to calculate the face matching distance. From the second and third parts, we can see that, with the combination of more loss functions, the MMPMR value increases.

3.3.4 Detectability Analysis

To thoroughly evaluate the detectability of MorphGANFormer attacks, we selected several popular methods used in face recognition [259], pre-trained deep models [266, 257, 258, 267] on ImageNet [268], and existing morphing attack detection methods [260, 76, 73, 261, 33], for comparison. We measure the attack detection performance on our generated attacks, and other types of attacks, like OpenCV [3], FaceMorpher [4], and StyleGAN2 [8], based on the bona fide faces in Doppelgänger and FRGC-morph databases.

We evaluate the detectability of our attacks as unknown attacks, i.e., novel attacks unknown to the detection algorithm. In this case, the training data come from the attacks of LMA [5], WebMorph [6], AMSL [70], MorGAN [5] and CIEMorGAN [71] attacks introduced in [34], and their corresponding bona fide faces, which contains 1,838 images (bona fide: 918; morphed: 920) in total. The test data are from Doppelgänger (153 morphed + 306 bona fide) and FRGC-morph (204 morphed + 408 bona fide) datasets, respectively. We trained a binary classifier using the training data. After the detector is well trained, it is used to predict bona fide and our MorphGANFormer attacks (or OpenCV [3], FaceMorpher [4], StyleGAN2 [8] attacks).

Following previous morphing attacks detection (MAD) studies [78, 269], we report performance using accuracy, D-EER, and ACER. Detection Equal-Error-Rate(D-EER) is the error rate for which both BPCER and APCER are identical. The average classification error rate (ACER) is calculated by the mean of the APCER and BPCER values. The attack presentation classification error rate (APCER) reports the proportion of morph attack samples incorrectly classified as bona fide presentation, and the Bona Fide Presentation Classification Error Rate (BPCER) refers to the proportion of bona fide samples incorrectly classified as morphed samples. The results are shown in Table 3.5. Compared to the

Table 3.6: Demorphing accuracy (%) on Doppelgänger and FRGC-morph.

	ArcFace	FaceNet	LBP
Doppelgänger Pairs	54.90	62.75	88.24
FRGC-morph Pairs	29.94	37.25	68.14

OpenCV, FaceMorpher, and StyleGAN attacks, the MorphGANFormer attacks are more challenging. Unlike vulnerability, we note that the detectability performance gap between the Doppelgänger and FRGC datasets is small.

3.3.5 Performance of De-Morphing

To quantitatively evaluate the performance of the demorphing result, ArcFace, FaceNet, and LBP are adopted to compare the restored facial image $\hat{B}_2$ with B_2 and B_1, respectively. When the system determines that $\hat{B}_2$ matches B_2, but does not match B_1, the demorphing is considered successful. We use a restoration accuracy introduced in FD-GAN [92] as a measure metric to check the demorphing performance. In our paper, we termed restoration accuracy as demorphing accuracy. The demorphing accuracy is defined as the percentage of the number of successfully demorphed facial images in the total number of demorphed facial images. The decision threshold for similarity scores is set as the value of the false match rate (FMR) at 0.1$\%$. Table 3.6 shows the result.

Fig. 3.13 shows some results of face demorphing on Doppelgänger dataset. We use morphed face and one trusted live capture of bona fide 1 to restore the face of bona fide 2, as shown in column 'Demorphed'. It can be clearly seen that demorphed image has a good resemblance to the face of bona fide 2, justifying the effectiveness of our defense strategy in the latent space.

Fig. 3.14 shows some results using randomly selected inputs to do demorphing. Fig. 3.14 (a) uses a morphed face generated by bona fide A and B, and the trusted live capture from a third identity C, as input. Fig. 3.14 (b) uses a real face image of identity B as morphed face to be input to the demorphing model, and the other real face image of identity

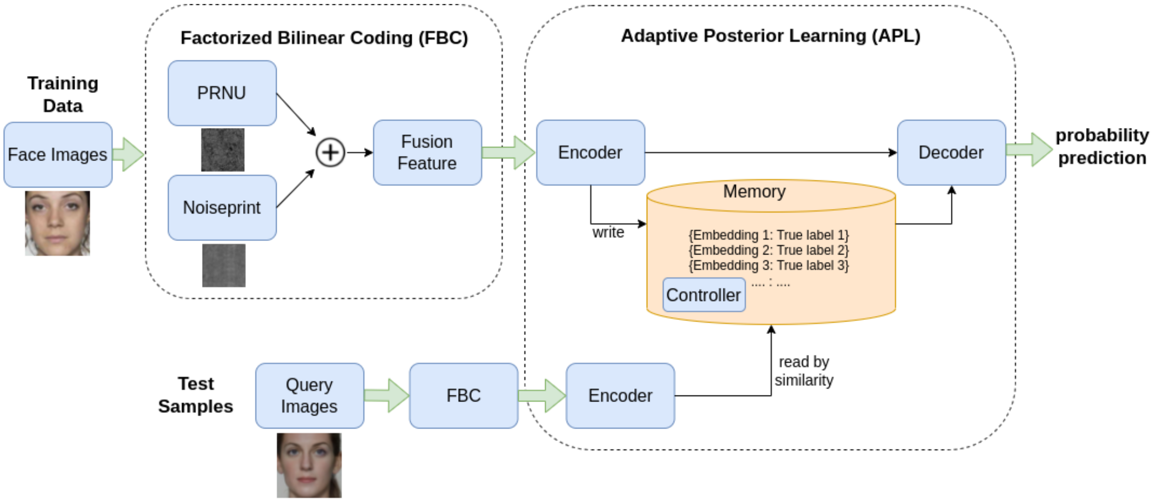


Figure 3.16: An overview of the proposed system (FBC-APL).

A as the trusted live capture. Fig. 3.14 (c) applies two face images of the same identity as inputs. The demorphed results are various and uncontrollable with low quality. Obvious artifacts can be easily spotted.

Fig. 3.15 presents the similarity scores distribution between the demorphed faces of bona fide 2 and real faces of bona fide 2 on two datasets based on FaceNet feature. It can be seen that demorphing can achieve reasonably good matching scores on both datasets, implying the detectability of our defense strategy in the latent space. Between Doppelganger and FRGC, we observe that FRGC has lower matching scores than Doppelganger, suggesting less vulnerability. The choices of bona fide pair for face morphing, which is related to the trade-off between detectability and vulnerability, deserves further systematic study.

3.4 Morphing Defense Methodology

This section introduces the fusion-based few-shot morphing attack detection and fingerprinting method. Fig. 3.16 shows the overall detection system consisting of two stages: feature extraction/fusion via factorized bilinear coding (FBC) and few-shot learning (FSL) for MAD/MAF. We will first elaborate fusion-based MAD in detail, and then discuss the extension into few-shot MAF.

3.4.1 Fusion-based Single-Image MAD

Noise is often embedded into image data during acquisition or manipulation. The uniqueness of the noise pattern is determined by the physical source or artificial algorithm, which can be characterized as a statistical property to reveal the source of noise [270]. The sensor pattern noise was first used for MAD task by making facial quantification statistics analysis, confirming its effectiveness [76]. Here, we consider two types of sensor noise patterns: Photo Response Non-Uniformity (PRNU) [271] and Noiseprint [33].

Photo Response Non-Uniformity (PRNU). PRNU originates from slight variations among individual pixels during the photoelectric conversion in digital image sensors [32]. Different image sensors embed this weak signal into the acquired images as a unique signature. Even though the weak signal itself is mostly imperceptible to the human eye, its uniqueness can be characterized by statistical techniques and exploited by sophisticated fingerprinting methods such as PRNU [271]. This systemic and individual pattern, which plays the role of a sensor fingerprint, has turned out robust to various innocent image processing operations such as JPEG compression. Even though PRNU is stochastic in nature, it is a relatively stable component of the sensor over its life span.

PRNU has been widely studied in camera identification because it is unrelated to the image content and present in every image acquired by the same camera. Most recently, PRNU has been proposed as a promising tool for detecting morphed face images [73, 74]. The spatial feature of PRNU can be extracted using the approach presented by Fridrich [271]. For each image I, the noise residual W_I is estimated as described in Equation 3.8:

$$W_I = I - F(I) \tag{3.8}$$

where F is a denoising function which filters out the sensor pattern noise. Clever design of mapping function F (e.g., wavelet-based filter [32]) makes PRNU an effective tool for various forensic applications.

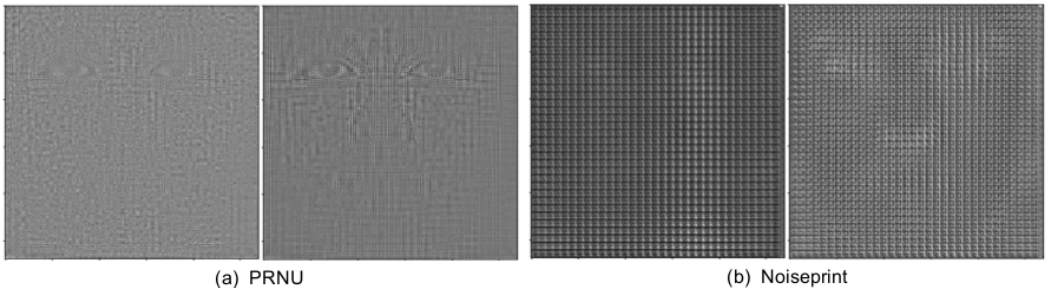


Figure 3.17: The average of (a) PRNU feature and (b) Noiseprint feature over 1000 randomly selected face images. Left column: bona fide; Right column: morphed face.

Noiseprint. Unlike model-based PRNU, data-driven or learning-based methods tackle the problem of camera identification by assuming the availability of training data. Instead of mathematically constructing unique signatures, Noiseprint [33] attempts to learn the embedded noise pattern from training data. A popular learning methodology adopted by Noiseprint is to construct a Siamese network [272]. The Siamese network is trained with pairs of image patches coming from the same or different cameras in an unsupervised manner. Similar to PRNU, Noiseprint has shown clear traces of camera fingerprints. It is worth noting that Noiseprint has performed better than PRNU when the cropped image patches become smaller, implying the benefit of exploiting spatial diversity [33].

To the best of our knowledge, Noiseprint has not been proposed for MAD in the open literature. Existing deep-learning based S-MAD often use pretrained networks such as VGG-face [78]. Our empirical study shows morphing-related image manipulation does leave evident traces in Noiseprint, suggesting the feasibility of Noiseprint-based MAD. Moreover, morphed faces are often manipulated on the whole face, whose spatial diversity can be exploited by cropping image patches by Noiseprint. To justify this claim, Fig. 3.17 (b) presents the Noiseprint comparison between bona fide and morphed faces averaged over 1,000 examples. Visual inspection clearly shows that the areas around the eye and nose present more significant (bright) traces than bona fide faces. By contrast, Fig. 3.17 (a) shows the comparison of the extracted PRNU patterns with the same experimental setting. Similar visual differences between bona fide and morphed faces can be observed; more importantly, PRNU and Noiseprint demonstrate complementary patterns (low vs. high

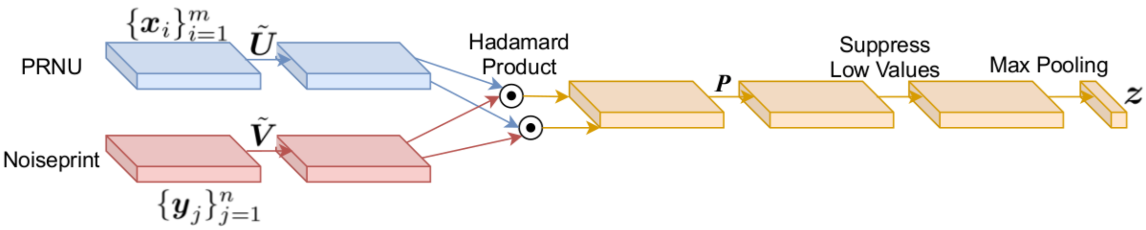


Figure 3.18: The architecture of FBC module. $\tilde{\boldsymbol{U}}$ and $\tilde{\boldsymbol{V}}$ are replacement of $\boldsymbol{U}$ and $\boldsymbol{V}$ to avoid numerically instable matrix inversion operations; $\boldsymbol{P}$ is a fixed binary matrix.

frequency) begging for fusion.

Feature Fusion Strategy. Fusion methods are usually based on multiple feature representations or classification models. By exploiting the diversity, the strategy of combining classifiers [273] has demonstrated improved recognition performance than single modality approaches. Recent works have shown that fusion methods based on Dempster-Shafer theory can improve the performance of face morphing detectors [274]. However, previous work [274] only considered scale space ensemble models and pretrained CNN models. For the first time, we propose to combine PRNU and Noiseprint by a recently developed similarity-based fusion method, called factorized bilinear coding (FBC) [275].

FBC is a sparse coding formulation to generate a compact and discriminative representation with substantially fewer parameters by learning a dictionary $\boldsymbol{B}$ to capture the structure of the whole data space. It can preserve as much information as possible and activate as few dictionary atoms as possible. Let $\boldsymbol{x}_i$, $\boldsymbol{y}_j$ be the two features extracted from PRNU and Noiseprint respectively. The key idea behind FBC is to encode the extracted features based on sparse coding and to learn a dictionary $\boldsymbol{B}$ with k atoms by matrix factorization. Specifically, sparsity FBC opts to encode the two input features $(\boldsymbol{x}_i, \boldsymbol{y}_j)$ into FBC code $\boldsymbol{c}_v$ by solving the following optimization problem:

$$\min_{\boldsymbol{c}_v} \left|\left| \boldsymbol{x}_i \boldsymbol{y}_j^\top - \sum_{l=1}^{k} c_v^l \boldsymbol{U}_l \boldsymbol{V}_l^\top \right|\right|^2 + \lambda ||\boldsymbol{c}_v||_1 \tag{3.9}$$

where λ is a trade-off parameter between the reconstruction error and the sparsity. The dictionary atom b_l of $\boldsymbol{B}$ is factorized into $\boldsymbol{U}_l \boldsymbol{V}_l^\top$ where $\boldsymbol{U}_l$ and $\boldsymbol{V}_l^\top$ are low-rank matri-

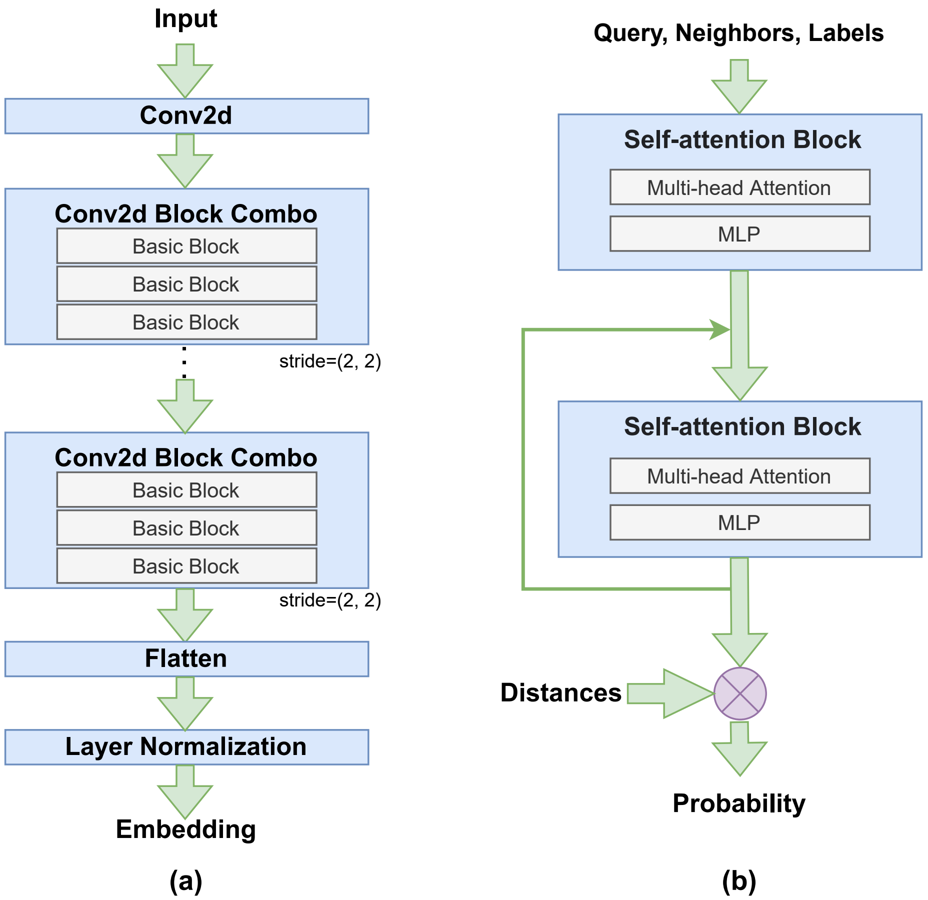


Figure 3.19: The architecture of few-shot learning (FSL) module: (a) encoder, (b) decoder.

ces. The l_1 norm $||\cdot||_1$ is used to impose the sparsity constraint on $\boldsymbol{c}_v$. In essence, the bilinear feature $\boldsymbol{x}_i\boldsymbol{y}_j^\top$ is reconstructed by $\sum_{l=1}^{k} c_v^l \boldsymbol{U}_l \boldsymbol{V}_l^\top$ with $\boldsymbol{c}_v$ being the FBC code, and c_v^l representing the l-th element of $\boldsymbol{c}_v$.

The above optimization can be solved by well-studied methods such as LASSO [276]. With two groups of features $\{\boldsymbol{x}_i\}_{i=1}^{m}$ and $\{\boldsymbol{y}_j\}_{j=1}^{n}$ at our disposal, we first compute all FBC codes $\{\boldsymbol{c}_v\}_{v=1}^{N}$ and then fuse them by max operation to attain the global representation $\boldsymbol{z}$:

$$\boldsymbol{z} = max\,\{\boldsymbol{c}_v\}_{i=1}^{N}\,. \tag{3.10}$$

The whole FBC module is shown in Fig. 3.18.

3.4.2 Few-shot learning for MAD/MAF

Based on the FBC-fused feature $\boldsymbol{z}$, we construct a few-shot learning module as follows. Inspired by the recent work of adaptive posterior learning (APL) [100], we have redesigned the FSL module to adaptively accept the feature vectors of any size (e.g., FBC-fused feature) as input. This newly designed module consists of three parts: an encoder, a decoder, and an external memory store. The encoder is used to generate a compact representation for the incoming query data; the memory saves the previously seen representation by the encoder; the decoder aims at generating a probability distribution over targets by analyzing the query representation and pairwise data returned from the memory block. We will elaborate on the design of these three components next.

Encoder. The encoder can convert the input data with any size to a compact embedding with low dimensionality. It is implemented by a convolutional network as shown in Figure 3.19(a), which is composed of a single first convolution to map the input to 64 feature channels, followed by 15 convolutional blocks. Each block is made up of a step of Batch Normalization, followed by a ReLU activation and a convolutional layer with kernel size 3. For every three blocks (one combo), the convolution contains a stride 2 to downsample the image. All layers have 64 features. Finally, the feature is flattened to a 1D vector and passed through Layer Normalization, generating a 64-dimensional embedding as encoded representation.

Memory. The external memory store is a database to store experiences. It is key-value data. Each row represents the information for one data point. Each column is decomposed into an embedding (encoded representation) and a true label. The memory store is managed by a controller deciding which embeddings can be written into the memory, at the same time, trying to minimize the amount of written embeddings. During the writing process, a quantity metric surprise is defined. The higher the probability the model assigns to the true class correctly, the less surprised it will be. If the prediction confidence in the correct class is smaller than the probability assigned by a uniform prediction, the embed-

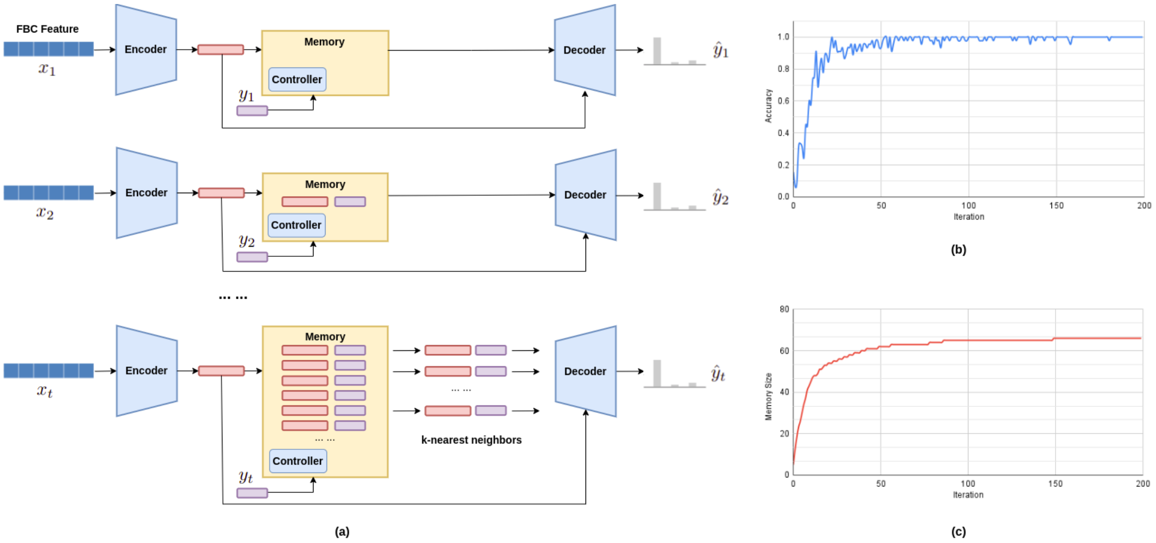


Figure 3.20: (a) APL training procedure over iterations. We train the APL module over a sequence of episodes (x_t, y_t), where x_t is FBC feature and y_t is true label. The memory is empty at the beginning. At each iteration, a batch of samples is fed to the module and a prediction is made. Cross-entropy loss L($\hat{y}_t$, y_t) is calculated, and a gradient update step is performed to minimize the loss on that batch alone. The loss is also fed to the memory controller for the network to decide whether to write to memory. (b) and (c) show the behavior of accuracy and size of memory in 9-class training scenario. APL stops writing to memory after having about 7 examples per class for classification.

ding should be written to memory. During the querying process, the memory is queried for k-nearest neighbors of the embeddings of queries from the encoder. The distance metric used to calculate the proximity between points is an open choice, and here we use two types (euclidean distance, cosine distance). Both the full row data for each of the neighbors and query embeddings are concatenated and fed to the decoder.

