PERSONALITY-AWARE DECEPTION DETECTION FROM BEHAVIORAL CUES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

By
Burak Mandıra
September 2021

Personality-Aware Deception Detection from Behavioral Cues
By Burak Mandıra
September 2021

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Hamdi Dibeklioğlu(Advisor)

Shervin Rahimzadeh Arashloo

Albert Ali Salah

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

PERSONALITY-AWARE DECEPTION DETECTION FROM BEHAVIORAL CUES

Burak Mandıra
M.S. in Computer Engineering
Advisor: Hamdi Dibeklioğlu
September 2021

We encounter with deceptive behavior in our daily lives, almost everyday. Even the most reliable ones among us can sometimes be deceptive either deliberately or unintentionally. Since people are not successful at detecting lies most of the time, it is necessary to use an automated deception detection system particularly in high-stake scenarios such as court trials.

We propose a fully automated personality-aware deception detection model that uses videos as input. To our knowledge, we are the first to consider analyzing the personality of subjects in a deception detection task. The proposed model is a multimodal approach where it uses both facial expression and voice related cues in addition to personality traits of subjects in its analyses. After personality traits are extracted, they are combined with deception features which are based on expression cues. Deception, voice, and personality modules are constituted from the spatiotemporal architectures such as 3D-ResNext and CNN-GRU to better comprehend the temporal dynamics of the input. Finally, it combines expression and voice modalities using a GRU based fusion model.

The evaluation of the proposed model is performed on Real Life Trials dataset that uses the records of court trials from real life. The results suggest that the use of personality traits facilitates the deception detection task. When the personality features are employed in addition to deception features, there is up to 20.4% (relative) improvement on the performance of the deception module. When the voice related cues are also considered upon that, we obtain 15.4% (relative) improvement additionally.

Keywords: deception detection, personality analysis, multimodal fusion.

ÖZET

DAVRANIŞSAL ÖZELLİKLERDEN KİŞİLİK DUYARLI ALDATMA ALGILAMA

Burak Mandıra
Bilgisayar Mühendisliği, Yüksek Lisans
Tez Danışmanı: Hamdi Dibeklioğlu
Eylül 2021

Günlük hayatımızda neredeyse her gün aldatıcı davranışlarla karşılaşıyoruz. Aramızdaki en güvenilir olanlar bile bazen kasıtlı veya kasıtsız olarak aldatıcı olabilir. İnsanlar çoğu zaman yalanları tespit etmekte başarılı olamadığı için, özellikle mahkeme davaları gibi yüksek riskli senaryolarda otomatik bir aldatma tespit sisteminin kullanılması gerekmektedir.

Videoları girdi olarak kullanan, tam otomatik, kişiliğe duyarlı bir aldatma algılama modeli öneriyoruz. Bildiğimiz kadarıyla, bir aldatma tespit görevinde deneklerin kişiliğini analiz etmeyi ilk düşünen biziz. Önerilen model, analizlerinde deneklerin kişilik özelliklerine ek olarak hem yüz ifadesini hem de sesle ilgili ipuçlarını kullanan çok modlu bir yaklaşımdır. Kişilik özellikleri çıkarıldıktan sonra ifade ipuçlarına dayalı aldatma öznitelikleri ile birleştirilir. Aldatma, ses ve kişilik modülleri, girdinin zamansal dinamiklerini daha iyi anlamak için 3D-ResNext ve CNN-GRU gibi uzaysal-zamansal mimarilerden oluşturulmuştur. Son olarak, GRU tabanlı bir füzyon modeli kullanarak ifade ve ses kiplerini birleştirir.

Önerilen sistemin değerlendirmesi, gerçek hayattan alınmış mahkeme duruşmalarının kayıtlarını kullanan Real Life Trials veri seti üzerinde gerçekleştirilmiştir. Sonuçlar, kişilik özniteliklerinin kullanımının aldatma tespiti görevini kolaylaştırdığını göstermektedir. Aldatma özniteliklerine ek olarak kişilik öznitelikleri de kullanıldığında, aldatma modülünün performansında %20,4'e kadar (göreli) iyileşme gözlemlenmiştir. Bunun üzerine sesle ilgili ipuçları da dikkate alındığında ek olarak %15,4 (göreli) iyileşme elde ediyoruz.

Anahtar sözcükler: aldatma algılama, kişilik analizi, çok kipli tümleştirme.

Acknowledgement

First and foremost, I would like to express my deepest appreciation to my thesis supervisor Asst. Prof. Dr. Hamdi Dibeklioğlu for his profound guidance, great patience and endless support throughout my M.Sc. studies, with my deepest gratitude.

I would also like to extend my gratitude to my father Muzaffer, my mother Habibe and my brother Kamil, who never wavered in their support and always motivated me.

I am also grateful to know outstanding people within our M.Sc. group, my sincere friends Diyala, Dersu and Oğuzhan.

My sincere thanks to Uçuş Kulübü and so many people that I have met in paragliding who gave me invaluable support through their companionship during my struggle.

Lastly, I would like to express my gratitude to TÜBİTAK for their support via National Scholarship Programme for M.Sc. Students - 2210.

"Si, comme la vérité, le mensonge n'avoit qu'un visage, nous serions en meilleurs termes. Car nous prendrions pour certain l'opposé de ce que diroit le menteur. Mais le revers de la verité a cent mille figures et un champ indéfini.

. . .

Mille routtes desvoient du blanc, une y va."

"If falsehood had, like truth, but one face only, we should be upon better terms; for we should then take for certain the contrary to what the liar says: but the reverse of truth has a hundred thousand forms, and a field indefinite.

. . .

There are a thousand ways to miss the white, only one to hit it."

Michel de Montaigne
Of Liars, The Essays

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Deception can be defined as a way of intentional acting that performed for causing people to believe what is false or invalid as true or valid. We observe deceptive behavior even in a single day. Even the most reliable ones among us can sometimes be deceptive in various ways either deliberately or unintentionally due to having poor memory. There are mainly two ways to lie: either via falsification or concealment [1]. Falsification is more complex than concealment as the liar also have to come up with a false statement as if it were true. It's overt and intentionally made. Exaggeration or understatements can be though as falsification. Examples of concealment, on the other hand, are using evasive and ambiguous language to hide one or more important aspects of a real subject. According to [2], liars prefer to conceal than to falsify when they lie. The reason is obvious since concealment is more advantageous: it does not require any extra statements that needs to be remembered later on in order not to be caught. For that reason, people with poor memory prefer concealment. Also, concealment can be seen morally less reprehensible as it's passive rather than being active. For instance, according to [3], liars feel less unrighteous when they lie indirectly (via concealing the truth) and the person who have been told a lie feels the opposite: such a lie is more indecent and vulgar than falsification. We almost always perceive deception as wicked but it does not necessarily to be. Under some strict circumstances, lying can be considered as a required act of maneuver [4]. For instance, telling lies

in wars can be seen as one of the strategic tactics and it can provide to have the edge over the enemy. Renouncing deception in that case is against one's own nation's interest. Similarly, according to Niccolò Machiavelli, manipulative, amoral and deceptive behavior is a necessary tool that make people achieve power and success [5].

Unfortunately, people are not successful detecting deceptive behavior most of the time. [6] contemplates the results of approximately 40 studies and concludes that people are 67% successful for detecting truths whereas 44% successful for detecting lies, with an average of 54% correct lie-truth judgements [7]. The reason why we are good at detecting truth is because of truth-default theory which states that people will default to be truthful in their communications and are inclined to see others as honest [8, 9]. There is also a cognitive bias called *halo effect* [10], or sometimes called *halo error*, in psychology which states that perception of people about a person, brand is heavily influenced by that particular person's other traits. For instance, people assign positive qualities such as having higher morality, better mental health, and greater intelligence to physically attractive individuals. A real life example to this error can be Bernard Madoff, who was a securities fraudster and ran the largest Ponzi scheme in history worth $64.8 billion. He often finds easy victims by appearing trustworthy, empathetic, and displaying convincing emotional masks [11]. This constant error in judgement stems from the people's prejudices, ideology, and social perception [12, 13, 14]. These reasons necessitate the need of deception detection methods especially for high-stake cases such as police interrogations [15, 16] and court trials.

According to the literature [2, 11, 17], there are some voluntary and involuntary, relatively short lasting emotional responses of human face which are called *micro-expressions* and leak the true emotional state of an individual for less than one-quarter of a second despite his or her attempt to conceal it. The study of [2] is based on the work of Charles Darwin who hypothesized that some facial muscle movements cannot be inhibited especially under the influence of strong emotions. In his third book, *The Expression of the Emotions in Man and Animals* [18], Darwin states that:

"A man when moderately angry, or even when enraged, may command the movements of his body, but ... those muscles of the face which are least obedient to the will, will sometimes alone betray a slight and passing emotion."

In their experimental analyses on emotional leakage in deceptive facial expressions, [11] find valuable supports for Darwin's hypothesis and states that high intensity emotions were more difficult to conceal than low intensity emotions during emotional neutralization. Hence, automated face focused deception detection models may rely on these micro-expressions for the detection of deceit. Recent studies use visual, acoustic, textual features, however, we only focus on visual face focused and acoustic features in this thesis.

1.1 Motivation

Since it is difficult for people to differentiate deceptive behavior, it's necessary to have a fully automated deceit detection system. There can be more than one way to detect deceits, however, it is necessary to achieve such an act in a covert way for many reasons. Firstly, when it is non-intrusive, it can be applied in real life scenarios without requiring the acquiescence of subjects. Another virtue of being covert is that the suspect is not aware of the deception detection process and he or she cannot prepare himself or herself to cheat the detection system as in the case of polygraph or functional magnetic resonance imaging (fMRI) [2, 19, 20, 21].

There are many studies for detecting deceit either fully automated or semi-automated, which shall be given in Section 1.2. However, to the best of our knowledge, there is no study in the literature that analyses the effect of personality in deception detection. Thus, in this thesis, we propose to integrate personality of subjects in our fully automated multimodal deception detection system which focuses on (facial) expression and voice related features.

1.2 Deception Detection

There may be a lot of cues to deception and their categorization can be complex. For instance, [22] categorizes them as leakage cues and deception cues, where the former discloses the truth that is being withheld and the latter just indicates the occurrence of deception without revealing the concealed information. Current studies in the literature are almost always based on the deception cues and cues are categorized as verbal and nonverbal [8, 6, 23, 24]. Examples of verbal cues are paraverbal characteristics such as voice, tone, pitch and linguistic content whereas examples of nonverbal cues are facial expressions, facial action units (AUs), gaze, posture, body language, other parts of body such as fingers, legs, feet. Different studies emphasis the importance of both cues. Sigmund Freud, for instance, emphasis the importance of nonverbal cues in [25] by saying:

> *"He that has eyes to see and ears to hear may convince himself that no mortal can keep a secret. If his lips are silent, he chatters with his fingertips; betrayal oozes out of him at every pore."*

Nonetheless, analyses about observers' performance on detecting deception on verbal and nonverbal cues in [7, 26] indicate that observers perform worse with nonverbal cues than with verbal cues and they perform best when they have access to both cues. In other words, people are more accurate in auditory than visual deception detection.

Categorization of deception, on the other hand, is mostly based on their severity, e.g. high-stake vs. low-stake deception [23, 24, 27, 28, 29]. In high-stake deception, the outcome of deceiving has utmost consequences for the deceiver whereas the act of low-stake deception has of little consequences. Early studies in the literature focus on detecting the low-sake deceptions whereas recent ones focus more on high-stake deceptions.