Decoder. The decoder takes the concatenation of query embedding, recalled neighbor embeddings from memory, labels and distances, as input. The architecture is a relational feed-forward module with self-attention. It processes each of the neighbors individually by comparing them with the query, and then does a cross-element comparison with a self-attention module before reducing the activations with an attention vector calculated from neighbor distances. The self-attentional blocks are repeated five times in a residual manner. The resulting tensors are called activation tensors. Besides, the distances between neighbors and query are passed through a softmax layer to generate an attention vector, which is summed with the activation tensor over the first axis to obtain the final logit result for classification. As shown in Figure 3.19 (b), the self-attentional block comprises a multi-head attention layer, Multi-Head Dot-Product attention (MHDPA) [277], for cross-element comparison, and a multi-layer perceptron (MLP) nonlinear layer to process each element individually.

Training. During APL training as shown in Fig. 3.20 (a), the query data (i.e., FBC-fused feature vector z), is passed through the encoder to generate an embedding, and this representation is used to query an external memory store. The memory is empty in the beginning. At each training episode, a batch of examples is shown to the model and a prediction is made. Cross-entropy loss is used to be fed to the memory controller to decide whether to write to memory. After the query is searched in memory, the returned memory contents as well as the query are fed to the decoder for classification. Fig. 3.20 (b) and (c) show the behavior (accuracy and memory size) of APL during one single episode. The accuracy of APL increases as it sees more samples and saturates at some point, indicating that additional inputs do not surprise the module any more. In the case of 9-class classification scenario, we have observed that about 7 examples per class are sufficient to reach the performance saturation.

3.4.3 Morphing Attack Fingerprinting

Morphing attack fingerprinting (MAF) refers to the multiclass generalization of the existing binary MAD problem. In addition to detecting the presence of morphing attacks, we aim at finer-granularity classification about the specific model generating the face morph. It is hypothesized that different attack models inevitably leave fingerprints in the morphed images (conceptually similar to the sensor noise fingerprint left by different camera models [32]).

Both PRNU [32] and Noiseprint [33] were originally proposed for identification of camera models, which has known to be fingerprinting in image forensics. The duality between image generation in the cyber and physical worlds inspires us to extend existing problem formulation of binary MAD [73, 74, 75, 76] into multiclass fingerprinting. Different camera models (e.g., Sony vs. Nikon) are analogous to varying face morphing methods (e.g., LMA [5] vs. StyleGAN2 [8]); therefore, it is desirable to go beyond MAD by exploring the feasibility of distinguishing one morphing attack from another. Fortunately, the system as shown in Fig. 3.16 easily lends itself to the generalization from binary to multiclass classification by resetting the hyperparameters, like the number of classes, data path for each class, etc. To learn a discriminative FBC feature for fingerprinting, a multiclass labeled data for training and testing should be prepared to be fed to the FBC module for retraining. When the FBC feature is available, it will be fed to the APL module for multiclass classification.

3.5 Morphing Defense Experiments

3.5.1 Evaluation Protocols

Based on the large-scale dataset collected for few-shot MAD and MAF benchmarks, we have designed the evaluation protocols for each task as follows:

• Protocol FS-MAD (few-shot MAD). This protocol is designed for the few-shot binary classification (bona fide/morphed). The training data comes from predefined types and a few (1 or 5) samples per new type. Test data comes from new types. Here, predefined types in our experiment contain five types of morphing results generated by FaceMorpher [4], OpenCV [3], WebMorph [6], StyleGAN2 [8] and AMSL [244], and their corresponding bona fide faces. The faces of these types are from FERET-Morphs, FRGC-Morphs, FRLL-Morphs and Doppelgänger Morphs datasets. The morphing faces generated by LMA [5], MorGAN [5] and CIEMorGAN [71], and their corresponding bona fide faces are treated as new types. The faces of these types are from CelebA-Morphs dataset.

• Protocol FS-MAF (few-shot MAF). This protocol is designed for the multiclass classification of fingerprinting on the hybrid large-scale benchmark and five separate morph

Table 3.7: Traditional MAD performance (Accuracy-%) comparison of different feature-level fusion methods. NP - Noiseprint; CN - Concatenation; CC - Convex Compression; ⊥ - spatial; □ - spectral.

Feature	CN	Sum	Max	CC	FBC (ours)
PRNU ⊥ + PRNU □	83.78	84.23	83.78	84.23	84.42
NP ⊥ + NP □	89.19	89.64	89.64	89.64	96.40
PRNU ⊥ + NP □	89.19	89.19	89.64	89.19	89.59
PRNU □ + NP ⊥	83.78	84.23	83.78	85.59	86.04
PRNU □ + NP □	86.94	85.59	85.59	86.94	84.68
PRNU ⊥ + NP ⊥	**91.44**	**91.89**	**91.89**	**94.59**	**96.85**

Table 3.8: Performance (%) comparison of few-shot MAD.

	1-shot			5-shot		
Method	Accu.	D-EER	ACER	Accu.	D-EER	ACER
Xception [266]	66.5	32.5	33.5	73.25	27	26.75
MobileNetV2 [257]	67	36.5	33	71.25	29	28.75
NasNetMobile [258]	59	40.5	41	66.25	35	33.75
DenseNet121 [267]	68.25	31.5	31.75	73.5	24.5	26.5
FaceNet [255]	66.75	30	33.25	66.75	30.5	33.25
ArcFace [259]	58	41	42	62.25	37.5	37.75
Meta-Baseline [278]	60.45	-	-	71.38	-	-
COSOC [101]	66.89	-	-	74.54	-	-
FBC-APL	**99.25**	**1.5**	**0.75**	**99.75**	**0.5**	**0.25**

datasets. Each morphing type and the bona fide type are treated as different categories, namely, FERET-Morphs, FRGC-Morphs, CelebA-Morphs, and Doppelgänger datasets all with 4 classes, FRLL-Morphs with 7 classes, while the hybrid with 9 classes. For each dataset, the data is split by the rule of 8:2. Training data consists of 1 and 5 images per class for 1-shot and 5-shot learning, respectively. Testing data contains nonoverlapping data with training in each dataset. To reduce the bias of imbalance distribution of data, similar numbers of faces for each class in each test set are maintained.

3.5.2 Experimental Settings

Data Preprocessing. Dlib face detector [256] is used to detect and crop the face region. The cropped face is normalized according to the eye coordinates and resized to a fixed size of 270×270 pixels. The feature extraction of PRNU and Noiseprint are conducted on the processed faces, respectively. The resulting vector dimension for each type of feature is 72,900 (270×270).

Performance Metrics. Following the previous studies of MAD [78, 269], we report the performance using four metrics, including: (1) Accuracy; (2) D-EER; (3) ACER; (4) Confusion Matrix. Detection Equal-Error-Rate(D-EER) is the error rate for which both BPCER and APCER are identical. Average Classification Error Rate (ACER) is calculated by the mean of APCER and BPCER values. Attack Presentation Classification Error Rate (APCER) reports the proportion of morph attack samples incorrectly classified as bona fide presentation, and Bona Fide Presentation Classification Error Rate (BPCER) refers to the proportion of bona fide samples incorrectly classified as morphed samples. Both APCER and BPCER are commonly used in previous studies of MAD [78, 269].

3.5.3 Comparison of Feature Fusion Strategies

We first compare different feature-level fusion strategies to combine the PRNU and Noiseprint patterns, including element-wise operation (sum/max), convex compression (CC) [279], vector concatenation, and our factorized bilinear coding (FBC) method [275]. We consider feature in both spatial and spectral domains. The PRNU and Noiseprint features extracted from images are treated as spatial features. The spectral features are obtained by applying discrete Fourier transform to spatial features. Any two types of features are fused to carry out traditional MAD tasks on a subset of the test data. Therefore, six different fusion features are generated. For concatenation, the final feature dimension is 145,800. For sum, max, and CC, it is 72,900. The fusion feature of FBC is as compact as 2,048-dimensional. All generated features are fed into SVM with a linear kernel for binary classification. As shown in Table 3.7, the spatial feature fusion of PRNU and Noiseprint performs the best for all six features, which can be attributed to the fact that the two patterns in the spatial domain contain more discriminative features (as shown in Fig. 3.17). Further, our FBC

Table 3.9: Accuracy(%) of 1-shot MAF classification on single and hybrid datasets.

Method	FERET-Morphs	FRGC-Morphs	FRLL-Morphs	CelebA-Morphs	Doppelgänger	Hybrid
	4-class	4-class	7-class	4-class	4-class	9-class
Xception [266]	29.47	25.26	17.68	16.67	21.05	15.11
MobileNetV2 [257]	31.58	33.68	31.3	55.19	25.26	17.33
NasNetMobile [258]	32.63	27.37	22.61	19.26	23.16	12.88
DenseNet121 [267]	46.32	26.32	22.03	47.04	23.16	19.33
FaceNet [255]	26.79	27.98	16.48	33.67	31.15	15.67
ArcFace [259]	29.33	39.64	26.12	28.33	18.03	15.22
Meta-Baseline [278]	51.05	51.44	34.77	61.43	33.43	53.46
COSOC [101]	54.58	64.37	35.22	63.19	34.3	59.55
FBC	96.93	98.83	94.06	99.5	56.67	96.11
FBC-all	98.11	99.48	98.42	100	84.17	96.78
FBC-APL	**98.82**	**99.61**	**98.24**	**99.67**	**91.67**	**98.11**

Table 3.10: Accuracy(%) of 5-shot MAF classification on single and hybrid datasets.

Method	FERET-Morphs	FRGC-Morphs	FRLL-Morphs	CelebA-Morphs	Doppelgänger	Hybrid
	4-class	4-class	7-class	4-class	4-class	9-class
Xception [266]	46.32	43.16	31.01	73.7	28.42	43.67
MobileNetV2 [257]	55.79	53.68	40	89.26	26.32	54.56
NasNetMobile [258]	48.42	40	24.35	67.41	27.37	37.33
DenseNet121 [267]	54.74	55.79	36.23	89.26	25.26	53.33
FaceNet [255]	23.16	35.79	15.94	40	30.53	18.11
ArcFace [259]	44.34	50.91	33.81	39.67	20.49	29.11
Meta-Baseline [278]	60.6	64.72	50.74	81.42	36.8	61.98
COSOC [101]	65.98	75.04	54.9	89.6	41.81	72.62
FBC	97.64	99.09	96.94	99.5	65.83	96.22
FBC-all	98.11	99.48	98.42	100	84.17	96.78
FBC-APL	**98.82**	**99.61**	**98.24**	**99.67**	**96.67**	**98.22**

based fusion achieves the highest accuracy among the five fusion strategies.

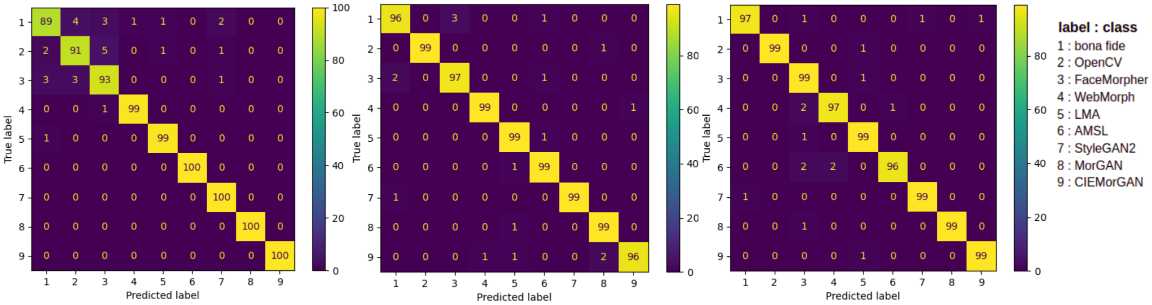
3.5.4 Few-shot Learning for MAD

We extend the traditional MAD problem to a few-shot learning problem. First, the PRNU and Noiseprint features are extracted respectively. Then a FBC module (VGG-16 [82] as backbone) is trained as a binary classifier for feature fusion by taking PRNU and Noiseprint features of the whole training set (all images of predefined types) as input. Based on the pretrained FBC module, 2,048-dimensional fusion representations are generated and then fed to the APL module for binary few-shot learning using the Cross Entropy loss. Here, the Euclidean distance is used to query the top 5 nearest neighbors from memory component. The output of the APL is a tuple of probability distribution for each class. The results in terms of Accuracy, D-EER, and ACER are shown in Table 3.8. Two FSL-based methods

[101, 278], two face recognition (FR) based methods [255, 259] and several popular pre-trained deep models [266, 257, 258, 267] on ImageNet [268], are adopted for comparison. Thanks to the effective fusion of two complementary patterns (i.e., PRNU and Noiseprint) and the APL module, our proposed FBC-APL clearly outperforms other competing methods by a large margin. We have also compared the performance of FBC and FBC-APL on datasets with different sizes as ablation studies.

3.5.5 Few-shot Learning for MAF

Different from few-shot MAD problem, in MAF, the FBC module uses ResNet50 [84] as backbone, and is pretrained as a nine-class classifier using all training data (about 80%) of the collected database. The obtained FBC fusion feature of the training samples is then fed to the APL module for multiclass few-shot learning. A cosine similarity score is adopted to compute the similarity between queries and embeddings stored in memory to find the top 3 nearest neighbors. From Table 3.9, 3.10 and Fig. 3.21, one can see that our FBC-APL has achieved outstanding performance, and some results are even better than FBC-all method which uses FBC features of all training data to fit SVM for classification.



(a) FBC-all

(b) FBC-FSL-1-shot

(c) FBC-FSL-5-shot

Figure 3.21: Confusion matrix of few-shot MAF classification on hybrid dataset.

3.6 Discussion and Limitation

The overall pipeline in Fig. 3.16 can be further optimized by end-to-end training. In our current implementation, the three steps are separated - i.e., the extraction of PRNU and Noiseprint features, FBC-based fusion, and APL-based FSL. From the perspective of network design, an end-to-end training could further improve the performance of FBC-APL model. Moreover, there still lacks large-scale and more challenging datasets for morphing attacks in the public domain. Validation of the generalization property for FBC-APL model remains to be finished, especially when novel face morphing attacks (e.g., transformer-based and 3D reconstruction-based face morphing) are invented.

In morphing attack work, we designed a transformer based morphing attack generation model which uses a weighted sum of four different loss functions: biometric loss, wing loss, perceptual loss and MSE. It can eliminate the blending artifacts and reduce the manipulation artifacts in the latent space, resulting in more visibly realistic and natural faces compared to previous works. The result can be further optimized by introducing more specific loss functions, which can effectively improve the similarity between given image and generated image by latent code. An interesting aspect of this work is that not all design choices lead to good results and that experimenting with the design choices provides further insights into the embedding.

Face morphing attacks have received increasing attention in recent years. Generation approaches such as GAN-based are among the leading techniques. However, existing methods suffer from noticeable blurring and synthetic-like generation artifacts. In this paper, we designed a transformer-based alternative to face morphing, which demonstrated its superiority to StyleGAN-based methods. Four particular loss functions were employed to maximize the similarity between the generated face image and the target face image. We also extended the study of transformer-based face morphing to demorphing, the dual operation. Future work includes an improved understanding of the trade-off between vulnerability and detectability as well as other morphing approaches such as diffusion models [280].

CHAPTER 4

FACIAL ANALYSIS ON AUTISM VIDEOS

In this chapter, face analysis technology is considered in autism diagnosis problem. First, an unique autism video database used in this part is introduced in Section 4.1. Second, methodology and experimental result of a three-class facial trait classification method is described in Sections 4.2 and 4.3, respectively. And then methodology and experimental result of a facial micro-expression analysis model on ASD and control group estimation is presented in Sections 4.4 and 4.5. Finally, discussion and limitation are given in Section 4.6.

4.1 CalTech ADOS Video Dataset Construction

For ASD diagnosis, we utilize a video dataset of ASD evaluations conducted using the Autism Diagnostic Observation Schedule (ADOS) [37], which is a structured but natural discussion between the interviewer and participant. It captures the complicated and rich behaviors of ASD in adults. We describe the unique database in detail, including details about the ADOS, how ADOS videos were acquired, participant information, video data information, ADOS scoring, and labels, and differences compared to other existing databases. All participants provided written informed consent using procedures approved by the Institutional Review Board (IRB) of West Virginia University (WVU) and California Institute of Technology (CALTECH).

4.1.1 Autism Diagnostic Observation Schedule (ADOS)

The Autism Diagnostic Observation Schedule (ADOS) is considered the clinical gold standard for the diagnosis of ASD. It consists of standardized activities that allow the examiner to observe the occurrence or nonoccurrence of behaviors that have been identified as important to the diagnosis of autism. Structured activities and materials, as well as less structured interactions, provide standardized contexts in which social and communicative behaviors are observed. The response of participants to each activity is recorded by highly trained interviewers, and the interactions between the interviewer and participant are videotaped. The interviewer provides a detailed scoring of multiple facets of ASD after completing the ADOS. These scores are used to formulate a diagnosis through the use of a diagnostic algorithm.

In effect, the ADOS interviews provide a one-hour observation period, during which an examiner presents the individual being assessed with numerous opportunities to exhibit behaviors of interest in the diagnosis of autism through standard presses for communication and social interaction [281]. Presses consist of planned social interactions in which the ADOS evaluators are likely to elicit specific behavioral responses to differentiate those with ASD.

4.1.2 ADOS Interview Participants and Video Acquisition

For the recruited individuals, thirty-three participants completed the ADOS from which the video data was generated. The reasons include that some recorded videos have low quality, and some recordings were just used to serve as practice for the examiners. Participants (age range = $16 \sim 37$ years; 26 males, 25.00 years; 7 females, 22.86 years), were primarily right-hand dominant (n=31). Nine participants were interviewed twice (first visit and return visit). The time span between the two interviews is about half a year. All participants had a diagnosis of ASD, informed by the Autism Diagnostic Observation Schedule, Second Edition (ADOS-2) [282], and confirmed by expert clinical judgment. They met the cutoff scores for ASD on the ADOS-2 revised scoring system for Module 4 [283]. The ADOS-2 was scored according to the latest method, and we also derived calibrated severity scores (CSS) for exploratory correlation analysis. (1) social affect (SA): 8.29 ± 4.55 (mean ± SD); (2) restricted and repetitive behavior (RRB): 2.43 ± 1.50; (3) severity score for social affect (CSS-SA): 6.0 ± 2.52; (4) severity score for restricted and repetitive behavior (CSS-RRB): 5.95 ± 2.40; (5) severity score for social affect plus restricted and repetitive behavior (CSS-All): 5.64 ± 2.79. The ASD group had a full-scale IQ (FSIQ) of 96.83 ± 13.48 (from the Wechsler Abbreviated Scale of Intelligence-2), a mean age of 24.05 ± 5.14 years.

The videos were acquired at the California Institute of Technology. The research diagnosis of ADOS is a lab routine for all participants with ASD. These videos were recorded for autism individuals separately in a quiet room at the hospital using a video camera of 9.1 Mbps data rate, 30 frames per second frame rate, and 720×480 image resolution. Both the examiner and the participant sit at a table, and the participant was asked to face the camera as much as possible when he or she can during recording. The captured information contains the participant's body behaviors, face emotions, hand gestures, eye contact, speech traits (volume, pacing), reciprocal social exchange with the examiner, etc.

The ADOS videos include 15 interview sections (scenes) between a clinician and a person suspected of having ASD. The tasks included in ADOS Module 4 include: (1) Construction task: the participant uses puzzle pieces to complete a diagram and is instructed to request more pieces when needed; (2) Telling a story from a book: since the book has very few words, the participant interprets the story from the visual cues including reading emotions on the faces of the people in the story; (3) Description of a Picture; the picture provides opportunities for interaction with the interviewer and to gauge spontaneous language; (4) Conversation and Reporting: based largely on the picture the participant saw in (3); (5) Current Work or School: a series of questions about these aspects of their life; (6) Social Difficulties and Annoyance: discussion about social interactions and how they perceive them; (7) Emotions: talking about the events/objects that elicit different emotions in the participant and asking them to describe their feelings; (8) Demonstration Task: the participant is asked to show and tell the interviewer how to do a typical procedure such as brushing their teeth; (9) Cartoons: a series of cards depicting cartoon characters that tell a story then the participant stands to retell the story and their use of the gestures, emo-

Table 4.1: The information of selected ADOS scenes.

Scene	Average Length(Seconds)	# Frame(In total)	# Frames (Average)
5	398	493,052	11,738
6	381	471,860	11,234
7	473	579,509	13,797
11	409	502,536	11,965
12	515	630,211	15,005
13	79	94,659	2,254
14	67	80,917	1,927

tions, and reference to relationships is evaluated; (10) Break: the participant is given a few items (magazines, toys, color pens, papers) and the interviewer observes their behavior during this free time; (11) Daily Living: questions about their current living situation gauge their level of independence; (12) Friends, Relationships, and Marriage: gauge the participant's understanding of the nature of these types of relationships; (13) Loneliness: the participant's understanding of loneliness is evaluated; (14) Plans and Hopes: what does the participant anticipate in the future for them self; (15) Creating a Story: the participant uses their imagination to create a novel story using some objects. All participants were evaluated in all sections.

The 15 ADOS sections were proceeded one by one in order. In each section, there are multiple standard questions/instructions. Usually, the interviewer poses questions, and each participant gives his or her corresponding responses, such as answering questions verbally, performing some actions such as gesturing with their hands, and so on. At the same time, the interviewer takes notes about the participant's responses in real time from the ADOS evaluation booklet. The time duration of each section depends on the response of the participant. Different scenes are designed for the analysis of different aspects, e.g., facial expressions, body action, and hand movements. For instance, sections 8 and 9 show more body actions, while 11 and 12 have more talking.

Table 4.2: The scoring categories of ADOS observations.

	Category	Items	Description
A	Language and communication	$A1 \sim A10$	speech, gesture, etc.
B	Reciprocal social interaction	$B1 \sim B13$	eye-contact, facial expression, speech, gesture, gaze, etc.
C	Imagination	C1	expressive language skills
D	Stereotyped behaviors and restricted interests	$D1 \sim D5$	hand or figure behaviors, etc.
E	Other abnormal behavior	$E1 \sim E3$	over activity, anxiety, tantrums, etc.

4.1.3 ADOS Interview Videos

ADOS is a semistructured assessment of communication, social interaction, and play (or imaginative use of materials) for individuals suspected of having autism. In this study, all individuals with ASD have been videotaped during their ADOS interviews, and all videos have been scored with consensus by ADOS-reliable clinical psychologists. The scores will serve as the ground truth to train the machine learning algorithm.

In summary, forty-two videos totaling 3,165 minutes have been captured. There are two people (participant and interviewer) in each video. Most of the time, each participant was asked to sit in the chair that faces the camera when talking with the interviewer, except the Cartoon section (#9) requires the participant to stand up to perform body movements to tell the story in the cartoon. Each video ranges in 50 $\sim$ 170 minutes. The average length of the videos is about 75.36 minutes.

For further analysis, the raw video data are first preprocessed. For each video, we mark the starting and end points of each task and split the video into 15 separate subvideos based on the 15 ADOS tasks. To study the feasibility to analyze the videos automatically, we chose carefully the interview sections for tasks 5, 6, 7, 11, 12, 13, and 14, focusing more on the facial dynamics for feature extraction. Since participants were sitting on the chair for most of the time during the interviews, it is proper to capture the dynamic features around their face regions in a short, continuous period. 292 subvideos with 2,852,744 frames were picked up finally with an average length of 334 seconds, as shown in Table 4.1. One can see that scenes 5, 6, 7, 11, 12 take a longer time than 13 and 14.

Table 4.3: ADOS score calculation and classification.

Label	Communication (Comm)	Social Interaction (RSI)	Comm + RSI	Operation
	A4, A8, A9, A10	B1, B2, B6, B8, B9, B11, B12		
Autism	≥ 3	≥ 6	≥ 10	AND
Autism Spectrum	< 3	< 6	< 10	OR
	≥ 2	≥ 4	≥ 7	AND
Non-Spectrum	< 2	< 4	< 7	OR

4.1.4 ADOS Scores and Labeling

The scoring of ADOS-2 Module 4 videos (based on the entire video of all interview questions, not a single observation) by ADOS experts includes the following 5 broad categories with 32 items: (A) Language and Communication, (B) Reciprocal Social Interaction, (C) Imagination/Creativity, (D) Stereotyped Behaviors and Restricted Interests, and (E) Other Abnormal Behaviors. In each scoring section (A-E), there are several detailed questions. Table 4.2 shows a detailed item list of each scoring category. Each item contains a few score levels: 0, 1, 2, 3, 7, and 8. Score $0 \sim 3$ indicates the severity level of the ASD behavior targeted in that question. 0 means the participant's response was at the level one would expect for a person without ASD, while a score of 3 would be highly indicative of ASD. Few questions include a possible score of 7 or 8 and indicate behaviors (e.g., limited by physical disability) that are not contributing meaningfully to the ASD scoring and thus would be scored 0 in the scoring total.

According to the ADOS-2 revised scoring system for the Module 4 algorithm [283], the ADOS scores of all 32 items need to be converted to Module 4 algorithm scores by (1) transferring the assigned ratings of 0,1, and 2 directly to the algorithm form (do not convert), (2) converting the assigned ratings of 3 to algorithm scores of 2, and (3) converting assigned ratings of 7 or 8 to algorithm scores of 0. The Module 4 algorithm adopts the transferred scores of categories A (Communication), B (Reciprocal Social Interaction), and A+B, for autism spectrum diagnostic classification (as shown in Table 4.3). There are three diagnostic categories in total. (1) Autism: all three totals are greater than or equal to the three separate corresponding autism cutoffs (A: 3, B: 6, A+B: 10); (2) Autism Spectrum: all three totals are greater than or equal to the three separate corresponding autism spectrum cutoffs (A: 2, B: 4, A+B: 7), but at least one of them is less than its corresponding autism cutoff (A: 3, B: 6, A+B: 10); (3) Non-Spectrum: any one of the three totals is less than the autism spectrum cutoffs (A: 2, B: 4, A+B: 7). According to ADOS-2 Module 4 classification, we have 17 videos with participants who have the diagnosis of Autism, 10 videos with participants who have the diagnosis of Autism Spectrum, and 15 videos of individuals whose ADOS score resulted in the diagnosis of Non-Spectrum, indicating they do not meet the criteria for a reliable ASD diagnosis.

4.1.5 Characterization of ASD Traits by ADOS Videos

Most existing databases for autism analysis and detection focus on capturing the gaze pattern [118, 130, 137] of individuals when they visually scan images of natural scenes, or they do face processing or facial emotion recognition [139, 140]. Some databases used in body behavior analysis just videotaped certain specific body movements, like grasping a bottle [131] or a simple upper-limb movement [132]. These gaze patterns, simple hand movements, and body behavior, are conventional characteristics present in ASD, which only present the one-fold symptom of autism. Besides, the number of images or video clips in these databases are not big either.

Our database of ADOS interview videos is rich in terms of the variety of behaviors exhibited, including facial dynamics, gaze patterns, eye contact, hand movements, body behavior, speech traits, etc. It is designed specially to capture more complicated and various behaviors in adults with ASD, not only one aspect of the developmental disorder. What is more, each video contains 15 ADOS interview tasks, which also provide abundant information for analysis. Besides, the database is very large and includes forty-two videos of thirty-three recruited individuals, totaling 3,165 minutes of video. After splitting each video into 15 separate subvideos by the time point of each scene, 292 subvideos with

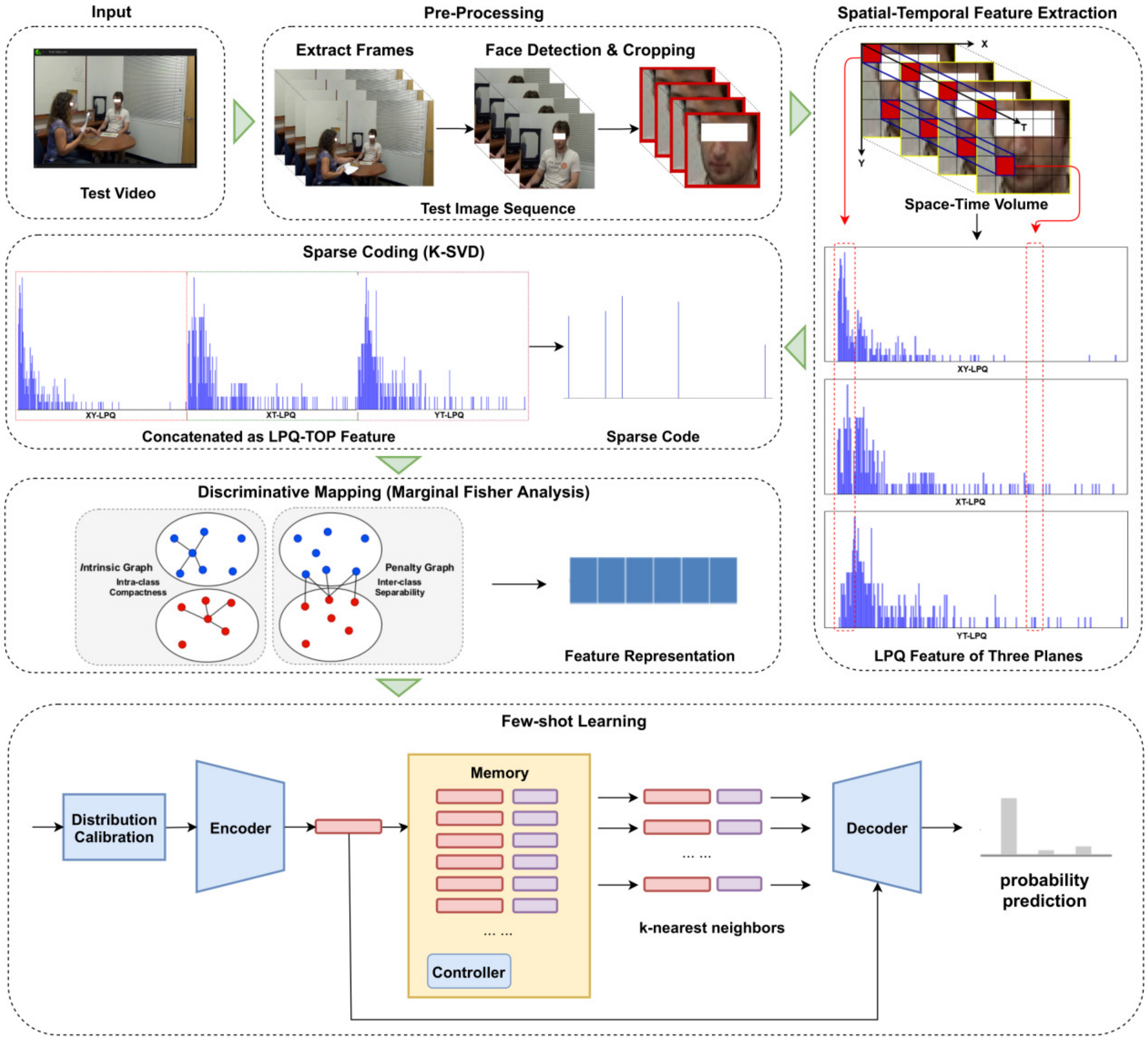


Figure 4.1: The pipeline of the video-based facial dynamics analysis for people with autism spectrum disorder (ASD) trait classification.

2,852,744 frames were obtained with an average length of 334 seconds. Lastly, all videos have already been scored by ADOS-reliable clinical psychologists with consensus, which gives us the ground truth for ASD diagnostic classification.

4.2 Computational Method of ASD Trait Classification

Our method focuses on the analysis of the facial dynamics of ASD people when they participate in the ADOS interviews. Figure 4.1 illustrates an overview of the framework. We first extract key frames containing the subject of interest from ADOS scenes and crop face regions as the preprocessing step. Then we conduct spatio-temporal feature extraction from the cropped video and apply sparse coding to generate discriminative features. Next, feature distribution calibration and adaptive posterior learning are performed for few-shot classification.

4.2.1 3D Spatio-Temporal Facial Feature Extraction

Unlike image-based facial expression analysis [114, 115], both spatial and temporal information in ADOS videos are important to the classification of autism traits. On the one hand, spatial information relevant to ASD is embedded in the form of facial appearance, static expression, and eye movements of participants. On the other hand, temporal information related to ASD is characterized by facial motion across frames, conveying the facial dynamics of the subjects, such as expression and microexpression changes, gaze patterns, and head pose variations. To fully exploit the discriminative information in space and time, it is plausible to consider a method of 3D spatio-temporal feature extraction by encoding both appearance and dynamics from the given input video.

Video Pre-processing. Before extracting dynamic facial features, a face detection method is applied to determine the facial region of the participant in video frames. A face detector Multi-Task Cascaded Convolutional Networks (MTCNN) [284] is adopted here, which is a deep cascaded multitask based face detector. All detected faces are cropped by a square bounding box, and 60 continuous face frames (about two seconds long) are integrated as a 3D volume for feature extraction. In some subvideo clips, the frames that the face detector fails are simply skipped.

Spatio-Temporal Feature Extension. Local Phase Quantization in Three Orthogonal Planes (LPQ-TOP) [285] is a descriptor extended from the purely spatial representation LPQ for spatio-temporal analysis. It is obtained from small space-time video volumes. The histograms from all space-time video volumes are concatenated as a feature vector to represent the corresponding face image sequence. First, the basic LPQ features, denoted as XY-LPQ, XT-LPQ, and YT-LPQ, are independently extracted from three orthogonal planes: XY, XT, and YT, respectively, while considering the co-occurrence statistics in these three directions. The XY plane provides the spatial domain, while the XT and YT planes have the temporal information. Thus, by using this dynamic texture descriptor, both appearance and motion in three directions are considered.

4.2.2 Discriminative Representation Learning

For the task of ASD trait classification, it is important to work with discriminative features instead of original data representations such as LPQ-TOP features. The problem of discriminative representation learning [286] has been extensively studied in the literature of ML and CV. Generally speaking, there are two classes of strategies: dictionary learning via sparse coding (e.g., K-SVD [287]) and dimensionality reduction via factor analysis (e.g., Marginal Fisher Analysis [288]). In this work, we have considered a combination of both methods to generate composite discriminative features for our autism video analysis.

Sparse Coding via K-SVD. After Spatio-Temporal feature extraction, hundreds of LPQ-TOP features are generated from a single subvideo. Sparse coding is used to organize these feature descriptors together, which aims at obtaining a sparse representation. Sparsity means that only a small fraction of the entries are nonzero among all coefficients of base vectors. This kind of representation can be discriminative and concise, as it could select a subset of base vectors which express the concentrated input signal. A popular approach to signal modeling is the synthesis-based sparse representation model, where a signal $\mathbf{x} \in \mathcal{R}^d$ is assumed to be composed as a linear combination of a few atoms from a given dictionary $\mathbf{D} \in \mathcal{R}^{d \times n}$ [289, 290]. The main activity in studying this model concentrated on estimating the representation of a corrupted signal and learning the dictionary **D** from signal examples. K-Singular Value Decomposition (K-SVD) [287] is one of the popular dictionary learning algorithms.

K-SVD is an unsupervised dictionary learning method and focuses on the representational power [291]. Given a set of n-dimensional input signals $\mathbf{Y} = [y_1, ..., y_N] \in \mathcal{R}^{n \times N}$, a dictionary $\mathbf{D} = [d_1, ..., d_K] \in \mathcal{R}^{n\times K}$ can be learned by lowering the reconstruction error via sparse coding as follows:

$$arg\ min_{D,X}\ \|Y - DX\|_2^2\ \ s.t.\ \ \forall i,\ \|x_i\|_0 \leq T, \tag{4.1}$$

where $\mathbf{X} = [x_1, ..., x_n] \in \mathcal{R}^{K\times N}$ consists of the sparse codes of the input signals **Y**, and **T** is a positive integer specifying the sparsity level.

An LPQ-TOP feature can be treated as a sparse linear combination of all dictionary words plus a residual or sparse error. The values of the coefficient of linear combination are generated as the sparse code. Finally, all these sparse codes on the whole subvideo are averaged as a descriptor of the visual-based nonverbal behavior manner for the subject. Each code can be considered as a typical behavior pattern. The average sparse codes provide a better characterization towards a clarified behavior manner.

Dimensionality Reduction via Marginal Fisher Analysis (MFA). For further enhancing the discriminative capability, a supervised dimensionality reduction algorithm called Marginal Fisher Analysis (MFA) [288] is utilized to map the sparse feature into a new space with a better discrimination. In comparison to Linear Discriminant Analysis (LDA), there is no assumption on the data distribution, thus it is more general for discriminant analysis. This method utilizes the graph embedding framework as a tool, designs two graphs that characterize the infraclass compactness and interclass separability, respectively, and optimizes their corresponding criteria based on the graph embedding framework by obtaining the optimal projection vector $\hat{v}$ to satisfy Eq. (4.2):

$$\hat{v} = arg\ min_{\mathbf{v}} \frac{v^T X L_{intra} X^T v}{v^T X L_{inter} X^T v} \tag{4.2}$$

where $\mathbf{X} = [x_1, ..., x_n]$ is input data, $\mathbf{L}_{intra}$ is within-class Laplacian matrix, and $\mathbf{L}_{inter}$ is between-class Laplacian matrix. $\mathbf{L}_{intra}$ is calculated by $\mathbf{D}_{intra} - \mathbf{S}_{intra}$, and $\mathbf{L}_{inter}$ is $\mathbf{D}_{inter} - \mathbf{S}_{inter}$. In them, $\mathbf{S}_{intra}$ is the affinity weight matrix where $s_{ij} = 1$ when x_i and

x_j are **k** nearest neighbors of each other in same class, otherwise $s_{ij} = 0$. $\mathbf{S}_{inter}$ is the opposite. **D** is a diagonal matrix, in which $\mathbf{D}_{i,i} = \sum_j s_{ji}$.

4.2.3 Few-shot Learning

Distribution Calibration (DC). The model is easy to become overfitted if it is trained on the data with a biased distribution containing only a limited number of samples such as ASD population. One effective strategy of combating such few-shot learning scenarios is to calibrate the distribution of these few-sample classes by transferring statistics from the classes with sufficient examples [292]. Inspired by the success of distribution calibration [292], we assume that a few examples can be sampled from the calibrated distribution for expanding the inputs to the classifier. Under the framework of autism trait classification, we further assume every dimension of the extracted facial feature in the previous subsection follows a Gaussian distribution. Therefore, the mean (μ) and the variance (σ) of the distribution of each class in the target data can borrow from that of similar classes (base data) whose statistics are better estimated with a few samples.

Similar to [292], the feature of target data can be transformed by Tukey's Ladder of power transformation [293] as described in Eq. (4.3) to reduce the skewness of the distribution:

$$\tilde{x} = \begin{cases} x^{\lambda} & if\ \lambda \neq 0 \\ log(x) & if\ \lambda = 0 \end{cases} \tag{4.3}$$

where λ is a hyperparameter to adjust the distribution. For each class, Eq. 4.4 is used to calibrate the mean $\hat{\mu}$ and the covariance $\hat{\sigma}$ for each class using $\tilde{x}$, and then the generated features are achieved from the calibrated distribution.

$$\hat{\mu} = \frac{\mu + \tilde{x}}{2}, \hat{\sigma} = \sigma + \alpha \tag{4.4}$$

where α is a hyperparameter determining the degree of dispersion of spatio-temporal features after discriminative mapping.

Adaptive Posterior Learning (APL). Next, the calibrated features are fed to an adaptive posterior learning (APL) [100] model to perform few-shot learning. The key idea behind APL is to approximate probability distributions by remembering the most surprising observations it has encountered. In the situation of ASD trait classification, the past observations can be recalled from an external memory module and processed by a decoder network. The objective of FSL is achieved by combining the information from different memory slots to generalize beyond direct recall. More specifically, our APL implementation consists of three parts: Encoder, Decoder and Memory.

The Encoder encodes the input. It is implemented by a convolutional network, which is composed of a single first convolution and 15 convolutional blocks. Each block has a Batch Normalization layer, a ReLU activation, and a convolutional layer with the kernel size of 3. For every three blocks, the convolution contains a stride of two to downsample the feature. Finally, the feature is flattened to a 1D vector and passed through a Layer Normalization function.

The memory stores the codes that the encoder has seen as key-value format. Key is the encoded embedding, and value is the true label. A controller is designed to decide which embedding can be written, at the same time, trying to minimize the amount of written embedding. A quantity metric surprise is defined to indicate the probability model assigns the input to the true class correctly. The higher the probability is, the less surprised it will be. If an embedding is surprising, it should be written in memory; otherwise it should be discarded. When querying, the memory is scanned for the top k nearest neighbors of the embedding of input. The concatenation of query embedding, recalled neighbor embeddings with memory labels and distances is fed to the decoder.

The Decoder predicts the probability distribution over targets. It is implemented by a relational feed-forward module with self-attention. It compares each neighbor individually with the query using a cross-element comparison with a self-attention module, and then reduces the activations with an attention vector calculated from neighbor distances. The self-attentional blocks repeat five times in a residual manner. The resulting tensors are called activation tensors. Besides, the distances between neighbors and query are passed through a softmax layer to generate an attention vector, which is summed with the activation tensor over the first axis to obtain the final logit result for classification.

4.3 Experimental Results of Trait Classification

The experiment is conducted on the ADOS videos that we introduced in Section 4.1. In this subsection, we first describe the implementation settings and procedure in detail. Then, we show the performance of individual scenes and the fusion of selected scenes. We have also compared this work with several recently proposed image-based classification methods to demonstrate the superiority of the proposed approach. Significant improvements of ASD trait classification accuracy have been achieved by the proposed feature-level and scene-level fusion strategies. Finally, the results of our ablation study are reported; limitations of the proposed method and discussions about future research directions are presented.

4.3.1 Experimental Setup

Preprocessing. To extract a facial representation from the videos, the first step is to apply face detection and cropping to get the region of interest for each video frame. In our experiments, we chose MTCNN [284] as the face detector to crop the participants' faces. All cropped faces are resized to grayscale images with a size of 100 $\times$ 100. Finally, about 98.9% of frames containing faces are successfully detected.

Facial Feature Extraction. In our experiment, we adopt a three-dimensional face region subvolume as the basis for feature extraction. Two-second long face frames are considered as the basic unit. Frames that faces are failed to be detected are removed directly. After 3D spatio-temporal feature extraction, a 768 dimensional feature vector is obtained for each

3D subvolume ($100 \times 100 \times 60$), in which the first 256 features are extracted from XY plane, the second 256 from XT plane, and the third 256 from TY plane. To learn the sparse coding dictionary, only LPQ-TOP features from the training data are employed. We borrow the settings from [39] in which the dictionary with sparse level 3 and word size 250 for K-SVD, and, for MFA, the number of nearest neighbors k is 4 for within-class definitions and 2000 for between-class definition.

Few-shot Learning. For calibration distribution, the extracted features of all selected scenes are treated as base data to estimate the mean and variance of each class for each scene. The mean of the feature vector is calculated as the mean of every single dimension in the vector. Covariance is used for a better representation of the variance between any pair of elements in the feature vector. Here, λ is 0.5, and α is 0.21. The calibrated features are then fed into APL module to predict probabilities. The APL module applies a squared l_2 distance to compute the distance between queries and embeddings stored in memory, and the top three nearest neighbors are returned from the memory. The threshold of the surprise measure metric is set to 0.75. We train 20000 episodes using a Cross Entropy loss, and save the model per 100 episodes. The model with the highest accuracy is selected to evaluate the performance on the test set.

Performance Evaluation. In our experiment, we perform a three-class classification. Due to the small amount of subjects in the database, we adopt a ten-fold cross-validation for each experiment. All videos were partitioned into ten subsets (2 subsets contain 5 videos each, and the other eight contain 4 videos each). One subset is retained as the validation data for testing the model, and the remaining nine subsets are used for training. The cross-validation process is then repeated ten times, with each of the ten subsets used exactly once as the validation data. The ten accuracy values are averaged to produce the final accuracy. We quantitatively measure the performance of the method in terms of accuracy of each scene and the fusion of multiple scenes. For fusion strategy, feature concatenation is applied to selected scenes and then fused feature is fed into few-shot learning module for

Table 4.4: Accuracy (%) of our method (LPQ-TOP+K-SVD+MFA+APL+DC), LPQ-TOP, LPQ-TOP+K-SVD, LPQ-TOP+K-SVD+MFA, and LPQ-TOP+K-SVD +MFA+APL, on individual scenes.

Scene No.	5	6	7	11	12	13	14
LPQ-TOP	61.40	61.35	52.29	55.11	51.29	47.98	44.72
LPQ-TOP+K-SVD	65.35	64.21	61.26	55.61	53.81	56.35	61.41
LPQ-TOP+K-SVD+MFA	81.45	81.51	73.49	69.58	72.09	76.50	72.00
LPQ-TOP+K-SVD+MFA+APL	88.67	87.35	79.17	79.17	83.33	87.25	83.33
LPQ-TOP+K-SVD+MFA+APL+DC (ours)	**89.64**	**89.55**	**86.83**	**87.12**	**88.33**	**88.67**	**88.49**

Table 4.5: Performance (%) of our method on feature level fusion (feature concatenation).

Features	Scenes	Accuracy(%)
TOP 3	5,6,13	90.00
TOP 5	5,6,12,13,14	91.67
TOP 7	5,6,7,11,12,13,14	**91.72**

classification.

4.3.2 Performance of Individual Scenes

Seven selected scenes (5, 6, 7, 11, 12, 13, 14) are experimented with ten-fold cross-validation, respectively. To facilitate the visual inspection, we have shown the fusion results in both Table 4.4 and Fig. 4.2. One can clearly observe the improvement of accuracy as our computational model becomes more sophisticated (significant gain is achieved by adding MFA and APL into the model).