A prelude to automated deception detection starts with rule-based approaches via decision trees based on sentence and word lengths, duration of hesitation

pauses, speech act profiling etc. on transcribed speech or texts [30, 31, 32]. Later, analyses on images and videos emerge. [33] extracts hand and face related nonverbal features such as hand-to-hand and hand-to-head distances via blob analysis in a controlled environment and uses support vector machine (SVM), multilayer perceptron (MLP), and decision tree (DT) for the classification. Similarly, [34] applies blob analysis on head and hands to extract features and uses them to calculate subjects' duration in predefined behavioral states to decide whether the subject is lying. [35] uses blob analysis to extract features and uses Gausssian Mixture Models (GMM) to cluster patterns and train Hidden Markov Model (HMM) to detect deception. [36] focuses solely on eye-based features such as closure, blinking and lateral movements of the iris. Nonetheless, [37] shows that alleged notion of certain eye movements being reliable indicators of deception is a common misbelief. All of these approaches are invasive in the sense that subjects must be queried in a controlled environment. Therefore, they are not applicable to real-life cases. As a remedy, [38, 39, 40] propose to use thermal facial screening to detect lies based on a physiological phenomenon, called "fight or flight response", under threat. [41] proposes a novel approach that uses distance and texture based features based on Facial Action Coding Systems (FACS) by manual face detection and facial landmark labeling. FACS is proposed in [42] and uses combination of AUs to represent facial expressions based upon the musculature of the face. [43] design a new set of features called "motion profiles" that are constructed based on subjects' head and hand movements as well as their facial expressions to differentiate simulated and involuntary facial expressions. They employ both SVM and nearest neighbour classifier. Even though their facial landmark and micro expression detection is automated, their head and hands blob analysis relies on an offline look-up-table comprised of hue and saturation color components based on the skin color distribution of face and hands. On the contrary to other works, [44] examines the dyadic synchrony such as rapport and coordination between the deceiver and the deceived as a measure of deception detection. [45, 46, 47, 48] emphasize the use of whole body motion in deception detection. [46, 48] utilize motion capture suits to record the 3D motions of 23 body joints and they calculate angle, distance, movement and symmetry between those body joint points as features to train simple classifiers like Gaussian Naive

Bayes classifier or logistic regression. [46] achieves 5-8% higher scores compared to a single body part features in their analyses.

[49] focuses on verbal cues of deception in the transcribed text of Italian high-stake court cases. Their method utilizes lemmatization, part-of-speech tagging, n-grams and other rule-based features such as LIWC based on letters, utterances and words. In their thorough analysis, interesting findings are discovered but their overall performance is not significantly better than chance and suffers from precision vs. recall trade-off. Similarly, [50] uses unigrams and LIWC features in cross-cultural dataset composed of English and Spanish from three different countries and finds out that cross-cultural datasets can be used together to boost the performance of deception detection models. [51] follows a multimodal approach where physiological, linguistic and thermal features are used altogether for the first time and trains a DT classifier on the concatenated features. As for the limitations of above methods, being invasive (i.e., querying subjects in a controlled environment), having some strict assumptions such as a stationary background with the only animate object being the subject, or the limited topic coverage of linguistic features (which may not be enough to generalize) might be counted.

[26] is an important milestone in deception detection where authors publicly announce a novel dataset consisting of videos collected from high-stake, real-life, public court trials. The video clips are transcribed via crowdsourcing using Amazon Mechanical Turk. They use linguistic and manually annotated visual (facial, head and hand related) features to train DT and random forest (RF) classifiers. This dataset is widely used by subsequent studies and it became the benchmark dataset for the deception detection. [52] uses linguistic, visual and acoustic features on this dataset and proposes a fully automated deception detection model. Similarly, [53] proposes an open-source multimodal feature extraction tool (OpenMM) and evaluates its performance on the same dataset with DT and RF classifiers. [54] propose another multimodal approach on the same dataset using visual (IDT), audio (MFCC) and transcript (Glove) features with Fisher encoding. They predict facial micro-expression features rather than manual annotation using derivative based motion information inherited from IDT features

and fuses them with the multimodal Fisher vectors. Nonetheless, their model requires ground truth labels per frame for micro-expression classifier training, which necessitate manual annotation. [55] proposes both semi-automated and fully automated deception detection models using visual, acoustic and textual features at the subject level where authors merge the features of same subjects via averaging to obtain a single feature per subject and uses RF, SVM and neural network (NN) classifiers on these per-subject features. [56] present a new deception detection approach based solely on hand gestures where subjects' hand skeleton joint positions in video frames are extracted and encoded by the Fisher vectors. They feed the final feature vectors to a Long-Short Term Memory (LSTM) network for the classification of videos.

1.3 Personality and Deceptive Behavior

Personality is often related to traits and traits can be defined as a set of patterns of behavior, thought and emotion that make a person unique. People with certain traits have great ability to deceive others under some circumstances. For instance, Paul Ekman states in [2] that,

> *"Not everybody is likely to feel afraid, guilty, angry, and so on when they know they are suspected of wrongdoing or lying. It depends in part upon the personality of the suspect."*

To categorize various personalities that people have, different personality trait models are developed in the psychology literature. The Sixteen Personality Factor (16PF) [57], Big Five [58] and HEXACO [59] are examples to them which tries to assess people's personality from an observer's point of view. Big Five model is one of the well-established ones and it consists of five independent traits. These are Extraversion, Agreeableness, Openness to experience, Conscientiousness and Neuroticism, which are mostly abbreviated and referred to as OCEAN. Table 1.1 provides some exemplars to the people having high scores on these traits. According to [58], the model is comprehensive, efficient and diverse in the sense that

Table 1.1: Example adjectives that describe the Big Five personality traits

Trait	**Adjectives**
Openness to experience	Adventurous, Curious, Imaginative, Insightful, Inventive, Deep, Intellectual
Conscientiousness	Careful, Diligent, Efficient, Thorough, Organized, Reliable, Self-disciplined, Dutiful
Extraversion	Energic, Outgoing, Talkative, Enthusiastic, Gregarious, Assertive, Social
Agreeableness	Friendly, Kind, Cooperative, Sympathetic, Altruistic, Appreciative, Forgiving, Generous
Neuroticism	Moody, Anxious, Self-pitying, Tense, Frustrated, Unstable, Worrying

it includes variety of personalities. The model is comprehensive in the sense that five dimension is enough to represent the links between personalities and that it can be used by various types of psychologists such educational, industrial, and clinical in many applied settings. Efficiency of the model is established by using as few as five dimensions to thoroughly describe all kind of personalities.

Recent studies suggest a link between personality and deceptive behavior [60, 61, 62, 63, 64, 65]. In [60], dominance, exhibition, and emotional sending ability were found to be highly positively correlated with successful deception ability under the availability of visual and sound of subjects. Although the correlation of personality and deceptive behavior are found to be sex-agnostic in [60], emotional sending ability depends on the sex due to nature of masculinity-femininity in nonverbal social skills [66, 67]. The positive correlation of emotional sending ability and truthfulness of subjects are found to be significant for males whereas it is near zero for females. In other words, subjects who have the ability to enact basic emotions are perceived more truthfully and it is particularly true for males.

In [63], authors show that Dark Triad personality traits, which are originally developed in [68], have high correlation with the perceived ability to deceive.

Machiavellianism (characterized by manipulation, exploitation of others, deceitful tendencies, high level of self-interest), narcissism (characterized by pride, grandiosity, egotism, lack of empathy, excessive self-love) and psychopathy (characterized by persistent impulsive and antisocial behavior, impaired empathy and remorse) forms the Dark Triad and they share a callous-manipulative interpersonal style. Among these traits, Machiavellianism is found to be the most highly correlated one, followed by psychopathy and narcissism with the perceived ability to deceive. Furthermore, they also investigate the relation between Dark Triad traits and Big Five [58] personality traits and found that narcissism was positively correlated with Extraversion whereas Machiavellianism and psychopathy were negatively correlated with Agreeableness and Conscientiousness. Their overall findings suggest that Agreeableness and Conscientiousness are significantly negatively correlated, and that Extraversion has a weak but significant positive association with the perceived ability to deceit. Furthermore, [64] states that there are individual differences in deceptive universal emotional expressions and that psychopathic traits exhibit less leakage of inconsistent emotions during deceptive displays. Similarly in [65], authors state that lying is part of personality and the result of their empirical research suggest that intelligent extraverts and people who have less agreeableness score are most likely to lie. This suggests that deception is not only affected by personality traits but also by intelligence.

Based on these findings, we include personality analysis of subjects in our deception detection system. To the best of our knowledge, we are the first to incorporate personality of subjects into a deception detection task, with a fully automated model.

1.4 Outline of the Thesis

The outline of the rest of the thesis is as follows: In Chapter 2, the proposed personality-aware deception detection system is presented in detail. Chapter 3 starts with providing some insights of the dataset used. Then, it continues with experimental setup, our final results and interpretations, comparison to other

studies in the literature followed by qualitative analysis. Finally, the thesis is concluded with a summary of the contributions, a brief outline of our final findings, implications of deception detection models and possible misuse, and plausible improvements to the system for the future work.

Chapter 2

Method

Our method is a multimodal analysis of videos using visual and audio inputs. We also integrate personality of subjects in the videos using visual analysis. The following sections describe the details of these analyses.

2.1 Overview

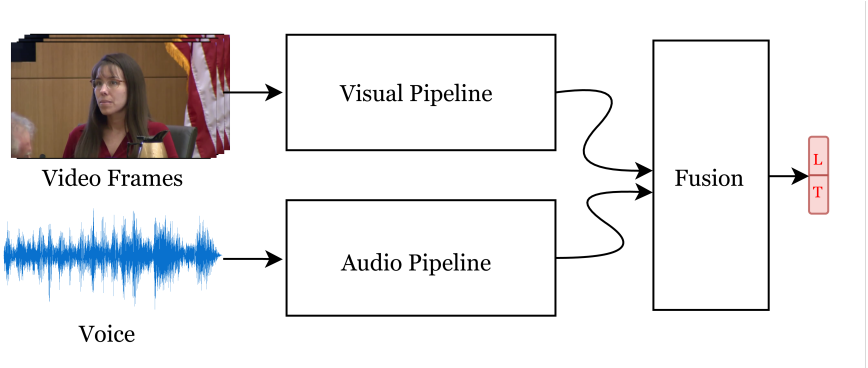


Figure 2.1: Overall pipeline of the proposed system. Solid red box represents the truth and lie scores.

The proposed system in this thesis is a fully automated multimodal deception detection system which is coarsely illustrated in Figure 2.1. It contains both

visual and audio modalities. The visual pipeline is depicted in Figure 2.2. As it can be seen, the visual model only focuses to facial boundary of the subjects.

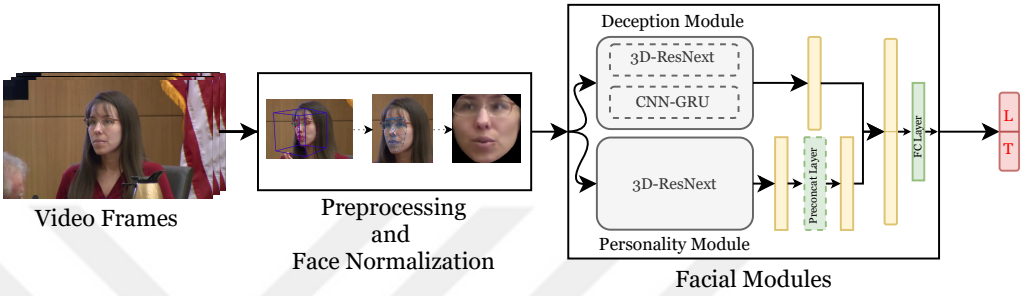


Figure 2.2: Visual pipeline of the proposed system. Dashed gray boxes represent the models that only one of them can be chosen. Dashed green box represents an optional layer. Solid light yellow boxes represent the extracted features. Finally, solid red box represents the truth and lie scores.

The visual pipeline is as follows: First, raw video frames are extracted and then passed to the preprocessing and face normalization step. In this step, facial landmarks are detected using OpenFace [69]. After obtaining facial landmarks for each frame, the average of facial landmarks are calculated using the all frames exist in all videos. Then, each facial frame is triangulated where the facial landmarks are being the corners using Delaunay method, provided from [69]. The goal of triangulation is to apply piece-wise affine transform onto each triangle so that we obtain a normalized facial image at the end. It also masks out only the face regions. Finally, these normalized facial images are rescaled and fed to the (facial) expression modules. Expression modules contain two submodules which work in parallel: deception module and personality module. In deception module, we have two alternative architectures to use: 1. 3D-ResNext [70], 2. The combination of convolutional neural network (CNN) and gate recurrent units (GRU): CNN-GRU. In both models, extraction of facial features from the normalized facial images is made through a deep convolutional neural network. In this submodule, we try to extract valuable features using the normalized facial frames without considering the personality of subjects using only one of the models above (dashed gray boxes in Figure 2.2). Meanwhile, personality module, focuses only on the personality of subjects using, again, the normalized facial

images. The personality module consists of 3D-ResNext model and is based on our work in [71]. The output of the personality module is high dimensional representation of subjects' personality which then, optionally, passed through an MLP that consists of two blocks of batch normalization (BN) layer, 1×1 convolutional layer and ReLU activation function (shown as 'preconcat layer' inside the green dashed box in Figure 2.2). Then, it is concatenated with the deception module's features.