From the result shown in Table 4.4 (last row), we can see scene 5 and 6 gained pretty high accuracies (89.64% and 89.55%), the following scenes 11, 12, 13 also obtained high accuracies (above 88%), and scenes 7, 11 had the lowest accuracies (86.83% and 87.12%). It can be noted that the lengths of different scenes vary, so does their discriminative power for ASD trait classification. It seems that talking topics that may cause mental or emotional stress on participants could reveal more explicit face behaviors showing autism spectrum disorder. For example, as shown in Fig. 4.2, scene 5 discusses work or school things to evaluate their realistic understanding of the possibilities for future employment, training,

Table 4.6: Accuracy (%) comparison with other competing methods.

Methods	Scene 5	Scene 6	Scene 7	Scene 11	Scene 12	Scene 13	Scene 14	Overall
Beary et al. [143]	66.07	64.29	55.36	62.5	62.5	67.86	66.07	68.46
Akter et al. [144]	73.57	61.07	64.64	64.64	57.5	68.21	70.57	75.9
Lu& Perkowski [145]	75.43	70.71	57.86	56.07	60.86	66.79	54.29	78.96
Ours	89.64	89.55	86.83	87.12	88.33	88.67	88.49	**91.72**

or experience necessary for future employment, and scene 6 talks about social difficulties and annoyance, which contains problems or troubles getting along with other people, like irritation, tease, or bullying.

4.3.3 Performance of Scene-level Feature Fusion

We have also compared the classification performance by fusing several scenes together at the feature level (i.e., via feature concatenation). The fusion experiments are conducted for the top three (5, 6, 13), top 5 (5, 6, 12-14), and top 7 (5-7, 11-14), respectively. The experimental results are shown in Table 4.5. From the table, one can see that top-seven fusion achieves the best performance with an accuracy of **91.72%** by fusing all selected seven scenes that are comparable with the standardized diagnostic scales, with the advantages of efficiency and objectiveness. By contrast, scene-level fusion achieves 91.67% and 90.00% for the top 5 and top 3 settings. The comparison result shows the effectiveness of combining multiple scenes to improve the performance.

To the best of our knowledge, there are little video-based facial analysis methods of autism published in the open literature. For autism research, only a few video-based methods with special constrains on the data exist (e.g., [151]); others have only done some preliminary analysis work on extracted frames [153]. Taking [151] for example. It uses face-based attention recognition models to detect and classify children with ASD on videos which record the facial behaviors of the participants when they are taking the attention tasks. However, the class labels should be annotated as with attention and without attention according to the attention behavioral rules. For comparison, three image-based methods on facial analysis of autism are chosen in our experiment. Since the codes are not released publicly, we try our best to reimplement the mentioned methods according to the description of the papers. Most methods are designed using transfer learning based on pretrained deep models, such as VGG16, MobileNet, etc. We get the output of the last layer before the final classification layer of the model as the feature of the input frame. To get proper video feature, for each subvideo, we extract the facial feature of one frame for each 60 continuous face frames and adopt K-SVD on these features to generate sparse codes. All codes are averaged as the final descriptor. SVM is used for three-class classification. Our method performs better than others as shown in table 4.6.

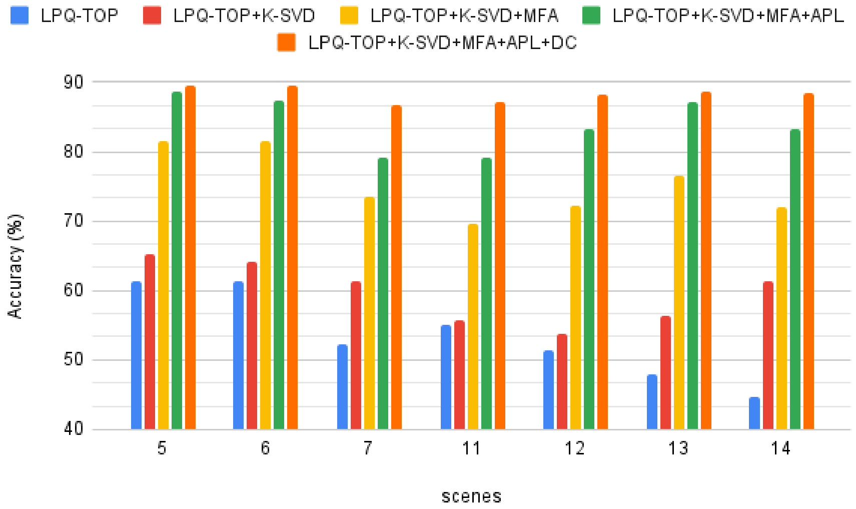


Figure 4.2: Histogram of accuracy (%) of LPQ-TOP+K-SVD+MFA+APL+DC(our method), LPQ-TOP, LPQ-TOP+K-SVD, LPQ-TOP+K-SVD+MFA, and LPQ-TOP+K-SVD+MFA+APL. **Best view in color.**

4.3.4 Ablation Study

During feature extraction, we had several strategies to derive more discriminative features, e.g., K-SVD, MFA. For better classification, DC and APL are designed to further improve the performance. Here, we validate the contribution of each component by conducting an ablation study. The results of our ablation study can be seen in Fig. 4.2.

In the experiment of LPQ-TOP component, the LPQ-TOP feature of each 3D space-time volume is extracted directly as one training/test sample for classification. In the remaining experiments, the feature of each subvideo is treated as one data sample. From the result, one can see that most accuracies are lower than that of the complete method. Figure 4.2 gives a visual illustration of the contribution from different components. All components are useful for the classification algorithm to improve the recognition accuracy, especially the addition of MFA and APL. Since there is no assumption on the data distribution, MFA is more general for discriminant analysis, which explains its significant role in improving the accuracy. The inter-class margin can better characterize the separability of different classes. The deep learning based APL module can better predict probability distributions of input by recalling past observations and combining information from different memory slots to generalize beyond direct recall.

4.4 Methodology of Facial Micro-Expression Analysis

In this task, we focus on the analysis of the facial micro-expression of ASD people in the ADOS interviews. Figure 4.3 illustrates the pipeline of the framework. It contains 4 steps: pre-processing, micro-expression spotting, feature extraction and classification. First, the face frames are extracted from the video by detection and cropping. Second, we use a spotting model to locate the onset, apex, and offset of each micro-expression movements. And then, discriminative feature from each spotted segment are learnt. The final representation is fed to SVM for classification.

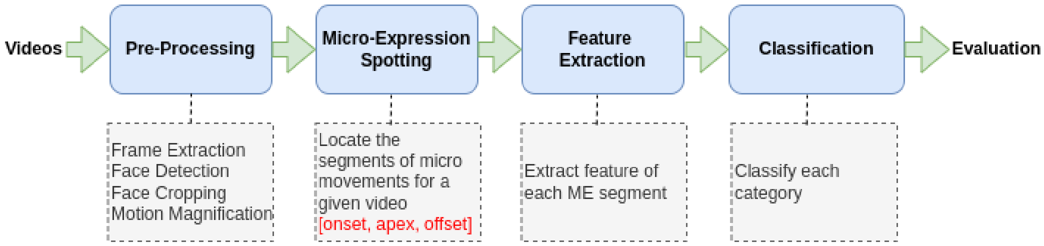


Figure 4.3: Basic framework of facial micro-expression analysis in ADOS videos.

4.4.1 Pre-processing

First, we split the whole video into 15 separate subvideos based on the 15 tasks, and choose scene 5-7 and 11-14 for face analysis. For each scene, a face detection method is applied to determine the facial region of the participant in video frames. A face detector Multi-Task Cascaded Convolutional Networks (MTCNN) [284] is adopted here, which is a deep cascaded multitask based face detector. All detected faces are cropped by a square bounding box.

Facial micro-expressions vary from the visible to the subtle. Before analysis, motion magnification is considered to enhance the intensity level of facial expressions, making subtle facial muscle movements more recognizable and distinguishable. We choose Eulerian Video Magnification (EVM) [294] to magnify tiny muscle movements and colors. The basic idea is to apply spatial decomposition, followed by temporal filtering to the input frames. The method first decomposes the input video sequences into different spatial frequency bands, and applies the same temporal filter to all bands. The filtered spatial bands are then amplified by a given factor α, added back to the original signal, and collapsed to generate the output video.

4.4.2 Optical Flow Feature

Optical flow feature has shown the usefulness of spatio-temporal motion information in micro-expression analysis [205, 295, 296]. In this task, we extract three different optical flow features (horizontal component, vertical component, and optical strain) of frames as the input data of the spotting model.

The optical flow features are computed between two frames, i.e. current frame F_i and the k-th frame after F_i, F_{i+k}, where k is half of the average length of an expression movement. For horizontal (u) and vertical (v) components, TV-L1 [297] is used to calculate the optical flow feature. Optical strain (ε) [205] is a derivative of optical flow. It captures subtle motion changes based on the elastic deformation of facial skin tissue (facial deformation).

Let I(x, y, t) be a video sequence, optical strain ε is defined as follows:

$$\varepsilon = \begin{bmatrix} \frac{\delta u}{\delta x} & \frac{1}{2}(\frac{\delta u}{\delta y} + \frac{\delta v}{\delta x}) \\ \frac{1}{2}(\frac{\delta v}{\delta x} + \frac{\delta u}{\delta y}) & \frac{\delta v}{\delta y} \end{bmatrix} \tag{4.5}$$

these three components (u, v, and ε) represent the input data of the spotting model.

4.4.3 Micro-Expression Spotting

Micro-expression spotting is to find the time intervals at which micro-expressions are detected [298]. The time interval is composed of onset, apex and offset frame. Onset frame is the moment when the facial muscle movements begin to increase, as the green box shown in Fig. 4.4. Apex frame (red box) is when a facial expression develops to its most obvious moment. Offset frame (blue box) is when the facial muscles return to a neutral appearance. What we want is the segment of frames from onset to offset.

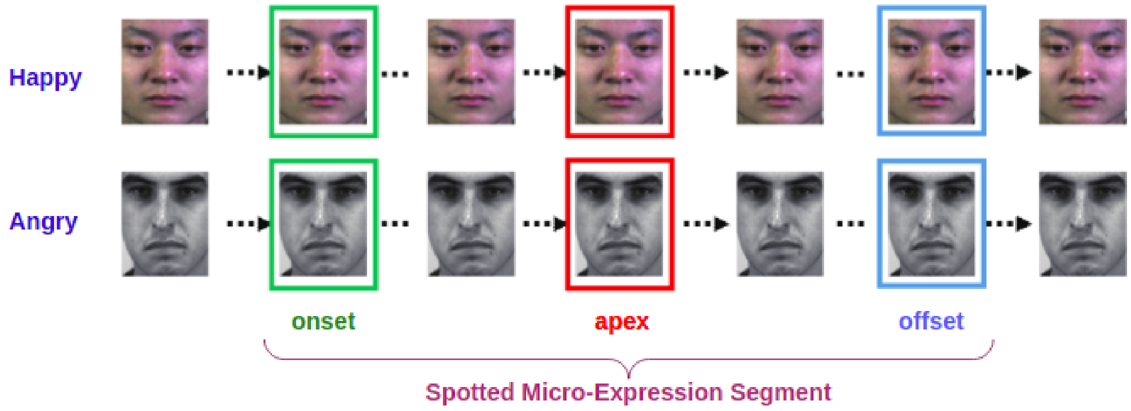


Figure 4.4: Onset (green box), apex (red box) and offset (blue box) frames of facial micro-expression segments.

Spotting task is a prerequisite for advanced facial micro-expression analysis. Proper spotting can decrease the redundant information of the data. We choose a shallow three-stream CNN model (SOFTNet) [299] as the basic model. It treats the spotting task as a regression problem that predicts how likely a frame belongs to a micro-expression. It takes three different optical flow features of frames as input. As shown in Fig. 4.5, three types of optical flow features (u, v, ε) of the i-th frame are fed into three streams, respectively. Each stream consists of a single convolutional layer with 3, 5, and 8 filters respectively, followed by a max-pooling layer to reduce the feature map size. The features from three streams are then concatenated by channel-wise fusion, followed another max-pooling layer. The output is flattened by a 400-node layer, and fully connected to a single output score via linear activation. Finally, the model predicts a spotting confidence score $\hat{s}_i$ of frame i.

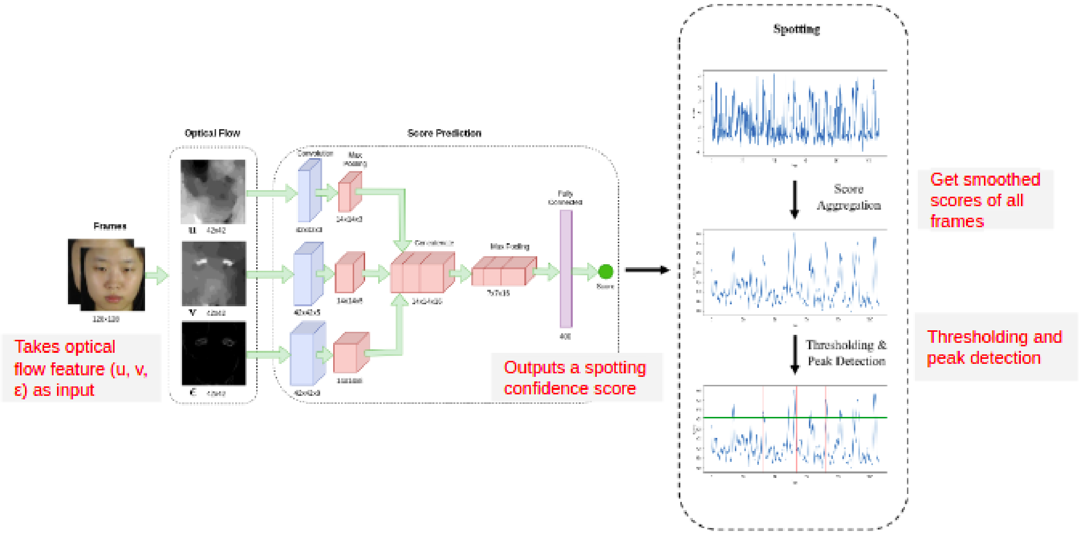


Figure 4.5: Framework of facial micro-expression spotting model.

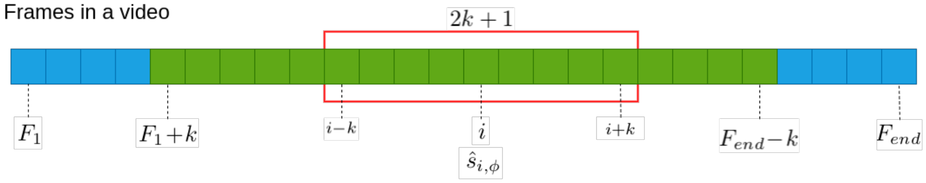


Figure 4.6: Score aggregation using a sliding window approach.

After obtain the confidence scores of all frames in a video, we use a sliding window approach to smooth the scores. The predicted scores of k frames before the current i-th frame, and k frames after the i-th frame, are averaged as the final score of frame i. As shown in Fig. 4.6, the aggregated score of i-th frame is calculated by:

$$\hat{s}_{i,\phi} = \frac{1}{2k+1} \sum_{j=i-k}^{i+k} \hat{s}_{j,\phi}, fori = F_1 + k, ..., F_{end} - k. \tag{4.6}$$

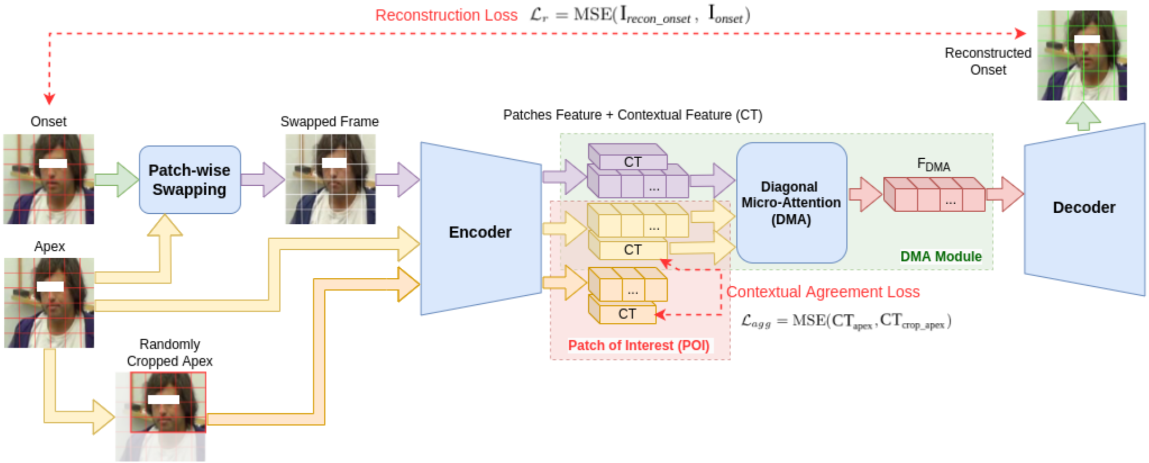
where k is half the average length of micro-expression, ϕ means all frames in the video.

Different input subvideos have different thresholds. Finally, we compute the threshold of the input video and use this score to spot the peaks in the video. The threshold T is computed by:

$$T = \hat{S}_{avg} + p \times (\hat{S}_{max} - \hat{S}_{avg}), \tag{4.7}$$

where $\hat{S}_{avg}$ and $\hat{S}_{max}$ are the average and maximum predicted scores over the given video, and p is a tuning parameter between 0 and 1. If the frame s is a local maxima and the minimum distance between two continuous local maxima is greater than k, this frame can be chosen as a peak frame. The final micro-expression interval $\hat{E} = [s - k, s + k]$ is composed of $2k + 1$ frames by extending k frames forward and backward.

4.4.4 Micro-Expression Feature Extraction



Reconstruction Loss $\mathcal{L}_r = \text{MSE}(\text{I}_{recon\_onset}, \text{I}_{onset})$
Reconstructed Onset
Onset
Patch-wise Swapping
Swapped Frame
Patches Feature + Contextual Feature (CT)
CT
...
Encoder
Diagonal Micro-Attention (DMA)
F_{DMA}
DMA Module
Decoder
Apex
Contextual Agreement Loss
$\mathcal{L}_{agg} = \text{MSE}(\text{CT}_{\text{apex}}, \text{CT}_{\text{crop\_apex}})$
Patch of Interest (POI)
Randomly Cropped Apex

Figure 4.7: BERT-based Auto Encoder: the framework for facial micro-expression feature extraction.

After we have obtained all the spotted intervals of ADOS videos, next step is to extract discriminative feature of each interval for autism diagnosis. The goal is to recognize tiny facial movements difficult for humans to perceive. We use a BERT-based Auto Encoder model (Micron-BERT) [300] as shown in Fig. 4.7 to automatically capture these micro-changes of facial texture across temporal dimensions in an unsupervised manner.

Micron-BERT consists of five main modules: Patch-wise Swapping, Encoder, Diagonal Micro-Attention (DMA), Patch of Interest (POI) and Decoder. Two input frames I_{onset} and I_{apex}, are first divided into a set of several non-overlapping patches, respectively. Patch-wise Swapping module randomly swaps two corresponding patches between I_{onset} and I_{apex} to create a swapped image $I_{swapped}$. Each patch in $I_{swapped}$ and I_{apex} is linearly projected into a latent vector by Encoder, with additive fixed positional encoding [9]. And then, the Diagonal Micro-Attention (DMA) module adopt a new attention mechanism to enforce the network automatically focusing on swapped patches and equip it with the ability to precisely spot and identify all changes between I_{onset} and I_{apex}. Besides, a cropped version of I_{apex} is obtained. Before feed image patches into Encoder, a Contextual Token is added to the beginning of the sequence of patches to learn the contextual information in the image. Patch of Interest (POI) module uses a contextual agreement loss by comparing the contextual agreement between the frame I_{apex} and Crop(I_{apex}), to enables the model to automatically explore the salient regions and ignore the background patches in an image. After DMA and POI process, the latent vector of $I_{swapped}$ is finally input to Decoder to reconstruct I_{onset} using construction loss function.

Patch-wise Swapping. Given two frames I_{onset} and I_{apex}, each frame $I \in R^{H \times W \times C}$ is divided into a set of non-overlapping patches with same size:

$$P = \{p_i\}_{i=0}^{n-1}, \tag{4.8}$$

where H, W, C are the height, width, and number of channels, respectively, and n is the number of patches. Each patch p_i has a resolution of $s \times s$.

We firstly randomly select a part of patches from some location of frame I_{onset} and

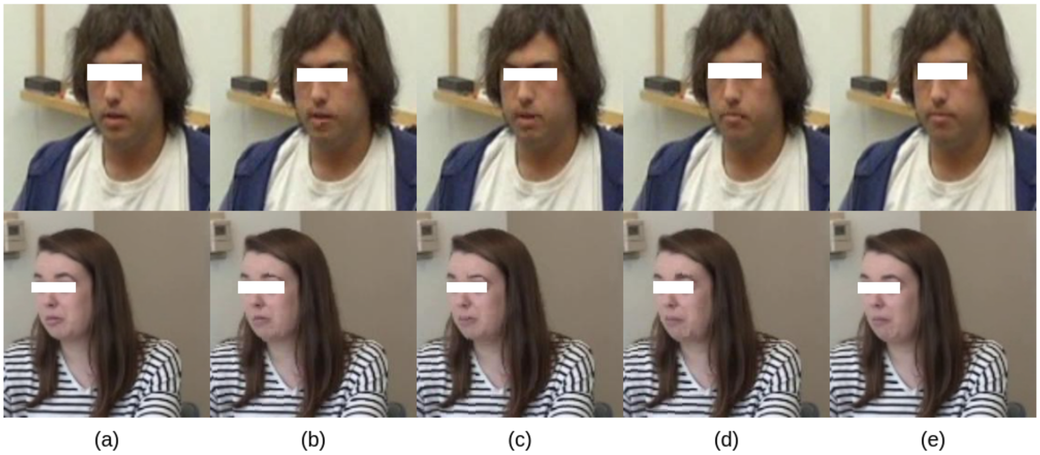


Figure 4.8: Samples of Patch-wise Swapping result with different swapping ratio: (a) 0, (b) 0.3, (c) 0.5, (d) 0.7, (e) 1.0.

patches of the remaining location are from corresponding location of frame I_{apex}. The collected patches are used to create the swapped image $I_{swapped}$. By doing so, the model can be forced to spot these changes and reconstruct I_{onset} from $I_{swapped}$. Moreover, if we replace I_{apex} with a frame that is closer to I_{onset} in the sequence of a micro-expression segment, the robustness on spotting these differences of the model can be further enhanced. Fig. 4.8 shows some samples of swapping result between onset and apex frames (image size: 224×224, patch size: 8×8), here, swapping ratio means the rate of patches from apex frame.

Encoder. The Encoder is to map input frame I into latent space. Each patch $p_i \in P$ as a basic unit is linearly projected into a latent vector $z_i \in R^{1 \times d}$, where d is dimension. The latent vector of I can be represented as:

$$Z = concat[\, z_0, z_1, ..., z_{n-1}] \in R^{n \times d}. \tag{4.9}$$

Encoder is composed of a stack of continuous blocks. Fig. 4.9 illustrates the architecture of one building blocks. It consists of one Multi-Head Attention (MHA) layer and one Multi-Layer Perceptron (MLP). One Layer Normalization (LN) is employed to the input signal before feeding to MHA and MLP layers. For example, for each patch p_i, the

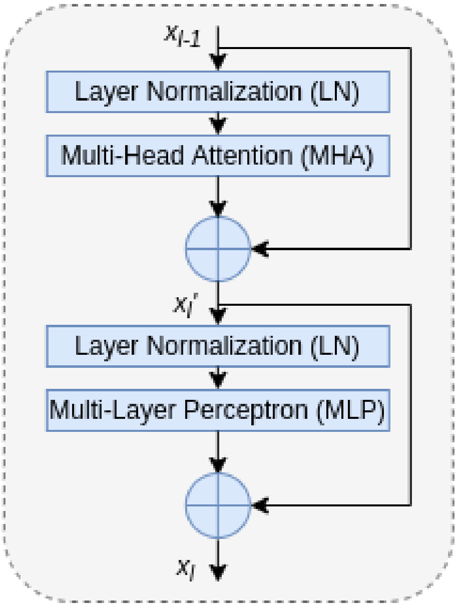


Figure 4.9: Building block of Encoder and Decoder.

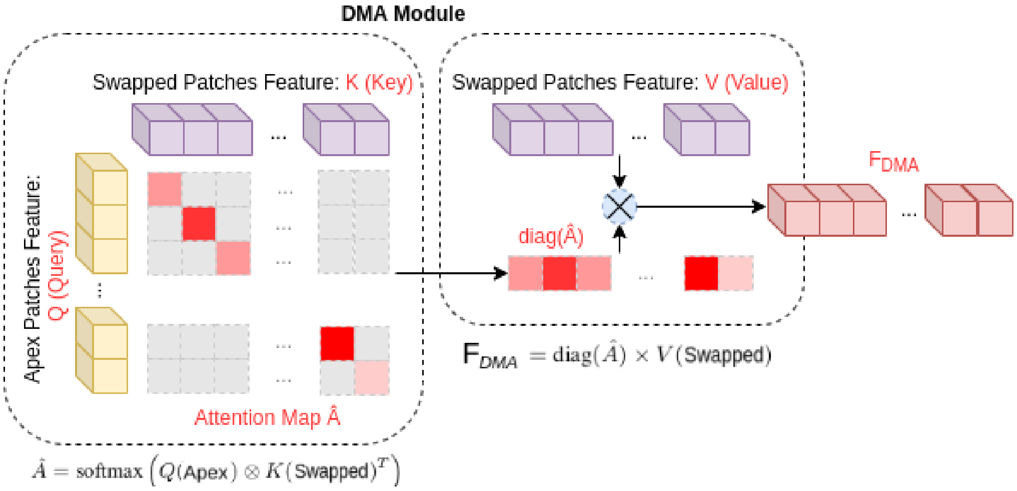
intermediate latent code x_l is represented as:

$$\begin{aligned} x_l^{'} &= x_{l-1} + MHA(LN(x_{l-1})), \\ x_l &= x_l^{'} + MHA(LN(x_l^{'})), \\ x_0 &= p_i \end{aligned} \tag{4.10}$$

Diagonal Micro-Attention (DMA). After Patch-wise Swapping, the image patches $P_{swapped}$ from $I_{swapped}$ contains two types of patches, p_{onset} from I_{onset} and p_{apex} from I_{apex}. The goal of DMA is to detect tiny differences between two frames. The details of DMA are illustrated in Fig. 4.10. First, an attention map $\hat{A}$ between $P_{swapped}$ and P_{apex} is constructed by:

$$\hat{A} = softmax(Q(apex) \otimes K(swapped)^T), \tag{4.11}$$

where the values in diagonal line (diag($\hat{A}$)) indicate correlations between two corresponding patches of $p_{swapped}$ and p_{apex}. diag(Â) can be applied as weights indicating importance



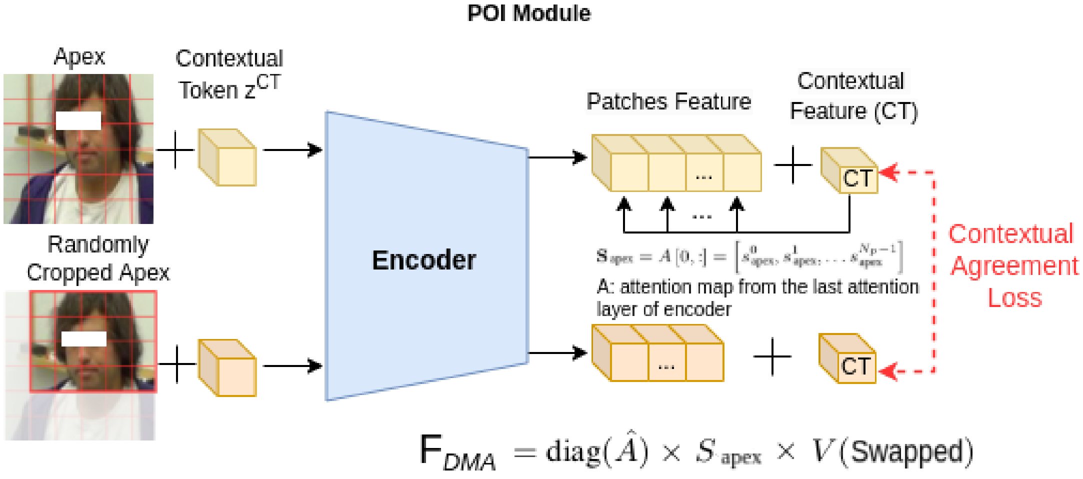
DMA Module
Swapped Patches Feature: K (Key)
Swapped Patches Feature: V (Value)
Apex Patches Feature: Q (Query)
Attention Map Â
diag(Â)
F_DMA
$F_{DMA} = \mathrm{diag}(\hat{A}) \times V(\text{Swapped})$
$\hat{A} = \mathrm{softmax}\left(Q(\text{Apex}) \otimes K(\text{Swapped})^T\right)$

Figure 4.10: Diagonal Micro-Attention (DMA) Scheme.

of each patch features. The final feature after DMA operation can be represented as:

$$F_{DMA} = diag(\hat{A}) \times V(swapped). \tag{4.12}$$

By applying Patch-wise Swapping and DMA, the model can better focus on facial regions that primarily consist of small facial movements between frames.



POI Module
Apex
Contextual Token z^{CT}
Randomly Cropped Apex
Encoder
Patches Feature
Contextual Feature (CT)
CT
CT
$S_{apex} = A[0,:] = \left[s^0_{apex}, s^1_{apex}, \cdots s^{N_p-1}_{apex}\right]$
A: attention map from the last attention layer of encoder
Contextual Agreement Loss
$F_{DMA} = \mathrm{diag}(\hat{A}) \times S_{apex} \times V(\text{Swapped})$

Figure 4.11: Patch of Interest (POI) Scheme.

Patch of Interest (POI). The ideal situation for Patch-wise Swapping and DMA is that all swapped patches come from locations within facial region. In this case, the model can preciously focus on facial micro-movements, and not be distracted by unrelated changes like background. So, a module named Patch of Interest (POI) is adopted to constrain the model to localize and highlight facial micro-expression interest regions rather than unrelated regions such as the background.

Since self-attention [9] allows each patch token to represent contextual information within the group to which it belongs, rather than representing an individual meaning like CNN, the transformer has a property of adaptive weight aggregation. POI module relies on the contextual agreement between frame I_{apex} and Crop(I_{apex}). Before fed into the Encoder, a Contextual Token z^{CT} is added to the beginning of the sequence of patches. z^{CT} can learn the contextual information in the image as it goes through the transformer blocks. The deeper z^{CT} passes through the blocks, the more information is accumulated from z_i. In essence, z^{CT} can be treated as a place-holder to store the contextual information of the image that are extracted from other patches in the sequence.

The detail is shown in Fig. 4.11. Given two input frames I_{apex} and Crop(I_{apex}) and added Contextual Tokens z^{CT}_{apex} and $z^{CT}_{crop\_apex}$, latent vectors are obtained by Encoder. The latent vector is composed of patch feature and contextual feature. We only focus on the contextual features. Contextual Agreement Loss is defined to enforce the similarity of contextual features of I_{apex} and Crop(I_{apex}) so that the model can discover the salient patches:

$$L_{agg} = MSE(CT_{apex}, CT_{crop\_apex}). \tag{4.13}$$

The POI in contextual feature (CT) of I can be extracted from the attention map A in the last attention layer of Encoder:

$$S = A[\,0,:] \; = [\,s_0, s_1, ..., s_{n-1}]\,, \tag{4.14}$$

where $\sum_{i=0}^{n-1} s_i = 1$. A high score of s_i means the corresponding patch contains richer contextual information.

The POI of frame I_{apex} in contextual feature S_{apex} can contribute to the final feature F_{DMA}:

$$F_{DMA} = diag(\hat{A}) \times S_{apex} \times V(swapped). \tag{4.15}$$

and this feature as final latent vector is fed into Decoder for reconstruction.

Decoder. Decoder and Encoder is designed symmetrically. Decoder has similar architecture to Encoder. Given a latent code F_{DMA}, the Decoder tries to reconstructed a frame image I_{recon}. The Reconstruction Loss is defined by:

$$L_r = MSE(I_{recon\_onset}, I_{onset}). \tag{4.16}$$

Loss Function. The final loss function of the BERT-based Auto Encoder is represented as:

$$L = \alpha \times L_r + \beta \times L_{agg}, \tag{4.17}$$

where α and β are the weights for reconstruction loss and contextual agreement loss.

By adopting a series of strategies like Patch-wise swapping, DMA and POI, the model can adaptively focus on only meaningful facial regions to detect the tiny moment changes and ignore the ones in the background.

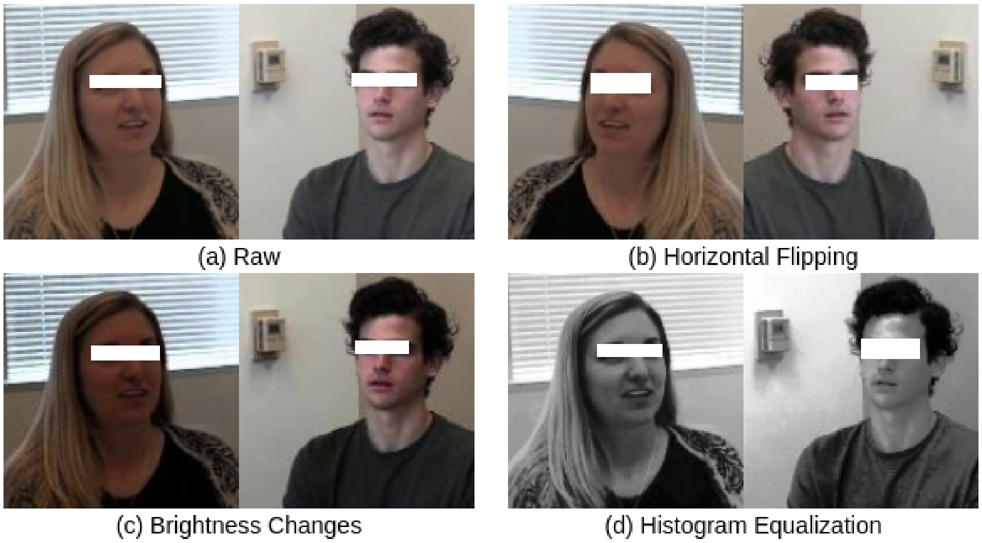
4.5 Experiments of Micro-Expression Analysis

The experiment is conducted on the ADOS videos that we introduced in Section 4.1 and later collected ADOS data from control group by WVU. In this subsection, we first describe the implementation settings and procedure in detail. Then, we will show the performance of individual scenes and the fusion of selected scenes. We will also compare this work with several recently proposed image-based classification methods to demonstrate the superiority of the proposed approach.

4.5.1 Experimental Setup

Data and Augmentation. To normalize the face resolution, the facial region in each frame is cropped and resized to 128×128 pixels. Cropping was performed after the square bounding box was detected from the first (reference) frame of each raw video. We choose a loose bounding box to make sure the whole face in each frame is included.

The original ADOS data contains 42 videos in ASD group and 9 videos later collected from control group. We make an image-based data augmentation on cropped face frames as shown in Fig. 4.12 to enlarge the scale of control group by horizontal flipping, brightness changing, and histogram equalization.


(a) Raw

(b) Horizontal Flipping

(c) Brightness Changes

(d) Histogram Equalization

Figure 4.12: Data augmentation results of (a) raw data by (b) horizontal flipping, (c) brightness changing, and (d) histogram equalization.

Spotting Model Setup. We use a pre-trained SOFTNet model to spot micro-expression intervals. The model was trained on micro-expressions data of $CAS(ME)^2$ [301] which contains 98 long videos consisting of 300 macro-expressions and 57 micro-expressions captured from 22 subjects. Both macro-expressions and micro-expressions were fully annotated with onset, apex, and offset by professional coders. The images are resized to 128×128. The half average length of the micro-expression k is 17 calculated based on $CAS(ME)^2$. The value of p in the Eq. 4.7 of calculating threshold is set to 0.55.

Micron-BERT Model Setup. We use a well-trained Micron-BERT Model to extract micro-expression features. The training strategy is that: (1) it is first trained on a large-scale unlabeled data for self-training, in which all raw frames are from $CAS(ME)^3$ [302] which provides 1,109 labeled micro-expressions and 3,490 labeled macro-expressions. The images are resized to 224×224 with 3 channels, and each is divided into n = 784 patches with same resolution of 8×8. Each patch is projected to a latent space with d = 512. Both Encoder and Decoder have 4 building blocks; (2) and then we take the pre-trained Encoder and DMA modules as the backbone to fine-tune the model for micro-expression recognition task on CASME II [303] which contains 247 micro-expression samples from 26 subjects of the same ethnicity. The input is onset and apex frames for each micro-expression movement; (3) finally, the well fine-tuned model is used as a feature extractor to obtain F_{DMA} feature defined in Eq. 4.12 of spotted micro-expression segments on ADOS data for binary classification.

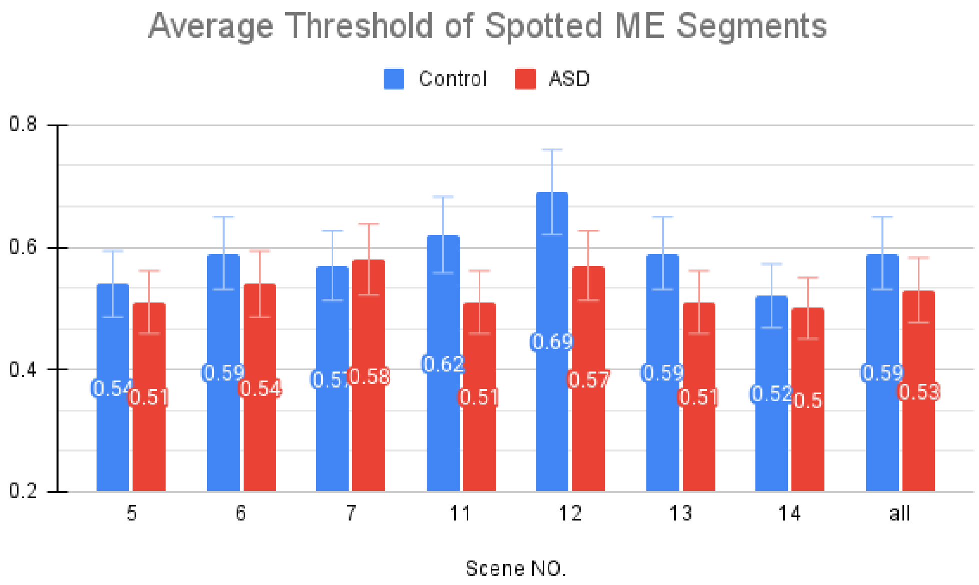


Figure 4.13: Average threshold of spotted micro-expression movements on each scene.

Measure Metrics. In this experiment, we formulate the autism diagnosis as a binary classification problem with ASD and control categories. We take similar evaluation procedure with previous trait classification work, 10-fold cross validation. The performance is measured by accuracy and F1 score. We quantitatively measure the performance of the method on each scene and the combination of multiple scenes.

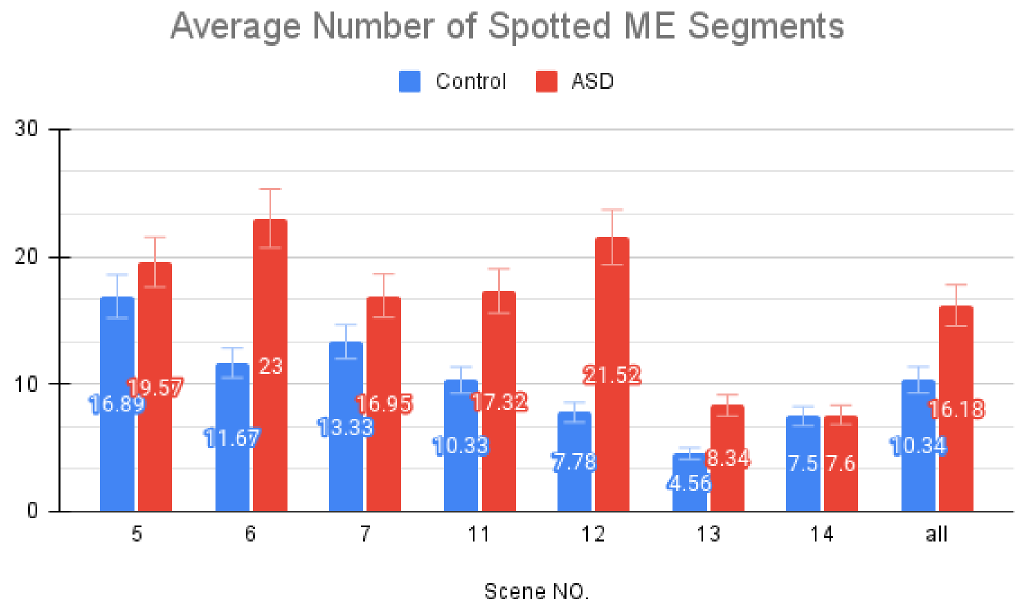


Figure 4.14: Average number of spotted micro-expression movements on each scene.

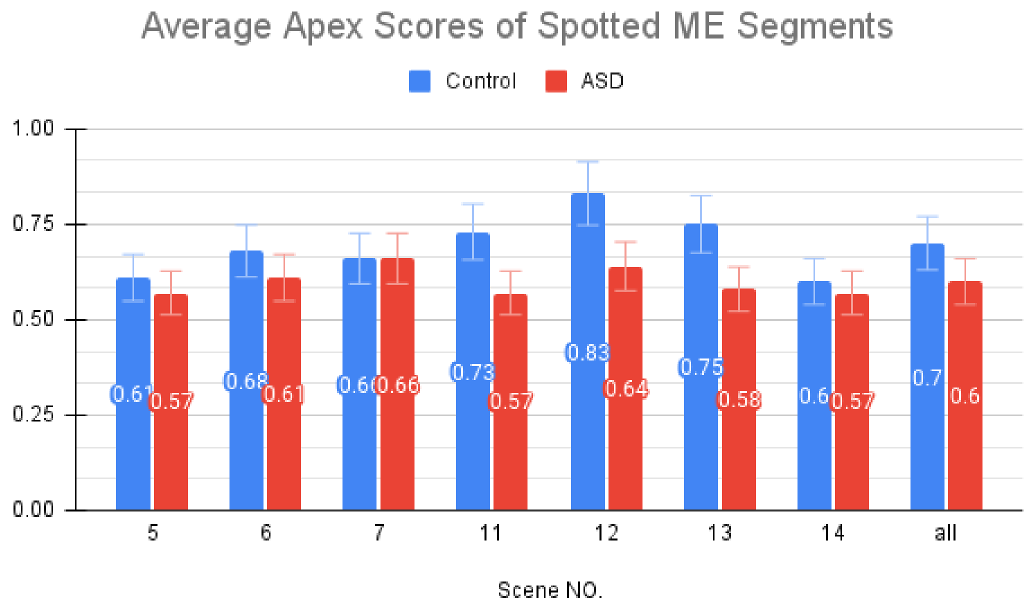


Figure 4.15: Average apex score of spotted micro-expression movements on each scene.

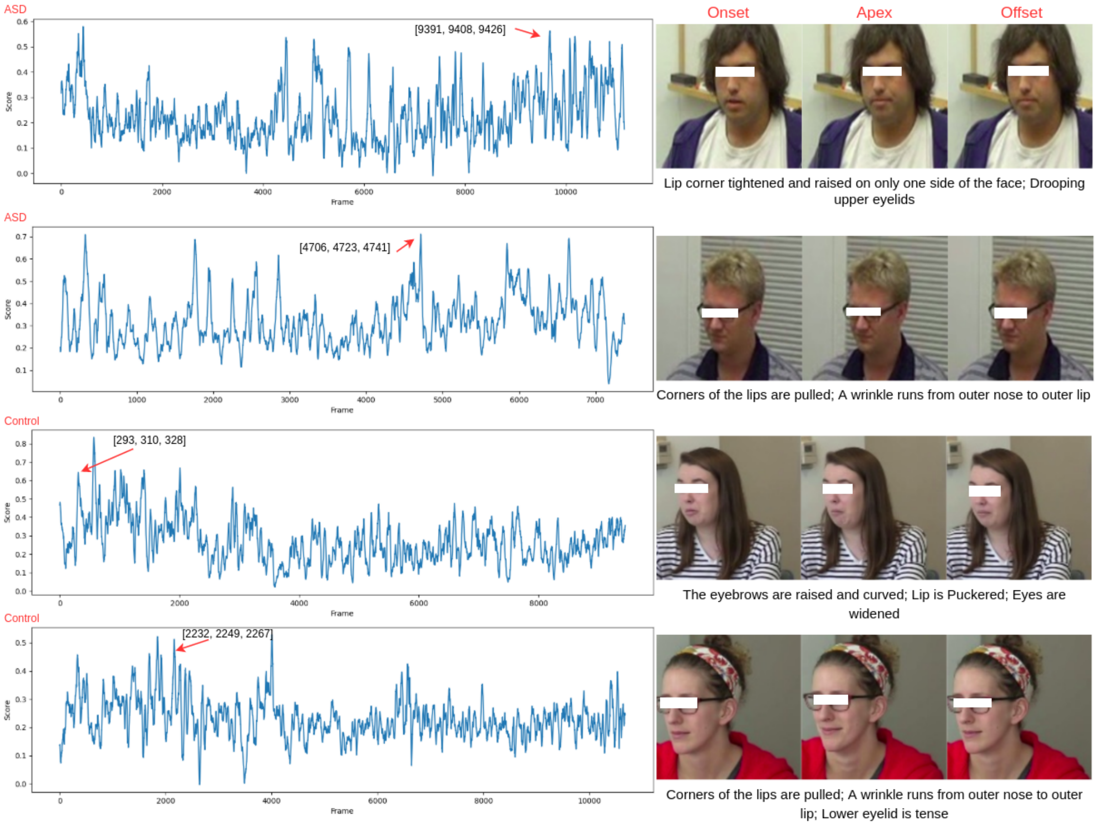
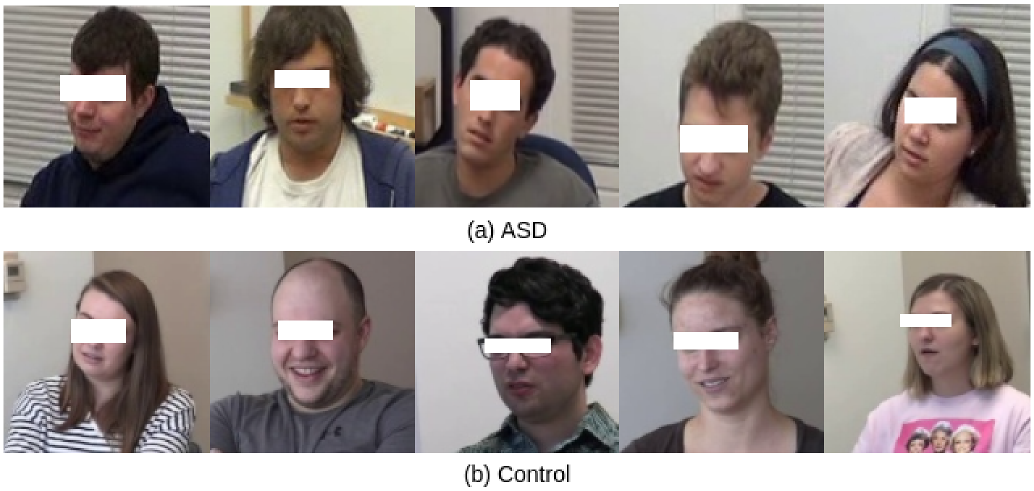


Figure 4.16: Some samples of spotted micro-expression movements on ADOS.