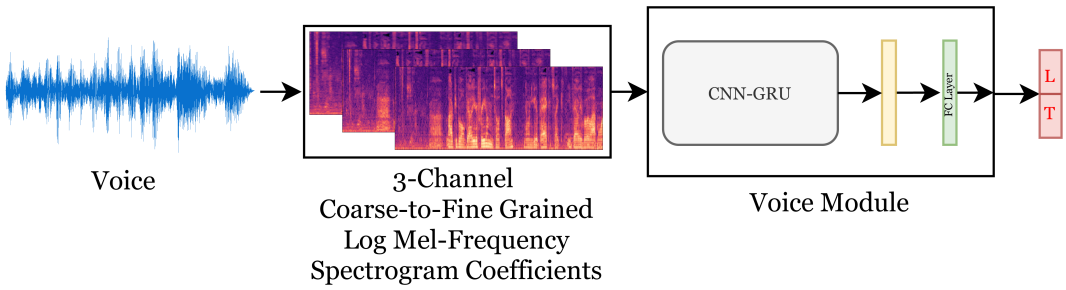


Figure 2.3: Audio pipeline of the proposed system

The first step of the audio pipeline is creating wav files from the videos. For that, ffmpeg is employed with default parameters. After that, feature extraction is performed using the voice in videos. With three different window lengths, log mel-frequency spectrogram coefficients are extracted with librosa [72]. Then, these three melspecrogram features are smartly concatenated to form a single 3-channelled coarse-to-fine grained voice feature per record. Then, these features are fed to the voice module which consists of CNN-GRU model. Finally our system combines expression and voice modules in the fusion module which is illustrated in Figure 2.9. Fusion module comprises a a bi-directional gated recurrent units (BiGRU) based recurrent neural network. Ultimately, our motivation for using a fusion module on top of each modality is to improve the performance of unimodals. The details of each of these modules and steps will be given in the following sections.

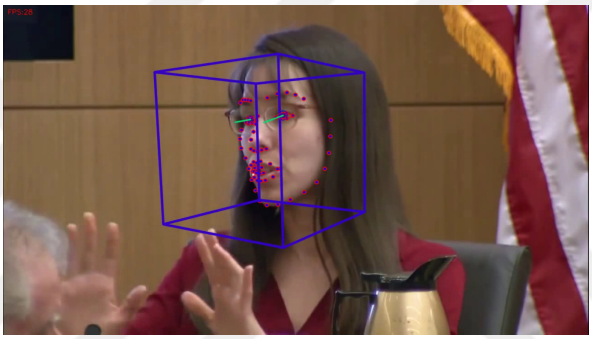
2.2 Expression Analysis

Our approach for (facial) expression analysis first start with a preprocessing step, namely face normalization, where facial frame images are normalized to ease the interpretation of CNN-based models. Once the normalization is completed, we carry on with applying our models onto these normalized facial frames. Details are provided in the following sections.

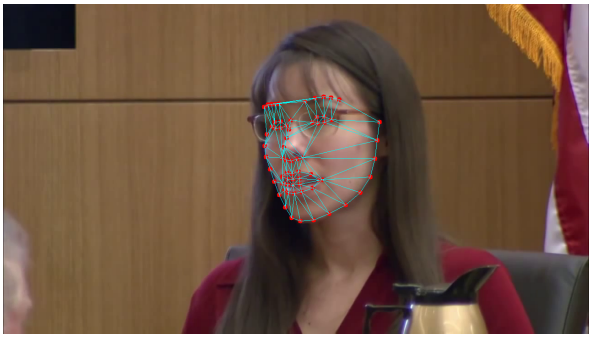
2.2.1 Preprocessing and Face Normalization

Before conducting visual analysis, we apply normalization on facial frames. We track 68 landmarks on the facial boundary, eyes eyebrows, nose, and mouth in the videos using a state-of-the-art tracker, namely OpenFace [69]. Then, we calculate the average pixel coordinates of these landmarks considering all frames in the dataset. Thereafter, using the OpenFace's facial triangulation matrix (which uses Delaunay method), we construct 112 triangles. Each of these small triangle patches are warped so as to transform the X and Y coordinates of the detected landmarks onto those of average normalized landmarks to normalize the facial image in each frame. This process includes removing the global rigid transformations such as translation, rotation and scale. At the end, facial frames are normalized in terms of translation, rotation and scale; and frontal view of the faces are produced. Obtained images are then scaled.

(a)



(b)



(c)

Figure 2.4: (a) Raw frame, (b) Visualization of the facial landmarks, gaze direction, and head pose obtained from OpenFace [69], and (c) Triangulation of the facial landmarks

As a result, each normalized face has a resolution of 512×512 pixels. This procedure is followed only once and then the resulting normalized facial images are used throughout the experiments. Examples of normalized facial images are shown in Fig. 2.5. Note that the deformations in the facial surface can better be interpreted since the normalized faces are directly comparable in a pixel-to-pixel

manner. In other words, each normalized facial image has facial landmarks such as tip of nose, centers of eyes and lips at the fixed locations in terms of pixels independent of the subject's facial anatomy.

Figure 2.5: Normalized face samples

2.2.2 Modeling

2.2.2.1 Deception Module

In order to conduct deception analysis, we utilize two different spatio-temporal models, namely 3D-ResNext-101 [70] and CNN-GRU. They are depicted as dashed gray boxes in Figure 2.2. Both of these models can model the spatial and temporal relations within the frames thanks to either via temporal (3D) convolutions or spatial (2D) convolutions along with the recurrent network.

ResNeXt [73] introduces a new concept in convolution layers which is called the *cardinality*. It is a different dimension from depth and width, and used in each ResNeXt block as group convolutions. The number of group convolutions are called *cardinality*, which divide the feature maps into small groups different from the original ResNet [74] bottleneck block. [73] shows that increasing the cardinality of 2D architectures is more effective than using wider or deeper architectures. A comparison of each architecture's blocks are shown in Fig. 2.6 and Fig. 2.7.

We use the 3D ResNext-101 [70] pretrained on the Kinetics dataset [75]. Even

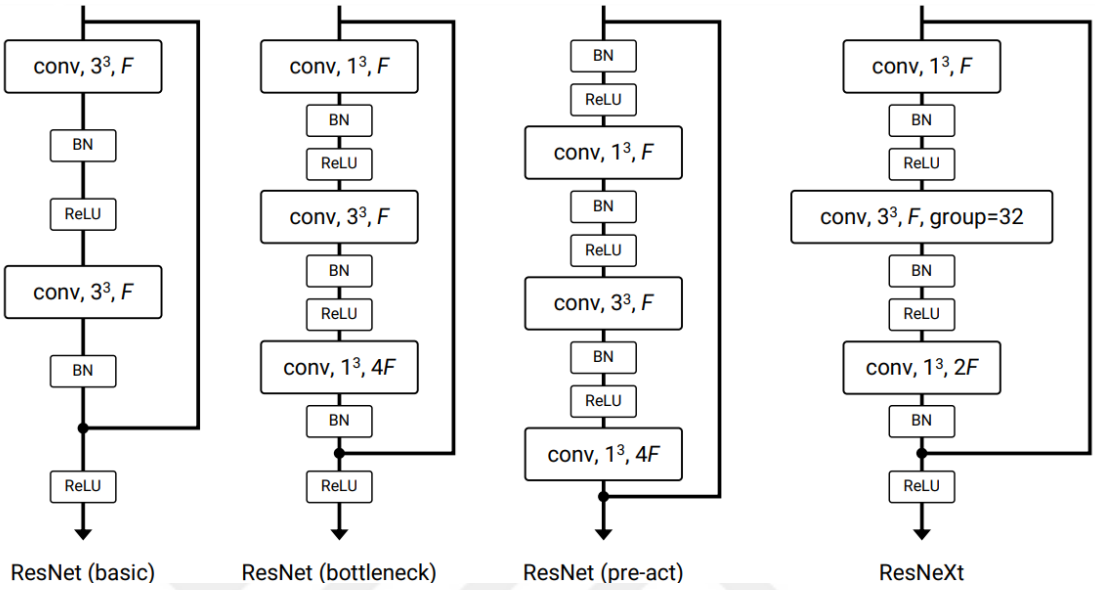


Figure 2.6: Various ResNet blocks and ResNext block from [70]. $conv, x^3, F$ represents a convolution layer with kernel size $x \times x \times x$ and the number of feature maps F. *group* is the number of groups of group convolutions. Finally, BN refers to batch normalization and shortcuts are connected via summation operation in both architectures.

though Kinetics is an activity recognition dataset (which seems irrelevant to our task), we use the pretrained model due to the scarcity of data and complexity of the model to prevent a potential overfitting that might occur in from scratch training. Originally, the model has five bottleneck blocks. We apply fine-tuning starting from the third block based on our preliminary experiments. We use random temporal sampling of 30 frames, which corresponds to 1.5 seconds, during training, and non-overlapping sliding window of the same size during inference. Mean scores of these sliding windows are considered as the final score of a video. Window size is chosen among the values [15, 30, 45, 60] through validation error. The last fully connected (FC) layer of the network is replaced with an FC layer having 2 neurons followed by softmax activation function and cross-entropy loss is utilized.

As for our second model, we use CNN-GRU architecture. We opt for 3D-ResNext-101 [70] as our CNN model. Since recurrent models are prone to vanishing/exploding gradients when used for longer temporal sequences, 3D-ResNext

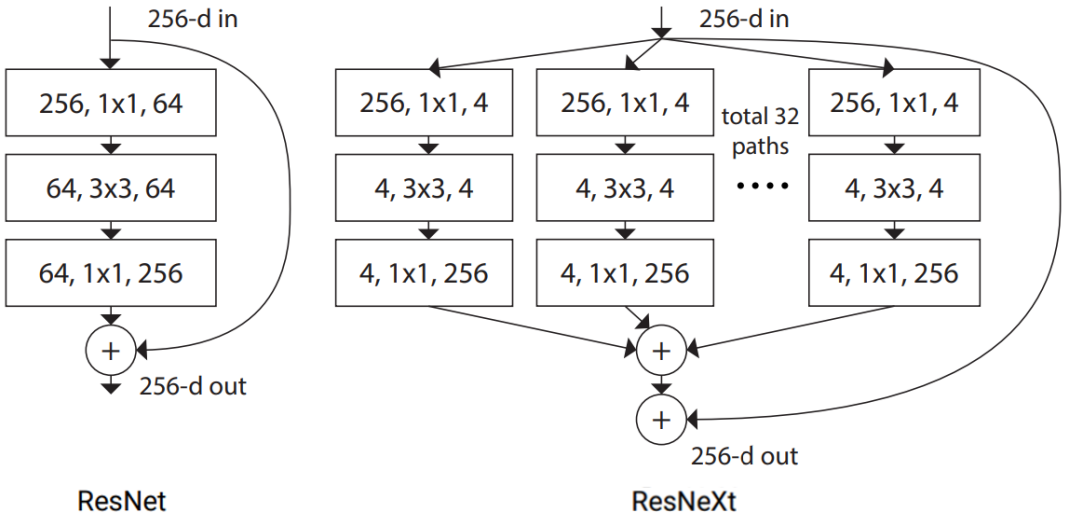


Figure 2.7: A concrete example from [73] for ResNet and ResNext block with 256-dimensional features and cardinality 32. A layer is represented as the number of input feature maps, filter size, and the number of output feature maps, respectively. For the sake of simplicity, *BN* and *ReLU* layers are not shown.

takes care of short-term temporal dynamics while GRU takes care of long-term temporal dynamics, which ease the burden of evaluating temporality solely by the recurrent model. Each video is divided into overlapping 30 frames sized segments and they are passed thorough 3D-ResNext to extract features. Then, the extracted features of each segment are passed to a bidirectional GRU to examine the longer temporal dynamics. At the top of last hidden state, a classification layer is added and the model is trained via cross-entropy loss.

2.2.2.2 Personality Module

There have been numerous studies that try to model personality traits from videos [76, 77, 78, 79, 80, 81, 82], which shows that it's possible to get significant insights about people's personality using their videos despite some recently raised concerns about the potential biases and unfairness of both annotators and algorithmic models [83, 84]. In our personality modeling, we consider subjects' apparent personality in the videos during the deception evaluation. Apparent

personality is subject to a possibly expert observer as opposed to subjects' self evaluation of their own personality. We refer the reader to [82] for a detailed analysis on apparent personality recognition from videos.

In our work [71] upon personality analysis, we develop a multimodal architecture to estimate the Big Five personality traits utilizing the audiovisual cues and transcribed speech. We shown that trait based induced behavior includes the sign of personality traits on two different datasets. Our findings in [71] show that expression based modalities outperform other modalities such as voice, transcribed speech and body pose in both datasets. Furthermore, our multimodal approach achieves on par results with the state of the art model in personality analysis in ChaLearn LAP First Impressions Dataset (FID) [85]. Based on our results in [71], personality traits of subjects in videos are determined using the visual model in [71] which is 3D-ResNext as it performs much better than the other visual as well as acoustic models. We use the weights of the model trained on FID dataset. This dataset consists of more than 10000 video clips collected from more than 3000 YouTube videos, where people speak to the camera in various environments under different conditions. Since it's huge and includes many diverse backgrounds, shooting angles and diversity of ethnicity and culture, it can be considered as having a similar characteristics with RLT dataset in terms of shooting angles, backgrounds and diversity of ethnicity and culture.

The personality module (3D-ResNext in [71]) normally predicts personality scores of the subject in the video in terms of Big Five personality traits. In other words, it's ultimate goal is to regress these scores using the ground truth labels. Since we do not have ground truth Big Five personality scores for the RLT dataset, we only use the model to represent the personality types in high dimensional subspace without training. Given that the model is trained with the videos of FID dataset and this dataset contains videos way more than RLT dataset, with similar characteristics, using the weights of this model *without finetuning* should be sufficient enough to extract personality features of subjects. Nonetheless, it should be noted that if we had person-specific training for personality analysis instead of using a single personality model for every subject, we would most probably attain better representations of each subject's personality. However,

having person-specific personality model requires a large quantity of videos and ground truth annotations per subject (which we have deficiency) in order to train those models properly without overfitting.

2.2.2.3 Personality-Aware Deception Detection

As it can be seen from Figure 2.2, deception module and personality module work in parallel. Extracted personality features are 2048-dimensional tensors. Optionally, we pass them through a 'preconcat layer' as shown in Figure 2.2. The ultimate goal of the preconcat layer is to adaptively learn how to interpret personality features along with the deception features through backpropagation. The structural design of this layer is determined via several experiments where we evaluated various design choices including MLP with custom convolutional and dense layers, convolving the features of deception module along with the personality features before the concatenation, or SCAN [86] like layer. As our final choice, two blocks of BN layer, 1×1 convolutional layer and ReLU activation function is used, where convolutional layers have 1024 and 512 filters, respectively. Thus, after the preconcat layer, the dimension of personality features are reduced to 512. Then, they are concatenated with the deception features coming from either 3D-ResNext or CNN-GRU architectures. In case of 3D-ResNext, the final feature dimension after concatenation becomes 2560 (2048 + 512). On the other hand, it changes for CNN-GRU architecture depending on where we integrate the personality features: either to last temporal unit's hidden representation, in that case it becomes 640 (2 * 64 + 512), or to each temporal unit's input in which case it becomes 128 (only the hidden dimension of bi-directional GRU model).

2.3 Voice Analysis

To elicit the information that voice carries, we employ CNN-GRU architecture as we did for visual analysis. However, we employ 2D-ResNext-101 [73] as our CNN model instead of its spatio-temporal counterpart. Our motivation for CNN-GRU

architecture is that it considers both spatial and temporal relations between audio frames thanks to the spatial convolutions in CNN and temporally recurrent unit in GRU. First, log mel-frequency spectrogram coefficients (which will be called as *melspectrogram* in short, in the rest of the thesis) of audios are computed for each video using windows of size 25, 50 and 100 milliseconds with 75% overlapping. Maximum frequency of audio records are clipped to 8000 Hz and 128 mel filter bands are used. Visualization of each of these features for an example audio file is given in Figure 2.8. As you can see, increasing the window size decreases the granularity (resolution) of the spectrogram, thus altering the representability of the features. Please note that we use the original spectrogram coefficients while forming our voice features rather than these visuals.

Then, each of these single-channeled melspectogram features are concatenated to form coarse-to-fine grained three-channeled melspectogram features. During the concatenation, each element of shorter sequences (e.g. 50 msec. and 100 msec. window sized features) are repeated along the temporal axis to match the longest sequence (e.g. 25 msec. window sized features) where extra elements due to repetition are cropped from the end of the sequence if there is any. While repeating the features, each three temporal features are synchronized in time axis so that each temporal feature of shorter sequences is employed more than once (to be more precise, two times in 25-50 msec. windowed features' synchronization and four times in 25-100 msec. windowed features' synchronization) in concatenation until the current window of the longest sequence overlaps with the next window of shorter sequence.

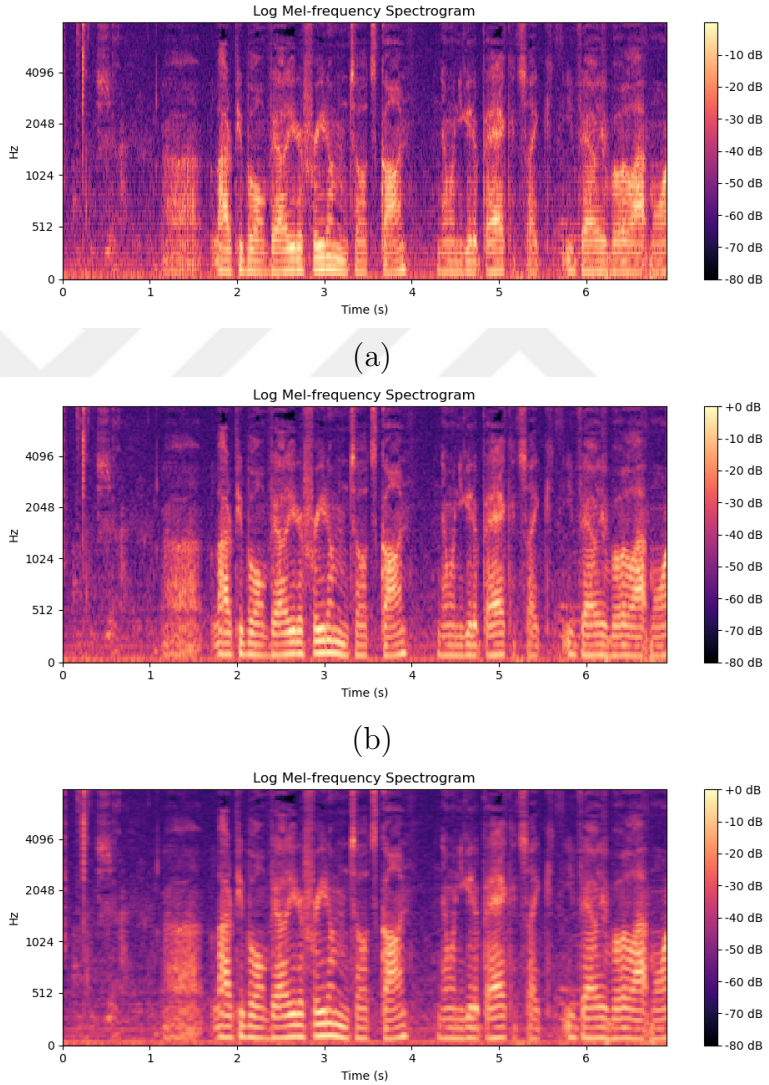


Figure 2.8: Log mel-frequency spectrograms with (a) 25 ms. window size, (b) 50 ms. window size, and (c) 100 ms. window size. All of them are obtained with 75% overlapping windows.

2.4 Fusion

We use GRU architecture for the fusion model as both modalities are already using GRU in their implementations. Fusion GRU's hidden states are initialized using the deception and voice modules' hidden state representations. The input of the fusion model is the attentional fusion of face and voice based CNN features which are normally fed to the respective GRU model while unimodal training. Dimension of facial features are reduced to make it same as voice features before applying the attention module with a linear layer followed by hyperbolic tangent function. Figure 2.9 shows the whole fusion model. CNN and GRU models in visual and audio pipelines are initialized using the pretrained unimodal weights. The whole system is trained via cross-entropy loss in an end-to-end manner except that the CNNs in unimodal pipelines are frozen which is due to the heavy memory consumption of 2D/3D-CNN models during the backpropagation through time (BPTT). To obtain better results in fusion, the centers of facial expression and voice segments are synchronized, i.e. the centers of both segments correspond to the exact same moment t. Zero padding is utilized when the visual segment is out of range for the video (e.g., at the beginning and end of videos).

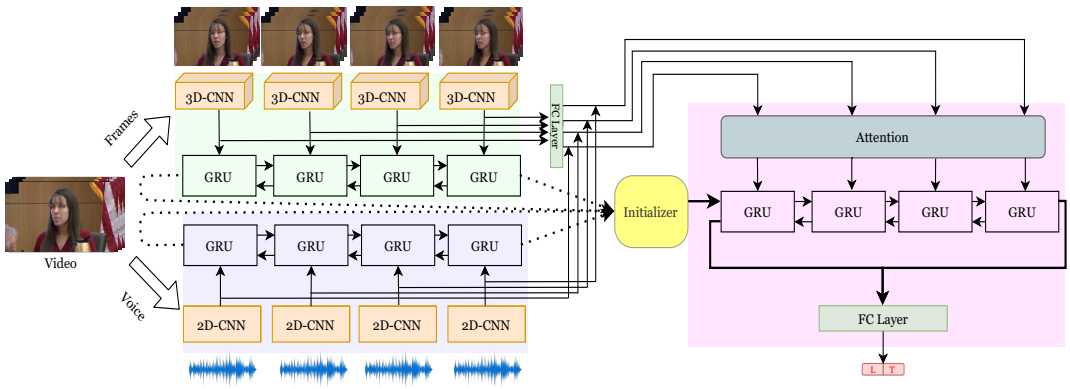


Figure 2.9: The fusion model

We initialize the fusion model via the concatenation of unimodal hidden states as it has been shown to be useful for other temporal modeling tasks such as video captioning, semantic language modeling, and tree topology modeling in NLP tasks [87, 88, 89, 90]. Let h_v and h_a denote the hidden states of facial expression

and voice GRU models at last time step, respectively. Then, the initial hidden state of fusion model h_f is computed as:

$$h_f = tanh(W[h_v \cdot h_a]) \ , \tag{2.1}$$

where W denotes weight, [.] is the concatenation operator and $tanh$ is the hyperbolic tangent function.

Finally, we employ modality-wise attention mechanism to fuse visual and audio modalities to obtain the input of the fusion model. Let c_v^t and c_a^t denote the representations of expression and audio CNN models and α_v^t, α_a^t denote their attention weights at time t, respectively. Then, the weighted representation c_f^t, which is the input of fusion model, can be computed as:

$$c_f^t = \sum_{i \in \{v', a\}} \alpha_i^t c_i^t \ , \tag{2.2}$$

$$\alpha_i^t = \frac{exp(\sigma(W c_i^t + b))}{\sum_{k \in \{v', a\}} exp(\sigma(W c_k^t + b))} \ , \tag{2.3}$$

$$c_{v'}^t = tanh(W_r c_v^t + b_r) \ , \tag{2.4}$$

where W and b denote weight and bias of the attention layer, respectively. W_r and b_r denote weight and bias of a single layer MLP for dimension reduction, $\sigma(.)$ is the sigmoid function and α_i is a scalar. Indices v and v' refer to visual modalities before and after dimension reduction, respectively, and index a refers to the audio modality.

Chapter 3

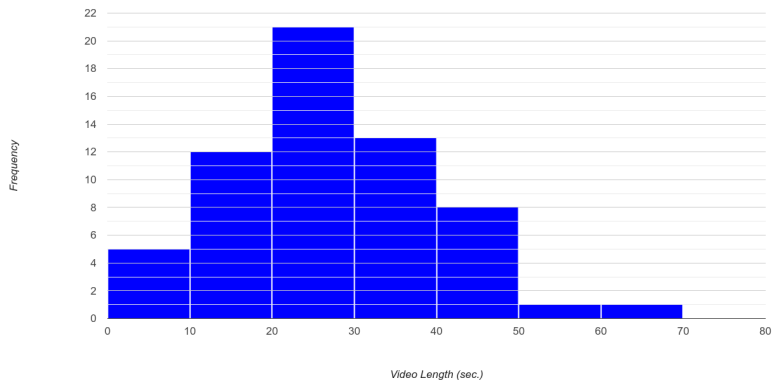
Experiments and Results

In this section, we start with providing the details of the dataset. Then we will present the results of our experiments in the following order. First, the experimental setups used in all of our experiments are given. Then, we present the expression, voice and fusion analyses in details. In our metric calculations (i.e., accuracy, precision, recall and F_1-score), lie is considered as *positive* class and truth as *negative* class throughout the experiments.