4.5.2 Spotting Result Analysis

Figs. 4.13 $\sim$ 4.15 show the statistics distribution of ASD and control groups on each scene. It is known that different people may perform different facial actions in terms of number of involved face units, scale of face areas, etc., so different input videos have different thresholds (mentioned in Eq. 4.7) to detect micro-expression. It can be found from Fig. 4.13 that control group presents higher threshold than ASD group, especially in scenes 11$\sim$13. In Fig. 4.14, one can see that more micro-expression movements in ASD group are spotted than that in control group. Scenes 12 and 6 even double the average number. It seems ASD participants show more micro-expression movements during the interview in terms of number. One of possible reasons may be the lower threshold calculated in ASD videos. Besides, as Fig. 4.15 shows, the control group shows higher apex scores than ASD which means the subtle facial movement changes on faces in control group are easier to be noticed.


(a) ASD

(b) Control

Figure 4.17: More facial expression samples of (a) ASD and (b) Control on ADOS.

Fig. 4.16 gives four samples of micro-expression movements. Left figure shows scores distribution of all frames in the scene, and right figure shows the onset, apex and offset frames of one selected micro-expression segment, and possible facial movements. By visual check, it is noted that ASD participants have more trouble making spontaneous expressions than control group. ASD participants tend to remain expressionless, less smiling. They produce looks that are odd or difficult to interpret, sometimes give ambiguous looks. Some examples are shown in Fig. 4.17.

4.5.3 Performance of Individual Scenes

Seven selected scenes (5, 6, 7, 11, 12, 13, 14) are experimented with ten-fold cross-validation based on subjects, respectively. The subjects in train and test sets are non-overlapping. We try different swapping ratio between onset and apex input frames to extract F_{DMA} feature of each micro-expression segment using well-trained Micron-BERT model. SVM is adopted for final binary classification. Majority voting is used to decide the final category of the subject belongs to. If more than half micro-expression segments of the

Table 4.7: Performance of our method on different scenes. Accu. - Accuracy.

Swapping	0		0.3		0.5		0.7		1.0	
Ratio	Accu.	F1 score	Accu.	F1 score	Accu.	F1 score	Accu.	F1 score	Accu.	F1 score
Scene 5	0.9482	0.9421	0.9481	0.9421	0.9482	0.9482	0.9421	0.9421	**0.9482**	0.9421
Scene 6	0.8982	0.8857	0.8982	0.8857	0.8857	0.8730	0.8982	0.8857	0.8982	0.8857
Scene 7	0.9107	0.9027	0.8982	0.8887	0.8982	0.8887	0.9125	0.9030	0.8982	0.8887
Scene 11	0.9458	0.9435	0.9458	0.9435	0.9333	0.9265	0.9458	0.9435	0.9333	0.9265
Scene 12	0.9357	0.9337	0.9357	0.9337	**0.9607**	**0.9603**	0.9232	0.9167	0.9232	0.9197
Scene 13	0.9446	0.9433	0.9446	0.9433	0.9446	0.9432	0.9446	0.9433	0.9446	**0.9433**
Scene 14	0.8917	0.8960	0.8917	0.8960	0.9042	0.913	0.9042	0.9130	0.9042	0.9130
Top 3	**0.9607**	**0.9590**	**0.9607**	**0.9590**	0.9482	0.9463	**0.9732**	**0.9730**	**0.9482**	0.9421
Top 5	0.9482	0.9463	**0.9607**	**0.9590**	0.9482	0.9421	0.9607	0.9603	**0.9482**	0.9421
Top 7	0.9482	0.9463	0.9482	0.9463	0.9357	0.9294	0.9482	0.9463	0.9357	0.9294

subject are classified as ASD, the final category of this subject is treated as ASD. Table 4.7 (upper part) shows accuracy and F1 score on individual scenes. Here, swapping ratio means the rate of patches from apex that are used in final swapped frame.

We explored five swapping ratio: 0, 0.3, 0.5, 0.7, 1.0. Here, 0 means onset as swapped frame, and 1.0 means apex as the swapped frame. From the result, we can see experiment with 0.5 swapping ratio gained the highest accuracy 0.9607 on scene 12, and the other ratios obtained similar highest accuracy around 0.9482. The performance among different scenes varies too. In almost cases, scenes 5, 11, 12, 13 gained pretty higher accuracy than scenes 14, 6, 7. It seems that discussion on work, study, and personal life of participants, could reveal more facial micro-expression movements than talking about their social difficulties and annoyance, events/objects that elicit different feelings.

4.5.4 Performance of Scene-level Fusion

We have also studied the performance by fusing several scenes together at the decision level (i.e., via majority voting). The fusion experiments are conducted for the top three, top 5, and top 7 scenes, respectively. The experimental results are shown in Table 4.7 (lower part). From the table, one can see that the highest accuracy **0.9732** among whole experiments occurs on top-3 fusion experiment with 0.7 swapping ratio. And with the addition of more scenes, the performance reduces gradually. In other cases, except 0.5, the

Table 4.8: Performance on different scenes by shuffling test label. Accu. - Accuracy.

Swapping	0		0.3		0.5		0.7		1.0	
Ratio	Accu.	F1 score	Accu.	F1 score	Accu.	F1 score	Accu.	F1 score	Accu.	F1 score
Scene 5	0.5481	0.5421	0.5635	0.5605	0.5465	0.5438	0.5528	0.5415	0.5569	0.5518
Scene 6	0.5229	0.5049	0.5137	0.5019	0.5452	0.5306	0.5134	0.4954	0.5086	0.4947
Scene 7	0.5431	0.5365	0.5173	0.5064	0.5291	0.5227	0.5636	0.5581	0.5104	0.5058
Scene 11	0.5134	0.5073	0.5115	0.5051	0.5518	0.5480	0.5267	0.5228	0.5119	0.5045
Scene 12	0.5749	0.5708	0.5663	0.5564	0.5872	0.5873	0.5366	0.5262	0.5769	0.5709
Scene 13	0.6011	0.6001	0.5903	0.5881	0.5907	0.5882	0.5970	0.5929	0.6002	0.5963
Scene 14	0.5047	0.5046	0.5090	0.5143	0.5060	0.5125	0.4982	0.5071	0.5339	0.5371

highest accuracy also occurs in top-3 fusion. It is noted that it is not always effective to improve the performance by combining more scenes.

4.5.5 Ablation Study

Does the model really capture discriminative micro-expression characteristics of ASD and control? We did a randomization test to check it by shuffling the labels of test data at random. When the shuffled test data are fed into well-trained classifier for prediction, in theory, the accuracy should be around 50%, which means random guessing. The results of our ablation study can be seen in Fig. 4.8. One can clearly observe that for most scenes in different swapping ratios the accuracy occurs around 0.5.

4.6 Discussion and Limitation

Although our model performs well on the ADOS interview videos, it still has limitations in realistic operations. First, our experiment adopts a scene-level fusion strategy, which requires manually splitting the entire hour-long videos into 15 separate scenes by time markers, and extracting features for each scene. It brings in modest time costs. Second, in facial dynamics analysis study, to reduce the bias of data distribution, a distribution calibration strategy is adopted in the few-shot learning module. It needs a base data with sufficient examples to estimate the mean and variance of each class, and transfers the statistics to calibrate the distribution of our data containing only a limited number of samples.

In our experiment, the extracted features of all selected scenes are treated as base data. Although it works, the volume of base data is not enough. It is better to obtain more base data for calibration. Third, in facial micro-expression analysis study, people with ASD usually do not show the emotions in a way that normal people would be able to recognize and understand, either they do not respond emotionally or their emotional responses might sometimes seem over-reaction. Over-reaction response is more like macro-expression, so our spotting model may fail to detect them, which needs further analysis in this situation. Last, there are several challenges for micro-expression analysis. The spotting task, relies on setting the optimal thresholds for any given feature. Different people may perform different extra facial actions, like some people blink habitually, while other people sniff more frequently. When recording videos, many comprehensive factors may significantly influence the micro-expression spotting, such as head movement, physical activity, recording environment, lighting condition, etc.

CHAPTER 5
CONCLUSION AND FUTURE WORK

In this dissertation, we discussed the face analysis application based on images and videos. In image analysis, face morphing attack detection and generation tasks are proposed in detail. In video analysis, we talked about the method of facial dynamics feature analysis, and facial micro-expression analysis from Autism Diagnostic Observation Schedule videos.

With the development of machine learning and deep learning technology, the research on face related analysis will keep ongoing. In the future, we will keep the study on generating new types of morphed faces on attack side. For morphing detection, the study of feasibility of detecting novel attacks as well as morphed face images from printed and scanned image data is also a promising points. As the 3D face photos appears, and the 3D face recognition is widely studied, 3D face morphing should gain more attention in the future. Hence, studying the feasibility of generating and detecting 3D morphed faces is needed too.

For autism diagnosis, so far, we have studied the facial dynamics features, and a more specific feature, facial micro-expression, on ADOS videos. Gaze estimation of participants in videos should also be a good point for autism diagnosis. As the fast development of multimodal fusion technology, it is feasible to conduct multimodal data fusion on this dataset, including both video and audio, by extracting a rich set of features, such as face, gaze, action, speech.

5.1 Face Morphing

Face morphing attacks pose a serious security threat to FRS. In this work, we proposed a few-shot learning framework for no-reference morphing attack detection and fingerprinting problem based on factorized bilinear coding of two types of camera fingerprint feature, PRNU and Noiseprint. Besides, a large-scale database which contains five types of face datasets and eight different morphing methods is collected to evaluate the proposed few-shot MAD and fingerprinting problem. The results show outstanding performance of the proposed fusion-based few-shot MAF framework on our newly collected large-scale morphing dataset.

We note that face morphing attack and defense research is likely to co-evolve. We designed a transformer-based GAN on the attack side. This powerful face morphing attacks model is based on a pre-trained GANformer [18] model which is trained on FFHQ [23] dataset. We generate morphing result by interpolating the latent codes of two bona fide faces. During the stage of extracting latent code of input face image, we combined perceptual loss [250], MSE loss, landmark-based Wing Loss [249] and face matching distance based on HOG feature. Finally, we created more realistic and higher quality morphed images with better identity preservation qualities.

5.1.1 Morphing Attacks

We can do a lot based on previous work. For morphing attacks, we can try to generate new types of morphed faces using various models and different bona fide faces sources. In [19], based on GANformer [18], the author proposed a compositional transformers, GANformer2, which is an iterative object-oriented transformer. It incorporates strong and explicit structural priors, to reflect the compositional nature of visual scenes, and synthesizes images through a sequential process. Another possible model is denoising diffusion probabilistic models [304], which is a class of latent variable models inspired by considerations from nonequilibrium thermodynamics.

3D face recognition is widely studied in recent years that has resulted in a few of real-life security-based applications using 3D face photos, such as national ID cards [305, 306, 307], driving license cards [305], and automatic border control gates (ABC) [308]. In the near future, the use of 3D will be realistic, especially in the border control scenario as both

ICAO 9303 [309], and ISO/IEC 19794-5 [310] standards are well defined to accommodate the 3D face model in the 3rd generation epassport. These can help human observers and automatic FRS to obtain accurate, secure, and reliable identity verification. Based on these factors, investigating the feasibility of generating 3D face morphing and studying their vulnerability and detection are in demand.

Face morphing attacks have received increasing attention in recent years. Generation approaches such as GAN-based are among the leading techniques. However, existing methods suffer from noticeable blurring and synthetic-like generation artifacts. In this paper, we designed a transformer-based alternative to face morphing, which demonstrated its superiority to StyleGAN-based methods. Four particular loss functions were employed to maximize the similarity between the generated face image and the target face image. We also extended the study of transformer-based face morphing to demorphing, the dual operation. Future work includes an improved understanding of the trade-off between vulnerability and detectability as well as other morphing approaches such as diffusion models [280].

5.1.2 Morphing Defense

On detection side, the defense models including both MAD and MAF might focus on the study of feasibility of detecting novel attacks as well as morphed face images from printed and scanned image data in both single image based MAD and non-reference MAD fields. There are several detection methods based on printed and scanned image data [311, 312, 313], which would be a good start for us for further study.

In comparison to the development of single image based morphing attack detection, differential image based detection with trusted live capture lags behind. The commonly used methods contains feature difference and de-morphing. For the feature difference method, the detection performance is sensitive to the type of image data and features, and the segmentation of face region. And for de-morphing methods, the performance is sensitive to the facial poses and imaging conditions. What's more, it requires constrained image data, i.e., trusted live capture. Based on our work finished work of morphing attacks generation using GANformer-based model, we can also design a framework that can performs de-morphing using GANformer-based model as a base model.

5.2 Face Video Analysis in Autism

We have studied the feasibility of developing a method for autism analysis using the ADOS interview video data, based on the facial features.

5.2.1 Facial Trait Classification

The facial trait classification model first extracts the spatio-temporal features of the video and uses the combination of K-SVD with MFA to get more discriminative representations. A few-shot learning module is designed to further improve the classification performance. The experimental results have shown that the proposed approach has a reasonably good result.

The study is significant. First, an effective method is proposed to analyze human facial behaviors from the ADOS interview videos. Second, objectively measuring and analyzing human facial behaviors provides an objective characterization of atypical behaviors in autism. Although a large literature documents abnormal social communicative behavior in autism, essentially all of them have focused on an extremely narrow aspect, typically conducting facial expression on images or videos shown on a screen, without real social interactions as the ADOS data. Third, using a ML method to characterize behavior and/or score ADOS videos will make it much faster and efficient. The algorithms can eventually serve as a screening tool to facilitate the analysis of hour-long ADOS videos.

5.2.2 Facial Micro-Expression Analysis

Recently facial micro-expression analysis has attracted great interest in wide application areas such as behavior analysis, video communication, criminal investigation, and clinical diagnosis. Micro-expressions often reveal true emotions that a person is attempting to suppress, hide, mask, or conceal. These expressions often reflect a person's real emotional state. So, micro-expressions are especially meaningful in ADOS diagnosis. Hence, we also utilize computer vision and machine learning methods to analyze face micro-expressions of the participants in the hour-long ADOS video sequences for the diagnosis of ASD. This work contains two key steps: micro-expression spotting and feature extraction. We first use a spotting model to locate the onset, apex, and offset of each micro-expression movements in the videos, and then use a BERT-based auto-encoder model to encoder onset and apex frames into latent feature space to capture subtle facial movements among apex and onset frames for autism diagnosis.

So far, we have studied the facial features of ADOS. The future work will focus on multimodal data fusion of ADOS videos including both video and audio, by extracting a rich set of features (e.g. face, gaze, action, speech). The next line of research will be "human-centered computing", i.e., centered on the ASD patient. The candidate research directions contain gaze pattern analysis by eye-tracking methods, human behavior analysis by human-object interaction technique, speech traits analysis by natural language processing technology, etc. For example, Gaze pattern analysis is more related to face characteristic. We know, understanding where people are looking is an informative social cue, which can be used for autism diagnosis too. Gaze360 [314] is a gaze-tracking method for robust 3D gaze estimation in unconstrained images. It consider both spatial information and temporal information, and directly output an estimate of gaze uncertainty. The experiment shows its generalization performance via a cross-dataset evaluation. This model is perfect to be applied to real-world use cases, like our ADOS videos, to estimate the ASD participants' focus of attention during the interview. Currently, we are also collecting ADOS videos about young kids. In the future, we will perform more useful characteristics analysis of young children for autism diagnosis.

REFERENCES

[1] A. Alzubaidi and J. Kalita, "Authentication of smartphone users using behavioral biometrics," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1998–2026, 2016.

[2] S. Venkatesh, R. Ramachandra, K. Raja, and C. Busch, "Face morphing attack generation and detection: A comprehensive survey," *IEEE transactions on technology and society*, vol. 2, no. 3, pp. 128–145, 2021.

[3] S. Mallick, "Opencv," in *https://learnopencv.com/face-morph-using-opencv-cpp-python/*, LearnOpenCV, Accessed: August 2021.

[4] yao pang, "Facemorpher," in *https://github.com/yaopang/FaceMorpher*, Accessed: August 2021.

[5] N. Damer, A. M. Saladie, 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)*, IEEE, 2018, pp. 1–10.

[6] L. DeBruine, "Webmorph morphing tool," in *https://github.com/debruine/webmorph*, Online Accessed: 2021, 2016.

[7] 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.

[8] 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*, 2020, pp. 8110–8119.

[9] A. Vaswani *et al.*, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.

[10] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, “Transformer-xl: Attentive language models beyond a fixed-length context,” *arXiv preprint arXiv:1901.02860*, 2019.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[12] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, “End-to-end object detection with transformers,” in *European conference on computer vision*, Springer, 2020, pp. 213–229.

[13] H. Chen *et al.*, “Pre-trained image processing transformer,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 299–12 310.

[14] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, “Swinir: Image restoration using swin transformer,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1833–1844.

[15] R. Liu *et al.*, “Fuseformer: Fusing fine-grained information in transformers for video inpainting,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 040–14 049.

[16] Y. Zeng, J. Fu, and H. Chao, “Learning joint spatial-temporal transformations for video inpainting,” in *European Conference on Computer Vision*, Springer, 2020, pp. 528–543.

[17] B. Zhang *et al.*, “Styleswin: Transformer-based gan for high-resolution image generation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 304–11 314.

[18] D. A. Hudson and L. Zitnick, “Generative adversarial transformers,” in *International conference on machine learning*, PMLR, 2021, pp. 4487–4499.

[19] D. Arad Hudson and L. Zitnick, “Compositional transformers for scene generation,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 9506–9520, 2021.

[20] L. Zhao, Z. Zhang, T. Chen, D. Metaxas, and H. Zhang, “Improved transformer for high-resolution gans,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 18 367–18 380, 2021.

[21] Z. Liu *et al.*, “Swin transformer: Hierarchical vision transformer using shifted windows,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 012–10 022.

[22] Y. Jiang, S. Chang, and Z. Wang, “Transgan: Two pure transformers can make one strong gan, and that can scale up,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 14 745–14 758, 2021.

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

[24] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR’05)*, Ieee, vol. 1, 2005, pp. 886–893.

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

[26] M. Ferrara, A. Franco, and D. Maltoni, “Face demorphing in the presence of facial appearance variations,” in *2018 26th European Signal Processing Conference (EUSIPCO)*, IEEE, 2018, pp. 2365–2369.

[27] A. Röttcher, U. Scherhag, and C. Busch, "Finding the suitable doppelgänger for a face morphing attack," in *2020 IEEE International Joint Conference on Biometrics (IJCB)*, IEEE, 2020, pp. 1–7.

[28] U. Scherhag *et al.*, "Biometric systems under morphing attacks: Assessment of morphing techniques and vulnerability reporting," in *2017 International Conference of the Biometrics Special Interest Group (BIOSIG)*, IEEE, 2017, pp. 1–7.

[29] N. Zhang, X. Liu, X. Li, and G.-J. Qi, "Morphganformer: Transformer-based face morphing and de-morphing," *arXiv preprint arXiv:2302.09404*, 2023.

[30] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," *arXiv preprint arXiv:1703.05175*, 2017.

[31] N. Yu, L. S. Davis, and M. Fritz, "Attributing fake images to gans: Learning and analyzing gan fingerprints," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7556–7566.

[32] J. Lukas, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, 2006.

[33] D. Cozzolino and L. Verdoliva, "Noiseprint: A cnn-based camera model fingerprint," *arXiv preprint arXiv:1808.08396*, 2018.

[34] N. Zhang, S. Jia, S. Lyu, and X. Li, "Fusion-based few-shot morphing attack detection and fingerprinting," *arXiv preprint arXiv:2210.15510*, 2022.

[35] R. L. Birdwhistell, "Toward analyzing american movement," *Nonverbal communication*, pp. 134–143, 1974.

[36] P. Ekman and W. V. Friesen, "Nonverbal behavior and psychopathology," *The psychology of depression: Contemporary theory and research*, pp. 3–31, 1974.

[37] C. Lord *et al.*, “Austism diagnostic observation schedule: A standardized observation of communicative and social behavior,” *Journal of autism and developmental disorders*, vol. 19, no. 2, pp. 185–212, 1989.

[38] R. Cao *et al.*, “Comprehensive social trait judgments from faces in autism spectrum disorder,”

[39] L. Wen, X. Li, G. Guo, and Y. Zhu, “Automated depression diagnosis based on facial dynamic analysis and sparse coding,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 7, pp. 1432–1441, 2015.

[40] N. Zhang, M. Ruan, S. Wang, L. Paul, and X. Li, “Discriminative few shot learning of facial dynamics in interview videos for autism trait classification,” *IEEE Transactions on Affective Computing*, 2022.

[41] B. Allaert, I. M. Bilasco, and C. Djeraba, “Micro and macro facial expression recognition using advanced local motion patterns,” *IEEE Transactions on Affective Computing*, vol. 13, no. 1, pp. 147–158, 2019.