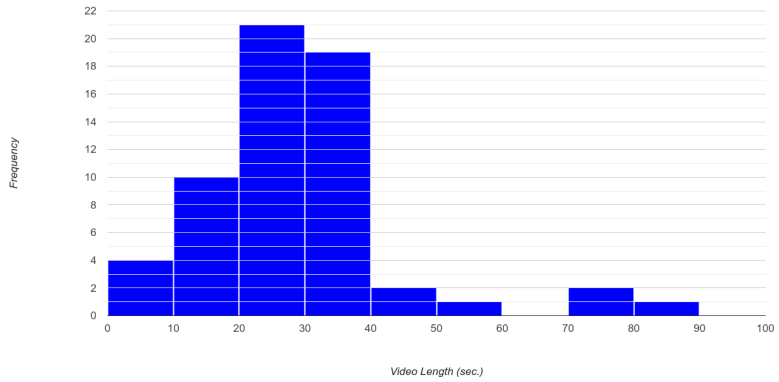
3.1 Dataset

We use Real-life Trial (RLT) dataset [26] which contains videos from real-life high-stakes court trials. Out of 121 videos, 61 deceptive and 60 truthful trial clips are formed by 21 female and 35 male subjects. Ages of the subjects vary between 16 and 60. The average duration of videos is 28 seconds where its 27.72 and 28.28 seconds for lie and truth videos, respectively. Figure 3.1 shows the histogram of video lengths for both classes. Average frame rate (FPS) of the videos is 26.66 where it varies between 10 to 30. The resolution of the videos ranges between 320×240 and 1920×1080 where almost 55% of the videos have less than 640×360 resolution. However, we rescale the facial boundary of frames

as described in Section 2.2.1. Figure 3.2 shows some sample frames from the dataset. Average audio sample rate is 44.7 kHz. Minimum audio sample rate is 44.1 kHz whereas the maximum is 48 kHz. 85% of the audio files (101 out of 121) have 44.1 kHz sample rate. We use the original sample rate of each audio file.



(a)



(b)

Figure 3.1: Histogram of (a) lie video lengths, (b) truth video lengths. Video lengths are given in seconds.

(a)

(b)

Figure 3.2: (a) Visually good samples, (b) Visually bad samples that contains either multiple subjects, occlusion of the main subject, or the face of main subject is not visible.

[26] compels some visual and vocal enforcement for videos, e.g. the face of defendant or witness should be identified during most of the footage and audio quality should be clear enough to understand what he or she is saying. Nonetheless, the videos are still noisy as some videos contains more than one subject and the main subject (the defendant) does not appear in all frames of the video (see Figure 3.2.b). This causes to noisy feature extraction. In the literature, those videos are either discarded or used as is which results in unsound features and thus inaccuracte results. We thereby manually labelled all the frames that the main subject appears solely to obtain better results. If multiple faces appear in the scene, we remove all faces except the main subject. 5 videos are removed (namely, 'trial_lie_045', 'trial_lie_050', 'trial_lie_053', 'trial_lie_060', and 'trial_truth_041') as either having no faces in the scene or occluded faces, which leaves 116 videos. Another important issue that needs utmost attention is that the dataset is unbalanced: the amount of videos per identity differs significantly. Performing K-fold or leave-one-out validation might cause videos of the same person both in testing and training split, resulting with erroneous metrics such as unsound high accuracy.

3.2 Experimental Setup

Leave-one-person-out (LOPO) validation is used to create 53 folds using 116 videos. Since RLT dataset contains imbalanced amount of videos per subject, using K-Fold or leave-one-out validation could cause one subject to appear in both training and test sets. In order to enforce having both balanced splits in terms of classes and subject nonoverlapping, subjects who have either too few (1) or too many videos (20% of the remaining videos) are always kept in the training set. 15% to 20% of the remaining videos are randomly separated as the validation. To have a balanced training set, we randomly downsampled the majority class in terms of quantity to have an equal number of instances from each class. The result of each previous method mentioned in Section 3.3.4 is evaluated with our splits. Therefore, some method's results might differ compared to their original results such as [53] mentioned 71.07% and 73.55% accuracy results for their visual

model with DT and RF classifiers, respectively, which are obtained erroneously with subject overlap. Each DT and RF result is obtained by taking an average of 20 repetitions.

We utilized PyTorch [91] in our deep learning based implementations. 1-layer bidirectional GRU with 64 hidden size is utilized in both visual and audio models. We opt for single layer models as we observed that use of stacked recurrent models are prone to overfitting even with heavy dropout rates. ResNext's cardinality is set to 32 in all experiments. 5 NVIDIA GTX 1080 Ti each having 12 GB memory were used in our experiments.

We trained the visual CNN-GRU model maximum 20 epochs with Adam optimizer using $1e^{-3}$ learning rate, $1e^{-4}$ weight decay and batch size of 16. First, we train the CNN part for 20 epochs and extract features for each epoch which will later be used for the training of GRU model. We opt for extracting features in each epoch due to data augmentation and randomization of batches. Then, using the extracted features, we train the GRU part for maximum 20 epochs with early stopping criteria to prevent overfitting. The reason why we did not train these models jointly is because using end-to-end spatiotemporal and recurrent models simultaneously on images consumes heavy GPU memory even with very small batch sizes due to BPTT.

The other visual model, namely 3D-ResNext, is trained for 60 epochs with SGD optimizer using $1e^{-4}$ learning rate, batch size of 16 and Nesterov's Accelerated Gradient approach (NAG) [92]. Our preliminary results show that using SGD with NAG produces better results than Adam optimizer for our model. Patience of learning rate scheduler is set to 15 epochs with the default reduction rate of 0.1. We used heavy weight decay of $1e^{-2}$ and random data augmentations to eliminate overfitting. During training, random horizontal flip with $p = 0.5$ and multi-scale corner crop with 5 scales such that the original image is cropped between the range 50 to 100% of its original size is utilized. The position of cropping is randomly selected from 4 corners and 1 center. No augmentation is used during inference.

The number of filters in each block of 2D-ResNext-101 in the audio CNN-GRU model is altered so that the model fits into our GPU memory. We reduced them to $1/4^{th}$ of their original values, i.e., 16, 32, 64, and 128 filters are utilized in blocks from 2 to 5, repectively. Furthermore, we added Spatial Dropout with drop probability of 0.2 between the third and fourth, and fourth and fifth bottleneck blocks to prevent overfitting. Note that Spatial Dropout promote independency between the feature maps which results in conceivably highly correlated adjacent pixels being either all 0 (dropped-out) or all active. It is proposed in [93] and it performs better than its standard i.i.d. variant especially when the training set size is small like in our case. The model is trained end-to-end for 150 epochs with Adam optimizer using $1e^{-3}$ learning rate, $1e^{-5}$ weight decay and batch size of 4. Training with higher batch sizes do not fit into our GPU's memory. Learning rate is reduced by 0.1 after each 30 epochs if the validation loss has not been improved. We also early stopped the training if the validation loss has not been improving for 50 epochs.

Finally, the personality model is not trained on RLT as ground truth personality trait scores are not available. Therefore, the pretrained model on First Impressions Dataset (FID) from our work [71] is utilized.

3.3 Results

In this section, we first asses the reliability of our expression, voice and fusion analyses, which were described in Section 2.2, 2.3, and 2.4, respectively. Then, we conclude with comparisons of our models with baseline models from the literature.

3.3.1 Expression Analysis

In this section, we present our facial expression analysis results which includes two different aspects: personality-unaware deception module results and results with the combined used of deception module and personality module, both were

described in Section 2.2.2.

3.3.1.1 Assessment of Personality-Unaware Deception Detection

We give the experimental results of our personality-unaware deception modules in this section. We propose to use two architectures, namely 3D-ResNext [70] and CNN-GRU, as described in Section 2.2.2.1. However, we also present the result of one more network for comparison, namely R(2+1)D [94]. R(2+1)D explicitly factorizes 3D convolutions into two separate and successive operations: a 2D spatial convolution and a 1D temporal convolution. The advantage of such a mechanism is in two-folds. First, it improves the network's capabilities to represent more complex functions by doubling the number of nonlinearities compared to its full 3D counterparts. Second, it facilitates the optimization of the network which produces less errors both in training and test sets especially in deep networks. R(2+1)D [94] performs better than its counterparts such as ResNet-18, ResNet-34 and I3D-RGB [95] on large-scale activity recognition datasets. Note that R(2+1)D does not consider the concept of cardinality in its architecture whereas 3D-ResNext does. Instead of training from scratch, we finetune R(2+1)D-101 which is pretrained on Kinetics [75] and Moments in Time [96] datasets. The results are given in Table 3.1.

Table 3.1: Results of the personality-unaware deception module

Model	**Accuracy**	**Precision**	**Recall**	**F_1-score**
R(2+1)D [94]	0.4790	0.4545	0.2542	0.3260
3D-ResNext [70]	0.5862	0.5717	0.6316	0.6002
CNN-BiGRU	0.4655	0.4545	0.4386	0.4464

Even though R(2+1)D is found superior to its counterparts in [94] and it is pretrained on MIT dataset in addition to Kinetics, 3D-ResNext outperformes it for our task. This also shows the superioty of ResNext over ResNet variants as R(2+1)D is found better than ResNet variants in [94]. We believe this is due to

the concept of cardinality and combined use of 3D kernels in spatiotemporal dimensions, that 3D-ResNext uses at the cost of extra computations. 3D-ResNext also outperforms the CNN-BiGRU architecture. Nonetheless, it's important to emphasize that the validation accuracy of CNN-BiGRU (0.6322) is significantly higher than that of 3D-ResNext (0.5466). By inspecting the test and validation figures, we deduce that the complex nature of CNN-BiGRU architecture may require more data to generalize well since having a small dataset creates discrepancies between the validation and test sets of folds. However, since its validation set accuracy is the highest, we continue to use it in our fusion module although its test set accuracy is the worst as given in Table 3.1.

3.3.1.2 Assessment of Personality-Aware Deception Detection

In this section, we present our findings of integrating the personality features into the personality-unaware deception modules as depicted in Figure 2.2. Our personality module consists only of 3D-ResNext based on our work [71] as stated in Section 2.2.2.2. We experimented various ways of integrating personality features into our personality-unaware deception module (PuDM). As stated in Section 2.2.2.1, we have two options for PuDM: 3D-ResNext [70] and CNN-GRU. Hence, we first present the details for the use of personality features with 3D-ResNext, and then continue with the CNN-GRU architecture.

3.3.1.2.1 Use of Personality Features with 3D-ResNext

We give the experimental results of various ways to integrate the personality features into our personality-unaware deception module (PuDM) which is 3D-ResNext in this case. For all the methods explained below, the personality features are integrated to PuDM features before the FC layer as shown in Figure 2.2.

The first way is directly concatenating the personality features with PuDM features. It implies that the extracted personality features are concatenated *per se* with the PuDM features along the feature dimension. We call it *Direct Concat*.

The second way is via the use of an MLP model that consists of two blocks of BN, Dropout (with drop probability of 0.5) and linear layers, where each block is followed by a ReLU activation function and linear layers have 1024 and 512 output neurons, respectively. We referred to it as *MLP*. As for the third way, we define another custom MLP that contains two blocks of BN layer, 1×1 convolutional layer and ReLU activation function, where convolutional layers have 1024 and 512 filters, respectively. It is called as 1×1 *Conv*. Finally, in the fourth and last way, we use something from the literature instead of the aforementioned custom layers: SCAN [86]. It is originally proposed as a model compression and acceleration technique with an attention mechanism. In short, it divides neural networks into multiple sections from different depth levels and each section contains an attention module to focus the relevant parts of the features followed by a shallow classifier which is used for knowledge distillation.

Note that our use of SCAN [86] for integration diverges from the original implementation in two substantial ways. First, we only apply SCAN to already extracted personality features from 3D-ResNext as opposed to applying it to all intermediate features of the backbone during the training. Secondly, we used their attention module and shallow classifier without FC layers. In other words, we do not employ self-distillation. Since both of these differences constitutes almost the main idea of SCAN, we call this kind of personality features integration as SCAN-*ish* since it is more appropriate to prevent a possible fallacy. The results of four different ways to integrate personality features into 3D-ResNext PuDM features are given in Table 3.2. The first line contains the results of our personality-unaware deception module (3D-ResNext) to better observe the effect of personality integration.

Interestingly, the way of integrating personality features has a great importance. Concatenation of the personality features to the PuDM features via *Direct Concat* decreases the results of sole use of 3D-ResNext as PuDM. This might be due to high feature correlation which increases redundancy in representations, given that the personality module is also 3D-ResNext. Similarly, *MLP* fails to fulfill the common expectation that linear layers with nonlinear rectifiers should adjust the features into better representations and results are decreased. On the

Table 3.2: 3D-ResNext [70] results with and without personality features

Integration Method	PuDM Features	Personality Features	Accuracy	Precision	Recall	F_1-score
None	✓		0.5862	0.5717	0.6316	0.6002
Direct Concat	✓	✓	0.5259	0.5128	0.7018	0.5926
MLP	✓	✓	0.5259	0.5200	0.4561	0.4860
SCAN-*ish* [86]	✓	✓	0.6034	0.6667	0.3860	0.4889
1×1 Conv	✓	✓	0.6034	0.5846	0.6667	0.6230

other hand, both SCAN-*ish* and 1×1 *Conv* are found to profitable as they improve the results compared to the sole use of PuDM features in the classification by 3% (relative). Both of them perform on par in terms of accuracy, however, 1×1 *Conv* has a better balance between precision and recall (i.e., its F1-score is higher than that of SCAN-*ish*). Also, SCAN-*ish* drops the recall of PuDM only features by 38.9% (relative) at the expense of less increase in precision by 16.6% (relative). The success of 1×1 *Conv* over SCAN-*ish* is possibly due to its simpler nature compared to more complex attention-based method, given that our dataset is scarce in number of samples. Hence, we conclude that it's better to pass personality features through a tailored learnable layer before concatenating them with the PuDM features and the choice of that tailored layer should be as decribed in the 1×1 *Conv* method.

3.3.1.2.2 Use of Personality Features with CNN-GRU

In this section, we provide the results of our experiments for integrating the personality features into our personality-unaware deception module (PuDM) which is CNN-GRU in this case. This time, we also analyze the effect of integration point in addition to the integration method. Note that Figure 2.2 depicts the the final version of the integration point if CNN-GRU is chosen as PuDM. Since 1×1 *Conv* is outperformed the other integration methods in Section 3.3.1.2.1, we only provide the results with it in our integration point experiments. After we decide the best point of integration, we also provide the results with *Direct*

Concat method for the sake of comparison.

For our CNN-GRU architecture, we have two options for the integration point of personality features. We can either unite them with the CNN-only extracted PuDM features (before feeding into GRU) at each temporal unit of the GRU model by repeatedly using the same personality features (since the personality of the subject cannot change over a short period of time) and feed the unified features to the GRU or merge them with the last hidden state representation of the GRU just before the fully connected (FC) layer. Table 3.3 presents the results of these two options with the *1×1 Conv* integration method which described in Section 3.3.1.2.1. The first row contains the results of CNN-GRU (PuDM) without the integration of personality features for highlighting the difference better.

Table 3.3: The effect of integration point of personality features on CNN-GRU architecture with *1×1 Conv* integration method

Integration Point	**PuDM Features**	**Personality Features**	**Accuracy**	**Precision**	**Recall**	**F_1-score**
None	✓		0.4655	0.4545	0.4386	0.4464
At temporal units	✓	✓	0.5172	0.5094	0.4737	0.4909
At FC layer	✓	✓	0.5603	0.5625	0.4737	0.5143

By inspecting the Table 3.3, we have two outcomes. First and the obvious one is the integration of personality features with *1×1 Conv* method enhances the results of the personality-unaware deception module (CNN-GRU) regardless of the integration point between 11% to 20% (relative). Second one is that the integration point is another important facet in addition to the integration method. Integrating the personality features at the FC layer produces 8% (relative) better results compared to the integration at each temporal unit. Considering something that does not change over time alongside the temporally changing facial dynamics apparently causes a problem for GRU. It may be possibly due to the redundancy of the repetition of same personality features. Consequently, we proceed our experiments with the integration at FC layer.

Next, we analyze the efficacy of different integration methods for CNN-GRU.

Table 3.4 presents the results for different integration methods of personality features with the CNN-GRU PuDM features at FC layer.

Table 3.4: The effect of integration method of personality features on CNN-GRU architecture at FC layer

Integration Method	**PuDM Features**	**Personality Features**	**Accuracy**	**Precision**	**Recall**	**F_1-score**
None	✓		0.4655	0.4545	0.4386	0.4464
Direct Concat	✓	✓	0.4914	0.4808	0.4386	0.4587
1×1 Conv	✓	✓	0.5603	0.5625	0.4737	0.5143

Like in the case of 3D-ResNext PuDM features, *1×1 Conv* outperforms the *Direct Concat* method in the integration of personality features with the CNN-GRU PuDM features at FC layer. It is possibly because the models learn how to interpret the personality features alongside the personality-unaware deception module (PuDM) features when used with learnable convolutional layers. On the other hand, as opposed to the case of 3D-ResNext PuDM features, *Direct Concat* method improves the results of sole use of CNN-GRU PuDM features. This may imply that, depending on the model, different features might be valuable assets.

Finally, we combine the results of 3D-ResNext and CNN-GRU personality-unaware deception modules (PuDM) with the integration of personality features using the best integration method and point found earlier in this section, in Table 3.5.

Table 3.5: Deception detection results with and without personality features

Model	**PuDM Features**	**Personality Features**	**Accuracy**	**Precision**	**Recall**	**F_1-score**
3D-ResNext [70]	✓		0.5862	0.5717	0.6316	0.6002
3D-ResNext + 1×1 Conv	✓	✓	0.6034	0.5846	0.6667	0.6230
CNN-BiGRU	✓		0.4655	0.4545	0.4386	0.4464
CNN-BiGRU + 1×1 Conv	✓	✓	0.5603	0.5625	0.4737	0.5143

Note that the validation set accuracy of CNN-BiGRU model is higher than that of 3D-ResNext model when both models consider the personality features too in Table 3.5. Therefore, we carry our analyses with it in spite of its worse test set accuracy compared to 3D-ResNext. By inspecting the results presented in Table 3.2, 3.3, 3.4 and 3.5, we conclude that personality features give valuable insights to the models about whether the subject is lying. It's favourable to pass personality features through a tailored learnable layer before concatenating them with the PuDM features and the design of that tailored layer should be as decribed in the 1×1 *Conv* integration method, i.e., two blocks of BN layer, 1×1 convolutional layer and ReLU activation function, where convolutional layers have 1024 and 512 filters, respectively.

3.3.2 Voice Analysis

We present our experimental findings for the voice analysis in this section. As described in Section 2.3, we employ CNN-GRU architecture in audio analysis. However, we also compare its performance against several possible combinations. First, we start with analyzing the choice of different recurrent architectures. Then, we proceed to analyze the choice of residual model in combination with the chosen recurrent model. Finally, we summarize our results with the best possible choices and draw conclusion by interpreting the results.

3.3.2.1 The Choice of Recurrent Model

We start our analyses with investigating the choice of recurrent architecture. We evaluate three variations of recurrent architectures: recurrent neural network (RNN) [97], long short-term memory (LSTM) [98] and GRU [99, 100]. In short, the differences between these three architectures are as follows. RNN is the inital model that has been used in the literature. However, it lacks the ability to model long temporal sequences due to vanishing (or exploding) gradient problem where when the sequence gets longer, gradient flow of backpropagation starts to cease

(or go infinite) towards the beginning of the sequence. Thus, LSTMs are proposed as a remedy to vanishing gradient problem. It has the ability to keep track of arbitrarily long input sequences. Both of these models date back to 1980-90s. GRU [99, 100], on the other hand, is proposed recently in 2014 and its basically an LSTM without a forget gate, thus having less parameters. It has been shown to perform superior results over LSTM especially on the small datasets since having less parameters make it easy to train.

In our recurrent model analyses, we use 1-channel melspectrogram features with 25 msec. window size instead of our proposed 3-channel melspectrogram features. This is because each of these recurrent models expects a 1D vector in each temporal unit, which in that case makes it hard to use 3-channel features. Indeed, we could flatten the channels and use them as a vector, however, it would break the spatial and temporal patterns between the channels. Thus, we proceed with using 1-channel features. We conduct many experiments with each of these models using different hyperparameters such as being bidirectional, hidden size, number of stacked layers, dropout rate etc. and present only a subset of these results which we found valuable to mention. Hence, we provide the results of bidirectional models with various hidden sizes for both single layer (in Table 3.6) and 2-layer (stacked) versions (in Table 3.7).

Table 3.6: Results of 1-layer recurrent models on 1-channel melspectrogram features

Model	**Hidden Size**	**Accuracy**	**Precision**	**Recall**	**F_1-score**
BiRNN	{32, 64, 128}	{0.3966, 0.4224, 0.4224}	{0.4154, 0.4429, 0.4394}	{0.4576, 0.5254, 0.4915}	{0.4355, 0.4806, 0.4640}
BiLSTM	{32, 64, 128}	{0.4310, 0.3534, 0.4655}	{0.4407, 0.3947, 0.4776}	{0.4407, 0.5085, 0.5423}	{0.4407, 0.4444, 0.5079}
BiGRU	{32, 64, 128}	{0.5259, 0.5776, 0.5776}	{0.5322, 0.5606, 0.5756}	{0.5593, 0.6491, 0.6441}	{0.5454, 0.6016, 0.6079}

For BiRNN, the hidden size of 64 and 128 produces on par results although hidden size of 32 has the highest validation accuracy of 0.6387, whereas the others have 0.6182 and 0.5959, respectively. However, all there choices produces very low accuracies, since the simple nature of RNN is not enough to solve this problem using the given features. We deduce that by inspecting the training accuracies of the folds: we see that even the training accuracy does not improve after 70-80%, where it generally fluctuates around 60-70% for most folds.

For BiLSTM, regardless of the hidden size chosen, we can see that it clearly overfits the training data and lacks generalization since its test accuracies presented in Table 3.6 are lesser than that of BiRNN even though they almost have similar validation scores. Also, while its training accuracies reach around 80-90% for many folds, the validation accuracies start to decrease after some epochs. Even though we follow an early-stopping mechanism, it is not sufficient to learn a better generalizable model with BiLSTM on our dataset.

As it can be seen, BiGRU outperforms the other recurrent models by far, in all metrics. It also achieves the highest validation accuracies irrespective of the hidden size. 64 and 128 hidden sizes performs on par though 64 hidden size has the highest validation accuracy (0.6713) as opposed to that of 128 hidden size (0.6583). Therefore, 64 hidden size ought to be chosen. The success of BiGRU over other architectures can be attributed to its simple yet effective design which makes it a perfect choice for moderately small datasets.

Seeing that BiGRU performs better than BiLSTM and BiRNN models with single layer, we presume it will also outperform them when stacked models are employed as the others are not good choices by design for our dataset. However, for the sake of completeness, we provide the results with 2-layer stacked models in Table 3.7 using dropout rate of 0.5 between the stacked layers.

Table 3.7: Results of 2-layer recurrent models on 1-channel melspectrogram features

Model	**Hidden Size**	**Accuracy**	**Precision**	**Recall**	**F_1-score**
BiRNN	{32, 64}	{0.5000, 0.4397}	{0.5079, 0.4423}	{0.5423, 0.3898}	{0.5245, 0.4144}
BiLSTM	{32, 64}	{0.3966, 0.3966}	{0.4068, 0.4267}	{0.4068, 0.5424}	{0.4068, 0.4776}
BiGRU	{32, 64}	{0.5862, 0.4828}	{0.5902, 0.4925}	{0.6102, 0.5593}	{0.6000, 0.5238}

The first thing to notice when we look at the Table 3.7, regardless of the model chosen, 64 hidden size results perform worse than that of 32 hidden size. This also the case for the validation set accuracies for all models. This shows that there is no need for higher hidden sizes as even with these sizes models are prone to overfitting. Next, we see a similar pattern to what we observe in Table 3.6. BiGRU achieves the best performance both in test and validation set accuracies,

as foreseen, regardless of the hidden size chosen. BiGRU with 32 hidden size performs better than BiGRU with 64 hidden size both in test and validation set accuracies. However, its performance reduces significantly in the case of 2 stacked layers with dropout rate of 0.5. Thus, we specifically investigate whether it can achieve better results when different dropout rates are utilized. Table 3.9 present the results of 2-layer stacked BiGRU model with various dropout rates.