[42] M. A. Rashidan *et al.*, “Technology-assisted emotion recognition for autism spectrum disorder (asd) children: A systematic literature review,” *IEEE Access*, vol. 9, pp. 33 638–33 653, 2021.

[43] G. Guo and N. Zhang, “A survey on deep learning based face recognition,” *Computer vision and image understanding*, vol. 189, p. 102 805, 2019.

[44] G. Guo and N. Zhang, “What is the challenge for deep learning in unconstrained face recognition?” In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, IEEE, 2018, pp. 436–442.

[45] D. ICAO, “9303-machine readable travel documents-part 9: Deployment of biometric identification and electronic storage of data in emrtds,” *International Civil Aviation Organization (ICAO)*, 2015.

[46] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," *IBM systems Journal*, vol. 40, no. 3, pp. 614–634, 2001.

[47] C. Hu, S. Jia, F. Zhang, and X. Li, "A saliency-guided street view image inpainting framework for efficient last-meters wayfinding," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 195, pp. 365–379, 2023.

[48] S. Jia, X. Li, and S. Lyu, "Model attribution of face-swap deepfake videos," in *2022 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2022, pp. 2356–2360.

[49] S. Marcel, M. S. Nixon, and S. Z. Li, *Handbook of biometric anti-spoofing*. Springer, 2014, vol. 1.

[50] R. Ramachandra and C. Busch, "Presentation attack detection methods for face recognition systems: A comprehensive survey," *ACM Computing Surveys (CSUR)*, vol. 50, no. 1, pp. 1–37, 2017.

[51] S. Jia, X. Li, C. Hu, G. Guo, and Z. Xu, "3d face anti-spoofing with factorized bilinear coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 4031–4045, 2020.

[52] S. Jia, X. Li, C. Hu, and Z. Xu, "Spoofing and anti-spoofing with wax figure faces," *arXiv preprint arXiv:1910.05457*, 2019.

[53] S. Jia, C. Hu, X. Li, and Z. Xu, "Face spoofing detection under super-realistic 3d wax face attacks," *Pattern Recognition Letters*, vol. 145, pp. 103–109, 2021.

[54] S. Jia, C. Hu, G. Guo, and Z. Xu, "A database for face presentation attack using wax figure faces," in *New Trends in Image Analysis and Processing–ICIAP 2019: ICIAP International Workshops, BioFor, PatReCH, e-BADLE, DeepRetail, and Industrial Session, Trento, Italy, September 9–10, 2019, Revised Selected Papers 20*, Springer, 2019, pp. 39–47.

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

[56] 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)*, IEEE, 2017, pp. 1–6.

[57] M. Ferrara, A. Franco, and D. Maltoni, “On the effects of image alterations on face recognition accuracy,” in *Face recognition across the imaging spectrum*, Springer, 2016, pp. 195–222.

[58] S. Daily, “Morphing,” in *https://www.sciencedaily.com/terms/morphing.htm*, Science Daily, Accessed: 05.2020.

[59] I Standard, “Information technology—biometric presentation attack detection—part 1: Framework,” *ISO: Geneva, Switzerland*, 2016.

[60] GIMP, “Gnu image manipulation program (gimp),” in *https://www.gimp.org*, Online Accessed: 2021, 2016.

[61] H. Zhang, S. Venkatesh, R. Ramachandra, K. Raja, N. Damer, and C. Busch, “Mipgan–generating robust and high quality morph attacks using identity prior driven gan,” *arXiv e-prints*, arXiv–2009, 2020.

[62] M. Thing, “Morphthing tool,” in *https://www.morphthing.com*, Online Accessed: 2021, 2021.

[63] 3Dthis, “3dthis face morph tool,” in *https://3dthis.com/morph.htm*, Online Accessed: 2021, 2021.

[64] faceswap, “Face swap online,” in *https://faceswaponline.com/*, Online Accessed: 2021, 2021.

[65] FantaMorph, “Abrosoft fantamorph tool,” in *https://3dthis.com/morph.htm*, Online Accessed: 2021, 2021.

[66] L. Development, “Facemorpher tool,” in *http://www.facemorpher.com/*, Online Accessed: 2021, 2021.

[67] 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.

[68] S. Milborrow and F. Nicolls, “Active shape models with sift descriptors and mars,” in *2014 International Conference on Computer Vision Theory and Applications (VISAPP)*, IEEE, vol. 2, 2014, pp. 380–387.

[69] 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*, 2014, pp. 1867–1874.

[70] AMSL, “Amsl face morph image data set,” in *https://omen.cs.uni-magdeburg.de/disclaimer/index.php*, Online Accessed: 2021, 2021.

[71] N. Damer, F. Boutros, A. M. Saladie, F. Kirchbuchner, and A. Kuijper, “Realistic dreams: Cascaded enhancement of gan-generated images with an example in face morphing attacks,” in *2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, IEEE, 2019, pp. 1–10.

[72] S. Venkatesh, R. Ramachandra, K. Raja, and C. Busch, “Face morphing attack generation & detection: A comprehensive survey,” *arXiv preprint arXiv:2011.02045*, 2020.

[73] 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)*, IEEE, 2018, pp. 1–9.

[74] L. Debiasi, U. Scherhag, C. Rathgeb, A. Uhl, and C. Busch, “Prnu-based detection of morphed face images,” in *2018 International Workshop on Biometrics and Forensics (IWBF)*, IEEE, 2018, pp. 1–7.

[75] 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.

[76] 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)*, IEEE, 2018, pp. 1–6.

[77] 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 *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 280–289.

[78] K. Raja *et al.*, “Morphing attack detection–database, evaluation platform and benchmarking,” *arXiv preprint arXiv:2006.06458*, 2020.

[79] 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*, 2017, pp. 10–18.

[80] C. Seibold, A. Hilsmann, and P. Eisert, “Style your face morph and improve your face morphing attack detector,” in *2019 International Conference of the Biometrics Special Interest Group (BIOSIG)*, 2019, pp. 1–6.

[81] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[82] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[83] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

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

[85] 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)*, IEEE, 2019, pp. 1–6.

[86] P. Aghdaie, B. Chaudhary, S. Soleymani, J. Dawson, and N. M. Nasrabadi, "Attention aware wavelet-based detection of morphed face images," *arXiv preprint arXiv:2106.15686*, 2021.

[87] 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)*, IEEE, 2018, pp. 187–192.

[88] N. Damer, S. Zienert, Y. Wainakh, A. M. Saladié, F. Kirchbuchner, and A. Kuijper, "A multi-detector solution towards an accurate and generalized detection of face morphing attacks," in *2019 22th International Conference on Information Fusion (FUSION)*, IEEE, 2019, pp. 1–8.

[89] 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*, Springer, 2018, pp. 444–452.

[90] 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)*, IEEE, 2019, pp. 106–112.

[91] 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. 92 301–92 313, 2020.

[92] F. Peng, L.-B. Zhang, and M. Long, “Fd-gan: Face de-morphing generative adversarial network for restoring accomplice’s facial image,” *IEEE Access*, vol. 7, pp. 75 122–75 131, 2019.

[93] S. Banerjee and A. Ross, “Conditional identity disentanglement for differential face morph detection,” *arXiv preprint arXiv:2107.02162*, 2021.

[94] G. Borghi, E. Pancisi, M. Ferrara, and D. Maltoni, “A double siamese framework for differential morphing attack detection,” *Sensors*, vol. 21, no. 10, p. 3466, 2021.

[95] S. Banerjee, P. Jaiswal, and A. Ross, “Facial de-morphing: Extracting component faces from a single morph,” *arXiv preprint arXiv:2209.02933*, 2022.

[96] E. Shiqerukaj, C. Rathgeb, J. Merkle, P. Drozdowski, and B. Tams, “Fusion of face demorphing and deep face representations for differential morphing attack detection,” in *2022 International Conference of the Biometrics Special Interest Group (BIOSIG)*, IEEE, 2022, pp. 1–5.

[97] S. Ravi and H. Larochelle, “Optimization as a model for few-shot learning,” 2016.

[98] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, “Learning to compare: Relation network for few-shot learning,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1199–1208.

[99] Q. Sun, Y. Liu, T.-S. Chua, and B. Schiele, “Meta-transfer learning for few-shot learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 403–412.

[100] T. Ramalho and M. Garnelo, “Adaptive posterior learning: Few-shot learning with a surprise-based memory module,” *arXiv preprint arXiv:1902.02527*, 2019.

[101] X. Luo *et al.*, “Rectifying the shortcut learning of background for few-shot learning,” *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[102] D. Cozzolino, F. Marra, D. Gragnaniello, G. Poggi, and L. Verdoliva, “Combining prnu and noiseprint for robust and efficient device source identification,” *EURASIP Journal on Information Security*, vol. 2020, no. 1, pp. 1–12, 2020.

[103] D. A. Salazar, A. E. Ramirez-Rodriguez, M. Nakano, M. Cedillo-Hernandez, and H. Perez-Meana, “Evaluation of denoising algorithms for source camera linking,” in *Mexican Conference on Pattern Recognition*, Springer, 2021, pp. 282–291.

[104] X. Lin and C.-T. Li, “Prnu-based content forgery localization augmented with image segmentation,” *IEEE Access*, vol. 8, pp. 222 645–222 659, 2020.

[105] F. Lugstein, S. Baier, G. Bachinger, and A. Uhl, “Prnu-based deepfake detection,” in *Proceedings of the 2021 ACM Workshop on Information Hiding and Multimedia Security*, 2021, pp. 7–12.

[106] C. Hu, M. Yin, B. Liu, X. Li, and Y. Ye, “Detection of illicit drug trafficking events on instagram: A deep multimodal multilabel learning approach,” in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021, pp. 3838–3846.

[107] C. Hu, M. Yin, B. Liu, X. Li, and Y. Ye, “Identifying illicit drug dealers on instagram with large-scale multimodal data fusion,” *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 12, no. 5, pp. 1–23, 2021.

[108] C. Hu, B. Liu, Y. Ye, and X. Li, "Fine-grained classification of drug trafficking based on instagram hashtags," *Decision Support Systems*, vol. 165, p. 113 896, 2023.

[109] C.-B. Hu, F. Zhang, F.-Y. Gong, C. Ratti, and X. Li, "Classification and mapping of urban canyon geometry using google street view images and deep multitask learning," *Building and Environment*, vol. 167, p. 106 424, 2020.

[110] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[111] H. Bao, L. Dong, S. Piao, and F. Wei, "Beit: Bert pre-training of image transformers," *arXiv preprint arXiv:2106.08254*, 2021.

[112] I. A. Rosenthal, C. A. Hutcherson, R. Adolphs, and D. A. Stanley, "Deconstructing theory-of-mind impairment in high-functioning adults with autism," *Current Biology*, vol. 29, no. 3, pp. 513–519, 2019.

[113] M. B. Harms, A. Martin, and G. L. Wallace, "Facial emotion recognition in autism spectrum disorders: A review of behavioral and neuroimaging studies," *Neuropsychology review*, vol. 20, no. 3, pp. 290–322, 2010.

[114] K. A. Pelphrey, J. P. Morris, G. McCarthy, and K. S. LaBar, "Perception of dynamic changes in facial affect and identity in autism," *Social cognitive and affective neuroscience*, vol. 2, no. 2, pp. 140–149, 2007.

[115] C. S. Monk *et al.*, "Neural circuitry of emotional face processing in autism spectrum disorders," *Journal of psychiatry & neuroscience: JPN*, vol. 35, no. 2, p. 105, 2010.

[116] O. Golan, Y. Sinai-Gavrilov, and S. Baron-Cohen, "The cambridge mindreading face-voice battery for children (cam-c): Complex emotion recognition in children with and without autism spectrum conditions," *Molecular autism*, vol. 6, no. 1, pp. 1–9, 2015.

[117] P. J. Webster, S. Wang, and X. Li, "Review: Posed vs. genuine facial emotion recognition and expression in autism and implications for intervention," *Frontiers in Psychology*, vol. 12, 2021.

[118] M. Jiang and Q. Zhao, "Learning visual attention to identify people with autism spectrum disorder," in *Proceedings of the ieee international conference on computer vision*, 2017, pp. 3267–3276.

[119] M. Ruan, P. J. Webster, X. Li, and S. Wang, "Deep neural network reveals the world of autism from a first-person perspective," *Autism Research*, vol. 14, no. 2, pp. 333–342, 2021.

[120] J. M. Rehg, "Behavior imaging: Using computer vision to study autism.," *MVA*, vol. 11, pp. 14–21, 2011.

[121] J. M. Rehg, A. Rozga, G. D. Abowd, and M. S. Goodwin, "Behavioral imaging and autism," *IEEE Pervasive Computing*, vol. 13, no. 2, pp. 84–87, 2014.

[122] K. L. Carpenter *et al.*, "Digital behavioral phenotyping detects atypical pattern of facial expression in toddlers with autism," *Autism Research*, vol. 14, no. 3, pp. 488–499, 2021.

[123] Y. Zhu, Y. Shang, Z. Shao, and G. Guo, "Automated depression diagnosis based on deep networks to encode facial appearance and dynamics," *IEEE Transactions on Affective Computing*, vol. 9, no. 4, pp. 578–584, 2017.

[124] G. Yang *et al.*, "An iot-enabled stroke rehabilitation system based on smart wearable armband and machine learning," *IEEE journal of translational engineering in health and medicine*, vol. 6, pp. 1–10, 2018.

[125] C. Hu *et al.*, "Spatiotemporal correlation analysis of hydraulic fracturing and stroke in the united states," *International Journal of Environmental Research and Public Health*, vol. 19, no. 17, p. 10 817, 2022.

[126] S. Thapaliya, “Evaluation of eeg and eye movement with machine learning for the classification of autism spectrum disorder,” Ph.D. dissertation, 2018.

[127] A. B. Dris, A. Alsalman, A. Al-Wabil, and M. Aldosari, “Intelligent gaze-based screening system for autism,” in *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)*, IEEE, 2019, pp. 1–5.

[128] W. Wei, Z. Liu, L. Huang, A. Nebout, and O. Le Meur, “Saliency prediction via multi-level features and deep supervision for children with autism spectrum disorder,” 2019.

[129] H. Duan *et al.*, “Learning to predict where the children with asd look,” in *2018 25th IEEE International Conference on Image Processing (ICIP)*, IEEE, 2018, pp. 704–708.

[130] Y. Tao and M.-L. Shyu, “Sp-asdnet: Cnn-lstm based asd classification model using observer scanpaths,” in *2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, IEEE, 2019, pp. 641–646.

[131] A. Zunino *et al.*, “Video gesture analysis for autism spectrum disorder detection,” in *2018 24th International Conference on Pattern Recognition (ICPR)*, IEEE, 2018, pp. 3421–3426.

[132] A. Crippa *et al.*, “Use of machine learning to identify children with autism and their motor abnormalities,” *Journal of autism and developmental disorders*, vol. 45, no. 7, pp. 2146–2156, 2015.

[133] Y. Tian, X. Min, G. Zhai, and Z. Gao, “Video-based early asd detection via temporal pyramid networks,” in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, 2019, pp. 272–277.

[134] S. Eraslan, Y. Yesilada, V. Yaneva, and S. Harper, “Autism detection based on eye movement sequences on the web: A scanpath trend analysis approach,” in *Proceedings of the 17th International Web for All Conference*, 2020, pp. 1–10.

[135] K. Ahuja *et al.*, "Gaze-based screening of autistic traits for adolescents and young adults using prosaic videos," in *Proceedings of the 3rd ACM SIGCAS Conference on Computing and Sustainable Societies*, 2020, pp. 324–324.

[136] J. Li, Y. Zhong, and G. Ouyang, "Identification of asd children based on video data," in *2018 24th International Conference on Pattern Recognition (ICPR)*, IEEE, 2018, pp. 367–372.

[137] G. Wan *et al.*, "Applying eye tracking to identify autism spectrum disorder in children," *Journal of autism and developmental disorders*, vol. 49, no. 1, pp. 209–215, 2019.

[138] D. N. Fernández, F. B. Porras, R. H. Gilman, M. V. Mondonedo, P. Sheen, and M. Zimic, "A convolutional neural network for gaze preference detection: A potential tool for diagnostics of autism spectrum disorder in children," *arXiv preprint arXiv:2007.14432*, 2020.

[139] W. Liu, M. Li, and L. Yi, "Identifying children with autism spectrum disorder based on their face processing abnormality: A machine learning framework," *Autism Research*, vol. 9, no. 8, pp. 888–898, 2016.

[140] M. Jiang, S. Francis, D. Srishyla, C. Conelea, Q. Zhao, and S. Jacob, "Classifying individuals with asd through facial emotion recognition and eye-tracking," Jul. 2019.

[141] D. A. Kaliukhovich, N. V. Manyakov, A. Bangerter, and G. Pandina, "Context modulates attention to faces in dynamic social scenes in children and adults with autism spectrum disorder," *Journal of Autism and Developmental Disorders*, pp. 1–14, 2021.

[142] M. Leo *et al.*, "Computational analysis of deep visual data for quantifying facial expression production," *Applied Sciences*, vol. 9, no. 21, p. 4542, 2019.

[143] M. Beary, A. Hadsell, R. Messersmith, and M.-P. Hosseini, “Diagnosis of autism in children using facial analysis and deep learning,” *arXiv preprint arXiv:2008.02890*, 2020.

[144] T. Akter *et al.*, “Improved transfer-learning-based facial recognition framework to detect autistic children at an early stage,” *Brain Sciences*, vol. 11, no. 6, p. 734, 2021.

[145] A. Lu and M. Perkowski, “Deep learning approach for screening autism spectrum disorder in children with facial images and analysis of ethnoracial factors in model development and application,” *Brain Sciences*, vol. 11, no. 11, p. 1446, 2021.

[146] A. E. Kowallik, M. Pohl, and S. R. Schweinberger, “Facial imitation improves emotion recognition in adults with different levels of sub-clinical autistic traits,” *Journal of Intelligence*, vol. 9, no. 1, p. 4, 2021.

[147] F. Lecciso *et al.*, “Emotional expression in children with asd: A pre-study on a two-group pre-post-test design comparing robot-based and computer-based training,” *Frontiers in Psychology*, p. 2826, 2021.

[148] C. Guo, K. Zhang, J. Chen, R. Xu, and L. Gao, “Design and application of facial expression analysis system in empathy ability of children with autism spectrum disorder,” in *2021 16th Conference on Computer Science and Intelligence Systems (FedCSIS)*, IEEE, 2021, pp. 319–325.

[149] B. Elshoky, O. A. S. Ibrahim, A. A. Ali, and E. M. Younis, “Comparing automated and non-automated machine learning for autism spectrum disorders classification using facial images,”

[150] A. Bangerter *et al.*, “Automated recognition of spontaneous facial expression in individuals with autism spectrum disorder: Parsing response variability,” *Molecular autism*, vol. 11, no. 1, pp. 1–15, 2020.

[151] B. Banire, D. Al Thani, M. Qaraqe, and B. Mansoor, “Face-based attention recognition model for children with autism spectrum disorder,” *Journal of Healthcare Informatics Research*, vol. 5, no. 4, pp. 420–445, 2021.

[152] J. Q. Zlibut, A. Munshi, G. Biswas, and C. Cascio, “Identifying and describing subtypes of spontaneous empathic facial expression production in autistic adults,” 2021.

[153] G. Alvari, C. Furlanello, and P. Venuti, “Is smiling the key? machine learning analytics detect subtle patterns in micro-expressions of infants with asd,” *Journal of clinical medicine*, vol. 10, no. 8, p. 1776, 2021.

[154] D. K. Oller *et al.*, “Automated vocal analysis of naturalistic recordings from children with autism, language delay, and typical development,” *Proceedings of the National Academy of Sciences*, vol. 107, no. 30, pp. 13 354–13 359, 2010.

[155] C. Ecker *et al.*, “Describing the brain in autism in five dimensions—magnetic resonance imaging-assisted diagnosis of autism spectrum disorder using a multiparameter classification approach,” *Journal of Neuroscience*, vol. 30, no. 32, pp. 10 612–10 623, 2010.

[156] Z. Sherkatghanad *et al.*, “Automated detection of autism spectrum disorder using a convolutional neural network,” *Frontiers in neuroscience*, vol. 13, p. 1325, 2020.

[157] F. Thabtah and D. Peebles, “A new machine learning model based on induction of rules for autism detection,” *Health informatics journal*, vol. 26, no. 1, pp. 264–286, 2020.

[158] G Devika Varshini and R Chinnaiyan, “Optimized machine learning classification approaches for prediction of autism spectrum disorder,” *Ann Autism Dev Disord. 2020; 1 (1)*, vol. 1001,

[159] I. M. Nasser, M. Al-Shawwa, and S. S. Abu-Naser, “Artificial neural network for diagnose autism spectrum disorder,” 2019.

[160] S. Raj and S. Masood, “Analysis and detection of autism spectrum disorder using machine learning techniques,” *Procedia Computer Science*, vol. 167, pp. 994–1004, 2020.

[161] J. Peral, D. Gil, S. Rotbei, S. Amador, M. Guerrero, and H. Moradi, “A machine learning and integration based architecture for cognitive disorder detection used for early autism screening,” *Electronics*, vol. 9, no. 3, p. 516, 2020.

[162] M. D. Hossain, M. A. Kabir, A. Anwar, and M. Z. Islam, “Detecting autism spectrum disorder using machine learning,” *arXiv preprint arXiv:2009.14499*, 2020.

[163] C. Küpper *et al.*, “Identifying predictive features of autism spectrum disorders in a clinical sample of adolescents and adults using machine learning,” *Scientific reports*, vol. 10, no. 1, pp. 1–11, 2020.

[164] F. Z. Subah, K. Deb, P. K. Dhar, and T. Koshiba, “A deep learning approach to predict autism spectrum disorder using multisite resting-state fmri,” *Applied Sciences*, vol. 11, no. 8, p. 3636, 2021.

[165] K. A. Pelphrey, N. J. Sasson, J. S. Reznick, G. Paul, B. D. Goldman, and J. Piven, “Visual scanning of faces in autism,” *Journal of autism and developmental disorders*, vol. 32, no. 4, pp. 249–261, 2002.

[166] A. Klin, W. Jones, R. Schultz, F. Volkmar, and D. Cohen, “Visual fixation patterns during viewing of naturalistic social situations as predictors of social competence in individuals with autism,” *Archives of general psychiatry*, vol. 59, no. 9, pp. 809–816, 2002.

[167] M. Freeth, T. Foulsham, and P. Chapman, “The influence of visual saliency on fixation patterns in individuals with autism spectrum disorders,” *Neuropsychologia*, vol. 49, no. 1, pp. 156–160, 2011.

[168] S. Wang *et al.*, “Atypical visual saliency in autism spectrum disorder quantified through model-based eye tracking,” *Neuron*, vol. 88, no. 3, pp. 604–616, 2015.