Table 3.8: Results of 2-layer BiGRU model on 1-channel melspectrogram features with various dropout rates

Dropout Rate	**Hidden Size**	**Accuracy**	**Precision**	**Recall**	**F_1-score**
0.3	{32, 64}	{0.4828, 0.4568}	{0.4928, 0.4697}	{0.5763, 0.3898}	{0.5313, 0.4260}
0.5	{32, 64}	{0.5862, 0.4828}	{0.5902, 0.4925}	{0.6102, 0.5593}	{0.6000, 0.5238}
0.7	{32, 64}	{0.5172, 0.5086}	{0.5283, 0.5143}	{0.4746, 0.6102}	{0.5000, 0.5582}
0.9	{32, 64}	{0.4310, 0.5259}	{0.4588, 0.5313}	{0.6610, 0.5763}	{0.5416, 0.5529}

The results with 64 hidden size in Table 3.9 suggest that increasing the dropout rate between the stacked layers increases the performance of 2-layer BiGRU model, where it achieves the highest accuracy with 0.9 dropout rate. Nonetheless, when we look at their validation set accuracies, we see that it starts to decrease after dropout rate of 0.5 where it achieves the highest validation accuracy with 0.6471. Therefore, the best result with 64 hidden size in Table 3.9 is with 0.5 dropout rate even though its worse test set results. The results with 32 hidden size produces more promising results compared to 64 hidden size. However, we see a similar trend here as well: Increasing the dropout rate increases the performance of the stacked model. However, after 0.7 dropout rate, validation accuracy starts to decrease. Therefore, the best results in Table 3.9 is with 32 hidden size, 0.7 dropout rate (as 32 hidden size has higher validation accuracy than that of 64 hidden size) in spite of its worse test set results, similar to 64 hidden size results. As we can see, none of these stacked 2-layer models can beat the performance of 1-layer BiGRU model with 64 hidden size.

Finally, we conclude that use of stacked layers are prone to overfitting even with heavy dropout rates. Our experiments show that 1-layer BiGRU model with 64 hidden size is the best choice among various setups using the 1-channel

melspectrogram features.

3.3.2.2 The Choice of Residual Model

Since we decide the best recurrent model in the previous section, it is now time to decide residual model when it is combines with the chosen recurrent model. Since we analyze the effect of residual model in CNN-GRU architecture, we also use our 3-channel melspectrogram features in our experiments in addition to 1-channel melspectrogram features, where we can also observe the efficacy of our 3-channel feature design in various residual architectures.

In addition to 2D-ResNext [73] (which is decided to be used in our voice analysis in Section 2.3), we experiment with 2D-ResNet-34 [74]. Since we finetune 2D-ResNext using the weights of Kinetics [75] datasets, we also employed pretrained weights of 2D-ResNet-34 on ImageNet [101]. Because 1-layer BiGRU with 64 hidden size is found to be the best option in Section 3.3.2.1, we conduct the following experiments using it as our recurrent model. The results using both 1-channel and 3-channel features are presented in Table 3.9.

Table 3.9: Results of CNN-RNN models using different residual architectures

Model	**Feature**	**Accuracy**	**Precision**	**Recall**	**F_1-score**
2D-ResNet + BiGRU	1-channel Mel-spectrogram (25 ms - 75% overlap)	0.5690	0.5692	0.6271	0.5967
2D-ResNext + BiGRU	1-channel Mel-spectrogram (25 ms - 75% overlap)	0.6034	0.6038	0.5614	0.5818
2D-ResNet + BiGRU	3-channel Mel-spectrogram (25/50/100 ms - 75% overlap)	0.6207	0.6471	0.5593	0.6000
2D-ResNext + BiGRU	3-channel Mel-spectrogram (25/50/100 ms - 75% overlap)	0.6379	0.6056	0.7544	0.6719

The outcomes of the results presented in Table 3.9 are in two-folds. First, 2D-ResNext performs better than 2D-ResNet both in test and validation set accuracies regardless of the features. This is indeed in accordance with our expectations as it has more layers and it has the *cardinality* concept which is shown to be

effective as described in Section 2.2.2.1. Secondly, we see that forming 3-channel melspectrogram features as depicted in Section 2.3 are found to be advantageous over their 1-channel counterparts regardless of the residual model used. However, we note that the use of 3-channel melspectrogram features reduces the validation set accuracy of 2D-ResNet model even though it facilitates achieving better test set accuracy. Precision of 2D-ResNext when 3-channel melspectrogram features are used does not change much compared to 1-channel features: only 3‰ increase. Nonetheless, we see a huge enhancement in recall by 34% (relative) which also boosts the accuracy by almost 6% (relative). We believe that this is because 3-channel melspectrogram features use coarse-to-fine grained signals together (i.e., audio frames with window size of 25, 50 and 100 milliseconds) which, apparently, carry more information and are more enlightening.

Finally, we compare the use of CNN-GRU architecture in contrast to RNN only architecture in Table 3.10 which also presents our best result obtained during the voice analyses.

Table 3.10: Results of CNN-RNN and RNN only architectures on melspectrogram features

Model	**Feature**	**Accuracy**	**Precision**	**Recall**	**F_1-score**
BiGRU	1-channel Mel-spectrogram (25 ms - 75% overlap)	0.5776	0.5606	0.6491	0.6016
2D-ResNext + BiGRU	1-channel Mel-spectrogram (25 ms - 75% overlap)	0.6034	0.6038	0.5614	0.5818
2D-ResNext + BiGRU	3-channel Mel-spectrogram (25/50/100 ms - 75% overlap)	0.6379	0.6056	0.7544	0.6719

We conclude by interpreting the results presented in Table 3.10. The results show that use of CNN-GRU architecture is superior to sole use of GRU model. Our motive behind using CNN-GRU architecture holds as it outperforms sole use of recurrent model GRU. Even though CNN-GRU has lower recall when using 1-channel features, the predictions of it is more precise and sound. When 3-channel features are employed, its precision also increases as well as the recall. Apart from considering both spatial and temporal relations, the motive behind the success of

CNN-GRU may be due to better modelling capability of longer sequences. After having passed through CNN, the original temporal dimension is decreased and it may ease the burden of GRU matching temporal patterns given that training recurrent models to learn long-term dependencies is difficult and shorter temporal sequences are more robust to vanishing or exploding gradient problems in temporal models [102, 103, 104]. Furthermore, the results emphasizes, once more, the efficacy of our 3-channel coarse-to-fine grained melspectrogram features. Using those features produces 6%, 3‰ , 34% (all relative) improvements for accuracy, precision and recall, respectively, over the use of 1-channel features with the proposed CNN-GRU architecture.

3.3.3 Assesment of Fusion

We fuse visual and audio CNN-GRU models in our fusion experiments as presented in Section 2.4. For our visual model, the one with personality features passed through an MLP (1×1 *Conv* method) is chosen. As we see from Table 3.5 and Table 3.10, the performance of audio model is almost 14% (relative) better than visual model with personality features. This may imply that audio features carry more information compared to visual features and visual features are more complex to extract valuable insights.

We try various fusion strategies and give the results of the ones that has highest validation accuracies in Table 3.11. First three results are obtained without using the attention mechanism that is proposed in Section 2.4. Instead, we fused the expression and voice representations by concatenation along the feature dimension so that the dimension of final feature vector is the sum of individual representation's dimensions (i.e., $2048 + 512 = 2560$) in each time step. In Projection$_B$ strategy, we project each representation to a new subspace where, hopefully, two feature sets can be represented jointly *before* the concatenation (which is why it has the subscript B). Similarly, in Projection$_A$ we project expression and voice representations to a new subspace *after* the concatenation. Projection$_{BA}$ is the combination of both where two projections are employed in total: one before

and another after the concatenation. These projections are in essence a linear layer followed by a hyperbolic tangent function to obtain a nonlinear mapping between the source and target subspaces similar to Equation 2.1 but using the representation of visual and audio CNNs instead of the hidden state representations h_v and h_a in the equation. In Input_{hidden}, on the other hand, we use the hidden representations of the expression and voice GRU models in each time step instead of the representations extracted with the corresponding CNN models in each modality. They are fed to the attention mechanism similar to how CNN representations are fed in Equation 2.2 and 2.3 but this time without an additional reduction step (i.e., Equation 2.4 is not used). Finally, in the 'Proposed' strategy, we use the attention mechanism along with the representations obtained from the corresponding CNN models as described in Section 2.4.

Table 3.11: Results of fusion BiGRU model with different strategies

Strategy	**Accuracy**	**Precision**	**Recall**	**F_1-score**
Projection_B	0.4310	0.4211	0.4211	0.4211
Projection_A	0.5948	0.5658	0.7544	0.6466
Projection_{BA}	0.5948	0.6316	0.4211	0.5053
Input_{hidden}	0.5690	0.5636	0.5439	0.5536
Proposed	0.6466	0.6429	0.6315	0.6371

The results of Projection_B strategy in Table 3.11 suggest that it is extremely important to map the representations coming from totally different spaces to a common subspace before using them. We believe it is the reason why Projection_B fusion strategy figures are extremely low since it applies the projection before the concatenation and because of this the projection layers do not learn useful mapping functions that will represent two feature sets in a subspace with common dynamics after the concatenation. This hypothesis is also inline with the results of Projection_A and Projection_{BA} since both of these projections apply at least one projection after the concatenation. Results of Input_{hidden} strategy is also interesting. Compared to the CNN representations, hidden representations of expression and voice GRU models do not bring much information for the fusion albeit the use of attention. When the proposed strategy is used, we achieve the

best results both in test and validation set. This also demonstrates the power of attention mechanism over the straightforward concatenation strategies, e.g., projection based strategies. We improve the accuracy of our model to 64.66% which is 15.4% and 1.4% (both relative) increase for visual and audio models (see Table 3.5 and Table 3.10). Precision is increased by 14.3% and 6.2% (both relative) for visual and audio models, respectively. On the other hand, recall is increased by 33.3% (relative) for the visual model and decreased by 19.5% (relative) for the audio model. We can conclude that the overall performance of fusion model is better than the unimodals.

3.3.4 Comparison to Other Methods

In this section, we present the baseline methods from the literature and compare our models with them under the visual, voice and fusion comparison sections.

3.3.4.1 Comparison to Visual Models

We first explain, in detail, the visual baseline methods whose results are given in Table 3.12. Then, we show that the proposed visual model outperforms them. Furthermore, we show the efficacy of using personality features in baseline models as well.

[26] is implemented with DT and RF classifiers with default parameters as mentioned in their papers using scikit-learn [107]. Originally, the authors use hand-labeled binary features such as smile, laughter, scowl, gaze, frowning, head movements, hand trajectory etc. Since we focus solely on facial expression features in our visual analysis, we excluded hand-related features from their feature set to obtain comparable results. It should be noted, however, that their method cannot be counted as a fully automated deception detection model and the performance of their model indeed relies heavily on the performance of human annotators.

Table 3.12: Visual baseline results

Model	Feature	Accuracy	Precision	Recall	F_1-score
DT* [26]	Hand-labeled	0.6711	0.6444	0.7377	0.6879
RF* [26]	Hand-labeled	0.7065	0.7004	0.7035	0.7019
DT* [53]	OpenFace	0.4966	0.4842	0.3772	0.4241
RF* [53]	OpenFace	0.5578	0.5720	0.3974	0.4690
RF [54]	Motion	0.5431	0.5459	0.4175	0.4731
Time-CNN [105, 106]	OpenFace	0.5070	0.5342	0.4526	0.4900
RF [55]	AUs	0.4224	0.2726	0.1070	0.1537
NN [55]	AUs	0.4000	0.2843	0.1456	0.1926
3D-ResNext [70]	CNN	0.5862	0.5717	0.6316	0.6002
CNN-BiGRU	CNN	0.4655	0.4545	0.4386	0.4464

*: only facial expression features are used

[53] is tested with DT and RF classifiers with default parameters as in the paper using scikit-learn [107]. Feature vectors for each frame is obtained via OpenFace [69] in default output (i.e. basics, gaze, pose, 2D and 3D facial landmark locations, rigid and non-rigid shape parameters, action units). To summarize each video with a single vector, statistical functionals (max, min, mean, median, std, variation, kurtosis, skewness, 25%, 50% and 75% percentiles) are applied to frame-level feature vectors.

[54] is evaluated using RF classifier with the IDT based motion features provided by the authors.

[106] is a CNN model originally proposed for univariate and multivariate time series classification. It tries to discover time series patterns through 1D convolutions on the temporal dimension of feature vectors. It is a very small model, comprised only of two convolutional, two pooling and one fully connected layer. We used the implementation provided by [105] with the features extracted via OpenFace [69] in default output.

[55] proposes subject-level instead of video-level detection of deceit in RLT dataset. They are using both manually annotated and automatically extracted features to feed SVM, RF and NN classifiers. In order to obtain comparable results, we only consider their automatically extracted visual features which are AUs extracted with OpenFace [69] and use them in video-level deception detection. Since they performed hyperparameter optimization for their SVM classifier for each test instance and RF and NN classifiers are found to be more robust to the choice of hyperparameters in their work, we only evaluate their RF and NN classifiers with the hyperparameters given in their paper. After obtaining 18 binary indicators of AUs per frame, average of frame indicators is used to obtain single feature vector per video. Then these video features are fed to RF and NN.