[169] S. N. Rigby, B. M. Stoesz, and L. S. Jakobson, "Gaze patterns during scene processing in typical adults and adults with autism spectrum disorders," *Research in Autism Spectrum Disorders*, vol. 25, pp. 24–36, 2016.

[170] R. M. Jones *et al.*, "Increased eye contact during conversation compared to play in children with autism," *Journal of autism and developmental disorders*, vol. 47, no. 3, pp. 607–614, 2017.

[171] S. R. Edmunds *et al.*, "Brief report: Using a point-of-view camera to measure eye gaze in young children with autism spectrum disorder during naturalistic social interactions: A pilot study," *Journal of autism and developmental disorders*, vol. 47, no. 3, pp. 898–904, 2017.

[172] A. Vernetti, A. Senju, T. Charman, M. H. Johnson, T. Gliga, B. Team, *et al.*, "Simulating interaction: Using gaze-contingent eye-tracking to measure the reward value of social signals in toddlers with and without autism," *Developmental cognitive neuroscience*, vol. 29, pp. 21–29, 2018.

[173] E. L. Ajodan *et al.*, "Increased eye contact during parent-child versus clinician-child interactions in young children with autism," 2019.

[174] M.-K. Kwon, A. Moore, C. C. Barnes, D. Cha, and K. Pierce, "Typical levels of eye-region fixation in toddlers with autism spectrum disorder across multiple contexts," *Journal of the American Academy of Child & Adolescent Psychiatry*, 2019.

[175] D. P. Kennedy and R. Adolphs, "Perception of emotions from facial expressions in high-functioning adults with autism," *Neuropsychologia*, vol. 50, no. 14, pp. 3313–3319, 2012.

[176] S. Wang and R. Adolphs, "Reduced specificity in emotion judgment in people with autism spectrum disorder," *Neuropsychologia*, vol. 99, pp. 286–295, 2017.

[177] M. H. Black *et al.*, "Mechanisms of facial emotion recognition in autism spectrum disorders: Insights from eye tracking and electroencephalography," *Neuroscience & Biobehavioral Reviews*, vol. 80, pp. 488–515, 2017.

[178] N. N. Capriola-Hall *et al.*, "Group differences in facial emotion expression in autism: Evidence for the utility of machine classification," *Behavior therapy*, vol. 50, no. 4, pp. 828–838, 2019.

[179] A. C. Miu, S. E. Pană, and J. Avram, "Emotional face processing in neurotypicals with autistic traits: Implications for the broad autism phenotype," *Psychiatry research*, vol. 198, no. 3, pp. 489–494, 2012.

[180] R. C. Leung, E. W. Pang, D. Cassel, J. A. Brian, M. L. Smith, and M. J. Taylor, "Early neural activation during facial affect processing in adolescents with autism spectrum disorder," *NeuroImage: Clinical*, vol. 7, pp. 203–212, 2015.

[181] P. Shah, G. Bird, and R. Cook, "Face processing in autism: Reduced integration of cross-feature dynamics," *cortex*, vol. 75, pp. 113–119, 2016.

[182] S. Wang, S. Sun, R. Cao, K. Kar, and H. Yu, "Multimodal investigations of human face perception in neurotypical and autistic adults," 2023.

[183] J. Wang, R. Cao, N. J. Brandmeir, X. Li, and S. Wang, "Face identity coding in the deep neural network and primate brain," *Communications Biology*, vol. 5, no. 1, p. 611, 2022.

[184] R. Cao *et al.*, "Feature-based encoding of face identity by single neurons in the human amygdala and hippocampus," *BioRxiv*, pp. 2020–09, 2020.

[185] G. Dawson, S. J. Webb, and J. McPartland, "Understanding the nature of face processing impairment in autism: Insights from behavioral and electrophysiological studies," *Developmental neuropsychology*, vol. 27, no. 3, pp. 403–424, 2005.

[186] P. H. J. M. Vlamings, L. M. Jonkman, E. van Daalen, R. J. van der Gaag, and C. Kemner, “Basic abnormalities in visual processing affect face processing at an early age in autism spectrum disorder,” *Biological psychiatry*, vol. 68, no. 12, pp. 1107–1113, 2010.

[187] U. Rutishauser, O. Tudusciuc, S. Wang, A. N. Mamelak, I. B. Ross, and R. Adolphs, “Single-neuron correlates of atypical face processing in autism,” *Neuron*, vol. 80, no. 4, pp. 887–899, 2013.

[188] M. Ruan, *Image and Video-Based Autism Spectrum Disorder Detection via Deep Learning*. West Virginia University, 2020.

[189] M. Ruan, X. Yu, N. Zhang, C. Hu, S. Wang, and X. Li, “Video-based contrastive learning on decision trees: From action recognition to autism diagnosis,” *arXiv preprint arXiv:2304.10073*, 2023.

[190] M. Delowar Hossain, M. Ashad Kabir, A. Anwar, and M. Zahidul Islam, “Detecting autism spectrum disorder using machine learning,” *arXiv e-prints*, arXiv–2009, 2020.

[191] F. F. Thabtah, “Autistic spectrum disorder screening data for adult,” *https://archive.ics.uci.edu/ml/m learning-databases/00426/*, 2017.

[192] F. F. Thabtah, “Autistic spectrum disorder screening data for children,” *https://archive.ics.uci.edu/m learning-databases/00419/*, 2017.

[193] F. F. Thabtah, “Autistic spectrum disorder screening data for adolescent,” *https://archive.ics.uci.edu learning-databases/00420/*, 2017.

[194] R. A. J. de Belen, T. Bednarz, A. Sowmya, and D. Del Favero, “Computer vision in autism spectrum disorder research: A systematic review of published studies from 2009 to 2019,” *Translational psychiatry*, vol. 10, no. 1, pp. 1–20, 2020.

[195] J. A. A. van Rentergem, M. K. Deserno, and H. M. Geurts, "Validation strategies for subtypes in psychiatry: A systematic review of research on autism spectrum disorder," *Clinical Psychology Review*, p. 102 033, 2021.

[196] A Saranya and R Anandan, "Figs-deaf: An novel implementation of hybrid deep learning algorithm to predict autism spectrum disorders using facial fused gait features," *Distributed and Parallel Databases*, pp. 1–26, 2021.

[197] L. Zhang and O. Arandjelović, "Review of automatic microexpression recognition in the past decade," *Machine Learning and Knowledge Extraction*, vol. 3, no. 2, pp. 414–434, 2021.

[198] Y.-H. Oh, J. See, A. C. Le Ngo, R. C.-W. Phan, and V. M. Baskaran, "A survey of automatic facial micro-expression analysis: Databases, methods, and challenges," *Frontiers in psychology*, vol. 9, p. 1128, 2018.

[199] S. Jia, S. Wang, C. Hu, P. J. Webster, and X. Li, "Detection of genuine and posed facial expressions of emotion: Databases and methods," *Frontiers in Psychology*, vol. 11, p. 580 287, 2021.

[200] M. F. Valstar and M. Pantic, "Fully automatic recognition of the temporal phases of facial actions," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 1, pp. 28–43, 2011.

[201] J. Li, C. Soladié, R. Séguier, S.-J. Wang, and M. H. Yap, "Spotting micro-expressions on long videos sequences," in *2019 14th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2019)*, 2019, pp. 1–5.

[202] A. Moilanen, G. Zhao, and M. Pietikäinen, "Spotting rapid facial movements from videos using appearance-based feature difference analysis," in *2014 22nd International Conference on Pattern Recognition*, 2014, pp. 1722–1727.

[203] X. Li *et al.*, “Towards reading hidden emotions: A comparative study of spontaneous micro-expression spotting and recognition methods,” *IEEE Transactions on Affective Computing*, vol. 9, no. 4, pp. 563–577, 2018.

[204] A. Davison, W. Merghani, C. Lansley, C.-C. Ng, and M. H. Yap, “Objective micro-facial movement detection using facs-based regions and baseline evaluation,” in *2018 13th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2018)*, 2018, pp. 642–649.

[205] M. Shreve, J. Brizzi, S. Fefilatyev, T. Luguev, D. Goldgof, and S. Sarkar, “Automatic expression spotting in videos,” *Image and Vision Computing*, vol. 32, no. 8, pp. 476–486, 2014.

[206] J. LI, S.-J. Wang, M. H. Yap, J. See, X. Hong, and X. Li, “Megc2020 - the third facial micro-expression grand challenge,” in *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, 2020, pp. 777–780.

[207] Y. Li, J. Wei, Y. Liu, J. Kauttonen, and G. Zhao, “Deep learning for micro-expression recognition: A survey,” *IEEE Transactions on Affective Computing*, 2022.

[208] Z. Zhou, G. Zhao, and M. Pietikäinen, “Towards a practical lipreading system,” in *CVPR 2011*, 2011, pp. 137–144.

[209] S.-T. Liong, J. See, K. Wong, and R. C.-W. Phan, “Less is more: Micro-expression recognition from video using apex frame,” *Signal Processing: Image Communication*, vol. 62, pp. 82–92, 2018.

[210] S.-J. Wang, W.-J. Yan, G. Zhao, X. Fu, and C.-G. Zhou, “Micro-expression recognition using robust principal component analysis and local spatiotemporal directional features,” in *Computer Vision - ECCV 2014 Workshops*, L. Agapito, M. M. Bronstein, and C. Rother, Eds., Cham: Springer International Publishing, 2015, pp. 325–338.

[211] Y. Zong, X. Huang, W. Zheng, Z. Cui, and G. Zhao, “Learning from hierarchical spatiotemporal descriptors for micro-expression recognition,” *IEEE Transactions on Multimedia*, vol. 20, no. 11, pp. 3160–3172, 2018.

[212] D. Patel, X. Hong, and G. Zhao, “Selective deep features for micro-expression recognition,” in *2016 23rd International Conference on Pattern Recognition (ICPR)*, 2016, pp. 2258–2263.

[213] J. Li, Y. Wang, J. See, and W. Liu, “Micro-expression recognition based on 3d flow convolutional neural network,” *Pattern Analysis and Applications*, vol. 22, no. 4, pp. 1331–1339, 2019.

[214] Z. Xia, X. Hong, X. Gao, X. Feng, and G. Zhao, “Spatiotemporal recurrent convolutional networks for recognizing spontaneous micro-expressions,” *IEEE Transactions on Multimedia*, vol. 22, no. 3, pp. 626–640, 2019.

[215] D. Y. Choi and B. C. Song, “Facial micro-expression recognition using two-dimensional landmark feature maps,” *IEEE Access*, vol. 8, pp. 121 549–121 563, 2020.

[216] X. Nie, M. A. Takalkar, M. Duan, H. Zhang, and M. Xu, “Geme: Dual-stream multi-task gender-based micro-expression recognition,” *Neurocomputing*, vol. 427, pp. 13–28, 2021.

[217] S. Happy and A. Routray, “Fuzzy histogram of optical flow orientations for micro-expression recognition,” *IEEE Transactions on Affective Computing*, vol. 10, no. 3, pp. 394–406, 2017.

[218] F. Xu, J. Zhang, and J. Z. Wang, “Microexpression identification and categorization using a facial dynamics map,” *IEEE Transactions on Affective Computing*, vol. 8, no. 2, pp. 254–267, 2017.

[219] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” *arXiv preprint arXiv:1609.02907*, 2016.

[220] N. Van Quang, J. Chun, and T. Tokuyama, "Capsulenet for micro-expression recognition," in *2019 14th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2019)*, IEEE, 2019, pp. 1–7.

[221] B. Sun, S. Cao, D. Li, J. He, and L. Yu, "Dynamic micro-expression recognition using knowledge distillation," *IEEE Transactions on Affective Computing*, 2020.

[222] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *European conference on computer vision*, Springer, 2004, pp. 25–36.

[223] A. Bruhn, J. Weickert, and C. Schnörr, "Lucas/kanade meets horn/schunck: Combining local and global optic flow methods," *International journal of computer vision*, vol. 61, no. 3, pp. 211–231, 2005.

[224] S. Jiang, Y. Lu, H. Li, and R. Hartley, "Learning optical flow from a few matches," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 592–16 600.

[225] M. Bai, W. Luo, K. Kundu, and R. Urtasun, "Exploiting semantic information and deep matching for optical flow," in *European Conference on Computer Vision*, Springer, 2016, pp. 154–170.

[226] A. Dosovitskiy *et al.*, "Flownet: Learning optical flow with convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2758–2766.

[227] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2462–2470.

[228] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8934–8943.

[229] A. Ranjan and M. J. Black, "Optical flow estimation using a spatial pyramid network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4161–4170.

[230] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *European conference on computer vision*, Springer, 2020, pp. 402–419.

[231] G. Yang and D. Ramanan, "Volumetric correspondence networks for optical flow," *Advances in neural information processing systems*, vol. 32, 2019.

[232] T.-W. Hui, X. Tang, and C. C. Loy, "Liteflownet: A lightweight convolutional neural network for optical flow estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8981–8989.

[233] C. Zhao, C. Feng, D. Li, and S. Li, "Of-msrn: Optical flow-auxiliary multi-task regression network for direct quantitative measurement, segmentation and motion estimation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 1218–1225.

[234] S. Jiang, D. Campbell, Y. Lu, H. Li, and R. Hartley, "Learning to estimate hidden motions with global motion aggregation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9772–9781.

[235] M. Hofinger, S. R. Bulò, L. Porzi, A. Knapitsch, T. Pock, and P. Kontschieder, "Improving optical flow on a pyramid level," in *European Conference on Computer Vision*, Springer, 2020, pp. 770–786.

[236] Z. Huang *et al.*, "Flowformer: A transformer architecture for optical flow," *arXiv preprint arXiv:2203.16194*, 2022.

[237] Y. Zheng, M. Zhang, and F. Lu, “Optical flow in the dark,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6749–6757.

[238] W. Yan, A. Sharma, and R. T. Tan, “Optical flow in dense foggy scenes using semi-supervised learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 259–13 268.

[239] Z. Huang, X. Pan, R. Xu, Y. Xu, G. Zhang, H. Li, *et al.*, “Life: Lighting invariant flow estimation,” *arXiv preprint arXiv:2104.03097*, 2021.

[240] NIST, “Feret,” in *https://www.nist.gov/programs-projects/face-recognition-technology-feret*, Information Technology Laboratory, Information Access Division Image Group, 2011.

[241] 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.

[242] NIST, “Face recognition grand challenge (frgc),” in *https://www.nist.gov/programs-projects/face-recognition-grand-challenge-frgc*, Information Technology Laboratory, Information Access Division Image Group, 2010.

[243] L. set, “Face research lab london set,” 2017.

[244] 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.

[245] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep learning face attributes in the wild],” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3730–3738.

[246] I. Goodfellow *et al.*, “Generative adversarial networks,” *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[247] A. Creswell and A. A. Bharath, "Inverting the generator of a generative adversarial network," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 7, pp. 1967–1974, 2018.

[248] 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*, 2019, pp. 4432–4441.

[249] Z.-H. Feng, J. Kittler, M. Awais, P. Huber, and X.-J. Wu, "Wing loss for robust facial landmark localisation with convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2235–2245.

[250] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*, Springer, 2016, pp. 694–711.

[251] L. Gatys, A. S. Ecker, and M. Bethge, "Texture synthesis using convolutional neural networks," *Advances in neural information processing systems*, vol. 28, 2015.

[252] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *arXiv preprint arXiv:1508.06576*, 2015.

[253] S. Liu and W. Deng, "Very deep convolutional neural network based image classification using small training sample size," in *2015 3rd IAPR Asian conference on pattern recognition (ACPR)*, IEEE, 2015, pp. 730–734.

[254] R. Abdal, Y. Qin, and P. Wonka, "Image2stylegan++: How to edit the embedded images?" In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8296–8305.

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

[256] D. E. King, “Dlib-ml: A machine learning toolkit,” *The Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[257] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[258] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, “Learning transferable architectures for scalable image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8697–8710.

[259] 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*, 2019, pp. 4690–4699.

[260] 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.

[261] N. Damer, C. A. F. López, M. Fang, N. Spiller, M. V. Pham, and F. Boutros, “Privacy-friendly synthetic data for the development of face morphing attack detectors,” *arXiv preprint arXiv:2203.06691*, 2022.

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

[263] L. Wolf, T. Hassner, and I. Maoz, “Face recognition in unconstrained videos with matched background similarity,” in *CVPR 2011*, IEEE, 2011, pp. 529–534.

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

[265] T. Ojala, M. Pietikäinen, and D. Harwood, “A comparative study of texture measures with classification based on featured distributions,” *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[266] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[267] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[268] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255.

[269] U. Scherhag, C. Rathgeb, J. Merkle, and C. Busch, “Deep face representations for differential morphing attack detection,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3625–3639, 2020.

[270] A. C. Popescu and H. Farid, “Statistical tools for digital forensics,” in *international workshop on information hiding*, Springer, 2004, pp. 128–147.

[271] J. Fridrich, “Digital image forensics,” *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 26–37, 2009.

[272] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, “Fully-convolutional siamese networks for object tracking,” in *European conference on computer vision*, Springer, 2016, pp. 850–865.

[273] J. Kittler, M. Hatef, R. P. Duin, and J. Matas, “On combining classifiers,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[274] A. Makrushin, C. Kraetzer, J. Dittmann, C. Seibold, A. Hilsmann, and P. Eisert, "Dempster-shafer theory for fusing face morphing detectors," in *2019 27th European Signal Processing Conference (EUSIPCO)*, IEEE, 2019, pp. 1–5.

[275] Z. Gao, Y. Wu, X. Zhang, J. Dai, Y. Jia, and M. Harandi, "Revisiting bilinear pooling: A coding perspective," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 3954–3961.

[276] 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.

[277] A. Santoro *et al.*, "Relational recurrent neural networks," *arXiv preprint arXiv:1806.01822*, 2018.

[278] Y. Chen, Z. Liu, H. Xu, T. Darrell, and X. Wang, "Meta-baseline: Exploring simple meta-learning for few-shot learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9062–9071.

[279] M. Norouzi *et al.*, "Zero-shot learning by convex combination of semantic embeddings," *arXiv preprint arXiv:1312.5650*, 2013.

[280] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[281] L. R. Qualls and B. A. Corbett, "Examining the relationship between social communication on the ados and real-world reciprocal social communication in children with asd," *Research in autism spectrum disorders*, vol. 33, pp. 1–9, 2017.

[282] J. R. Pruette, "Autism diagnostic observation schedule-2 (ados-2)," *Google Scholar*, 2013.

[283] V. Hus and C. Lord, "The autism diagnostic observation schedule, module 4: Revised algorithm and standardized severity scores," *Journal of autism and developmental disorders*, vol. 44, no. 8, pp. 1996–2012, 2014.

[284] 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.

[285] B. Jiang, M. F. Valstar, and M. Pantic, "Action unit detection using sparse appearance descriptors in space-time video volumes," in *Face and Gesture 2011*, IEEE, 2011, pp. 314–321.

[286] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent k-svd," in *CVPR 2011*, IEEE, 2011, pp. 1697–1704.

[287] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[288] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 1, pp. 40–51, 2007.

[289] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM review*, vol. 51, no. 1, pp. 34–81, 2009.

[290] M. Elad, *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer Science & Business Media, 2010.

[291] R. Rubinstein, T. Faktor, and M. Elad, "K-svd dictionary-learning for the analysis sparse model," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2012, pp. 5405–5408.

[292] S. Yang, L. Liu, and M. Xu, "Free lunch for few-shot learning: Distribution calibration," *arXiv preprint arXiv:2101.06395*, 2021.

[293] J. W. Tukey *et al.*, *Exploratory data analysis. Addison-Wesley Series in Behavioral Science.* Reading, Mass. URL https://cds.cern.ch/record/107005., 1977, vol. 2.

[294] H.-Y. Wu, M. Rubinstein, E. Shih, J. Guttag, F. Durand, and W. Freeman, “Eulerian video magnification for revealing subtle changes in the world,” *ACM transactions on graphics (TOG)*, vol. 31, no. 4, pp. 1–8, 2012.

[295] S.-T. Liong, Y. S. Gan, J. See, H.-Q. Khor, and Y.-C. Huang, “Shallow triple stream three-dimensional cnn (ststnet) for micro-expression recognition,” in *2019 14th IEEE international conference on automatic face and gesture recognition (FG 2019)*, IEEE, 2019, pp. 1–5.

[296] H.-Q. Khor, J. See, S.-T. Liong, R. C. Phan, and W. Lin, “Dual-stream shallow networks for facial micro-expression recognition,” in *2019 IEEE international conference on image processing (ICIP)*, IEEE, 2019, pp. 36–40.

[297] J. S. Pérez, E. Meinhardt-Llopis, and G. Facciolo, *Tv-l1 optical flow estimation. image processing on line, 2013: 137–150, 2013*, 2013.

[298] H. Pan, L. Xie, Z. Wang, B. Liu, M. Yang, and J. Tao, “Review of micro-expression spotting and recognition in video sequences,” *Virtual Reality & Intelligent Hardware*, vol. 3, no. 1, pp. 1–17, 2021.

[299] G.-B. Liong, J. See, and L.-K. Wong, “Shallow optical flow three-stream cnn for macro-and micro-expression spotting from long videos,” in *2021 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2021, pp. 2643–2647.

[300] X.-B. Nguyen, C. N. Duong, L. Xin, G. Susan, S. Han-Seok, and K. Luu, “Micronbert: Bert-based facial micro-expression recognition,” in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[301] F. Qu, S.-J. Wang, W.-J. Yan, H. Li, S. Wu, and X. Fu, “Cas (me) ˆ2: A database for spontaneous macro-expression and micro-expression spotting and recognition,” *IEEE Transactions on Affective Computing*, vol. 9, no. 4, pp. 424–436, 2017.

[302] J. Li *et al.*, "Cas (me) 3: A third generation facial spontaneous micro-expression database with depth information and high ecological validity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[303] W.-J. Yan *et al.*, "Casme ii: An improved spontaneous micro-expression database and the baseline evaluation," *PloS one*, vol. 9, no. 1, e86041, 2014.

[304] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[305] IDEMIA, "Stereo laser image," 2022.

[306] A. A.Deeb, "Uae reviews features of new id card, 3d photo included," 2022.

[307] J. W. J. ter Hennepe, "3d photo id," 2022.

[308] F. B. A. Systems, "3d face enrolment for id cards," 2022.

[309] ICAO, "Machine readable travel documents. part 11: Security mechanisms for mrtds. technical report doc 9303," *technical report doc 9303*, 2021.

[310] I. J. S. Biometrics, "Iso/iec 39794-5:2019 information technology — extensible biometric data interchange formats — part 5: Face image data," 2019.

[311] 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)*, IEEE, 2017, pp. 555–563.

[312] 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*, Springer, 2020, pp. 255–265.

[313] 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)*, IEEE, 2019, pp. 1–8.

[314] P. Kellnhofer, A. Recasens, S. Stent, W. Matusik, and A. Torralba, "Gaze360: Physically unconstrained gaze estimation in the wild," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6912–6921.