Having seen the baseline results in Table 3.12, we integrate our personality features into some of the recently published baseline models in addition the proposed models for comparison. The results are presented in Table 3.13. Facial features refer to AUs for [55] and OpenFace [69] extracted features in default format for Time-CNN [105, 106], both using the normalized facial frames as described in Section 2.2.1. Personality features refer to the features extracted using the personality module as described in Section 2.2.2.2.

As can be seen from Table 3.13, the proposed visual models outperform the baseline models when personality features are used. Moreover, the use of personality features alongside the facial expression features even helps baseline methods to achieve better results in addition to our personality-unaware deception modules. When personality features are used, both RF and NN models of [55] improved their performances by 13% and 27% (both relative), respectively. The improvement is more significant in NN classifier as it has more complexity to deal with high dimensional data.

3.3.4.2 Comparison to Voice Models

Table 3.14 shows the comparison of some of the recent models from the literature alongside the proposed voice model. For RF and NN implementation of [55], we

Table 3.13: Visual baseline results with and without personality features

Model	Facial Features	Personality Features	Accuracy	Precision	Recall	F_1-score
RF [55]	✓		0.4224	0.2726	0.1070	0.1537
RF [55]	✓	✓	0.4793	0.3871	0.1018	0.1612
NN [55]	✓		0.4000	0.2843	0.1456	0.1926
NN [55]	✓	✓	0.5069	0.4791	0.3912	0.4378
Time-CNN [105, 106]	✓		0.5070	0.5342	0.4526	0.4900
Time-CNN [105, 106]	✓	✓	0.5138	0.5395	0.4526	0.4922
3D-ResNext [70]	✓		0.5862	0.5717	0.6316	0.6002
3D-ResNext + 1×1 Conv	✓	✓	0.6034	0.5846	0.6667	0.6230
CNN-BiGRU	✓		0.4655	0.4545	0.4386	0.4464
CNN-BiGRU + 1×1 Conv	✓	✓	0.5603	0.5625	0.4737	0.5143

used the standard deviation of subjects pitch (pitch $std - f_0$) as it is found to be the best acoustic feature in their setup.

NN [55] performs the worst among other models. Having single dimensional pitch features might be reason behind RF's success compared to NN. Even though Time-CNN has the highest recall, it suffers from low precision. This might be interpreted as its predictions are biased towards the lie class. The best performing model, on the other hand, is our CNN-GRU model followed by BiGRU.

The effect of feature channel size can also be seen from Table 3.14. Except Time-CNN, using 3-channel features boosts the recall of all models dramatically while also improving the precision. Note that three different features are concatenated along the feature dimension rather than forming channels in Time-CNN. Increasing the feature dimension does not help it to catch subtlety in the data. The reason of decrease in Time-CNN's performance may be due to its use of 1D convolutions on the temporal dimensions. Also, this transformation of features

Table 3.14: Audio baseline results

Model	Feature	Accuracy	Precision	Recall	F_1-score
RF [55]	Pitch (std-f_0)	0.5264	0.4855	0.4361	0.4595
NN [55]	Pitch (std-f_0)	0.4793	0.4592	0.4205	0.4390
Time-CNN [105, 106]	1-channel Mel-spectrogram (25 ms - 75% overlap)	0.5017	0.4696	0.7895	0.5889
Time-CNN [105, 106]	Mel-spectrogram (25/50/100 ms - 75% overlap)	0.4655	0.4363	0.3333	0.3779
BiGRU	1-channel Mel-spectrogram (25 ms - 75% overlap)	0.5776	0.5606	0.6491	0.6016
2D-ResNext + BiGRU	1-channel Mel-spectrogram (25 ms - 75% overlap)	0.6034	0.6038	0.5614	0.5818
2D-ResNext + BiGRU	3-channel Mel-spectrogram (25/50/100 ms - 75% overlap)	0.6379	0.6056	0.7544	0.6719

breaks the temporal and spatial relations between our synchronized coarse-to-fine grained features. Considering other models, 3-channel features help models to distinguish truth and lie classes more easily without creating a bias towards one class. Compared to use of 1-channel features, using 3-channeled melspectrogram features might bring significant performance improvements up to 5.7% (relative).

3.3.4.3 Comparison to Audio-Visual Models

To compare our fusion performance, we consider visual baseline models without the integration of the personality features. We applied late fusion for Time-CNN where lie scores of expression and voice models are fused with equal weights. Also, 1-channel melspectrogram features are used for Time-CNN instead of flattened 3-channel version since it is found to perform worst in Section 3.3.4.2. We apply early fusion for visual and audio modalities of NN [55] as they proposed. Table 3.15 presents the results of these methods along with our method's.

After the fusion, we obtain a slight increase in the performance of [55]. On the

Table 3.15: Fusion baseline results

Model	Accuracy	Precision	Recall	F_1-score
NN [55]	0.4817	0.4236	0.3029	0.3532
Time-CNN [105, 106]	0.5040	0.4997	0.5812	0.5374
BiGRU	0.6466	0.6429	0.6315	0.6371

other hand, the performance of Time-CNN [105, 106] is decreased slightly. For our fusion model, we improved the unimodal results and achieved 15.4% and 1.4% (both relative) increase compared to expression and voice modules, respectively. The results of the proposed fusion model are significantly higher than others. Thus, we can conclude that the overall performance of fusion model is better than that of unimodals and our final model's performance improves upon the results of other work.

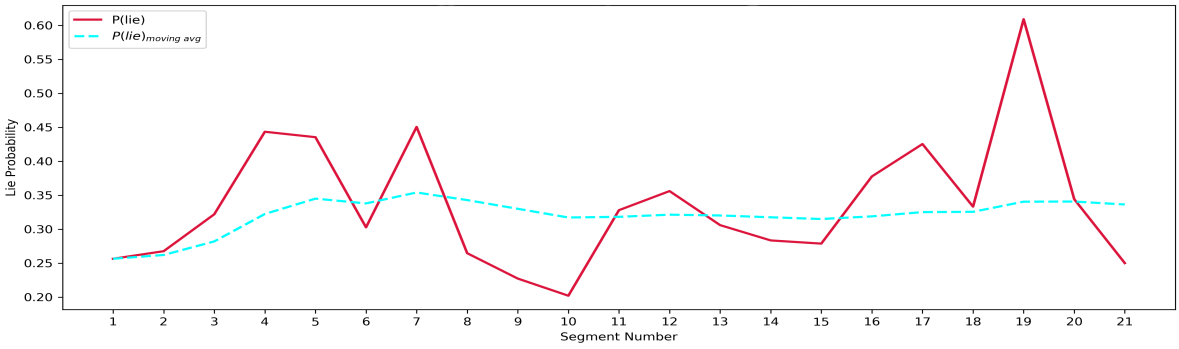
3.3.5 Qualitative Analysis

We provide the details of our qualitative analysis using the expression modules (i.e., deception module and personality module together) in this section both for extracting informative cues and failure cases.

3.3.5.1 Discussion on Informative Cues

When we analyze the model's predictions to elicit some informative cues, we observe that the model is highly susceptible to minor changes in subject's face. For instance, in the case of talking loudly, using hands effectively, and shifting from one facial expression to another, the model suspects deceptive behavior when all or some of these start to cease. It is also the case for the other way around, i.e., very silent and smooth talking session followed by a sudden activity. In other words, the model behaves like an anomaly detector. To illustrate some of these situations, we choose two exemplars that have moderately higher probabilities among many successful classifications. Figure 3.3 and Figure 3.4 present the

visualizations of these two successfully classified cases from both truthful and deceptive classes, respectively. Note that the sampled frames from the segments in these figures are chosen after ignoring the first ten frames from each segment with the motivation that segment scores are mostly depend on the content presented towards the center. This is just a preference and the selection of these frames does not change the cues presented in this section at all.



(a)

(b)

(c)

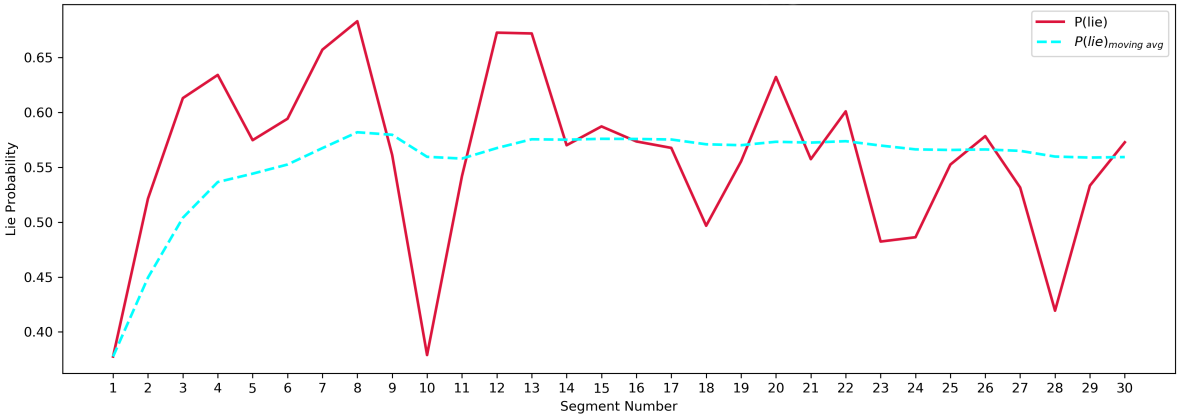
Figure 3.3: Qualitative analysis visualizations for the video *trial_truth_039* which is correctly classified as *truthful* since the mean lie probability of segments is below 50%. (a) Change of lie probability over time, (b) 280., 290., 300. frames (left to right) from the 10^{th} segment which produces the lowest lie probability, (c) 550., 560., 570. frames (left to right) from the 19^{th} segment which produces the highest lie probability.

When we inspect the change of lie probability over time in Figure 3.3(a), we see that lie probabilities of segments do not go beyond the 50% threshold most of the time and mean of them is below 50% at the end which means that the video is correctly classified as *truthful*. We then present some of the frames corresponding to the peak points in the lie probabilities. Figure 3.3(b) shows the frames taken from the 10^{th} segment which produces the lowest lie probability over the entire video segments. The subject looks down throughout this segment and closes her eyes while talking. The model predicts that the subject is the most truthful in this segment which might be due to the dramatic decrease in the movements of face muscles. Figure 3.3(c) gives the frames taken from 19^{th} segment which produces the highest lie probability. During this segment, the subject opens her eyes widely and lifts her eyebrows. Interestingly, the subject never lifts her eyebrows in the rest of the video and she never opens her eyes as wide as she did in this segment which may explain why the model predicts the highest lie probability in this segment.

Change of lie probability over time for the other exemplar is given in Figure 3.4(a). We see that lie probabilities of segments mostly go beyond the 50% threshold and mean of them is above 50% at the end which means the video is correctly classified as *deceptive*. Not surprisingly, the subject's facial expressions changes a lot and his hand movements are too much throughout the entire video. Figure 3.4(b) presents the frames taken from the 8^{th} segment which produces the highest lie probability over the entire video. The subject raises his hands then moves them down and bends his neck to the right while talking during this segment. Similarly, the subject looks to the left in 12^{th} and 13^{th} segments both of which produce the highest lie probabilities after the 8^{th} segment. On the other hand, Figure 3.4(c) gives the frames taken from 10^{th} segment which produces the lowest lie probability. During this segment, the subject ceases talking, rotates his head a bit to the left while looking to a point in front of him without blinking and continues looking without any movement or talking. Considering that subject moves his hands a lot and has a lot of expressions throughout the entire video, the model interprets this deathly silent stationary state of the subject as not a sign of deception. It is presumably true since there is no facial expression

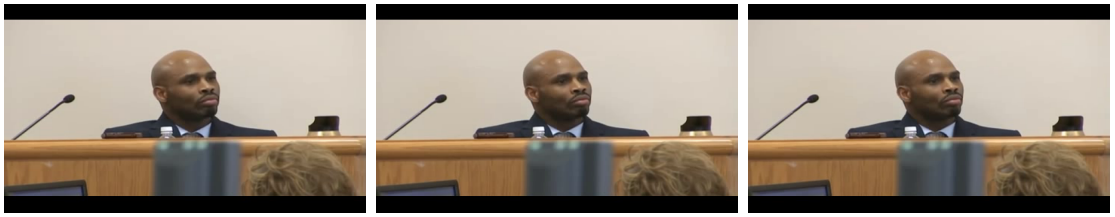
change and therefore no facial muscle movements during this segment.

Overall, the model learns to differentiate patterns that subjects display, and apparently the predicted extreme lie scores correspond to the patterns that are seen very rarely over the entire video.



(a)

(b)



(c)

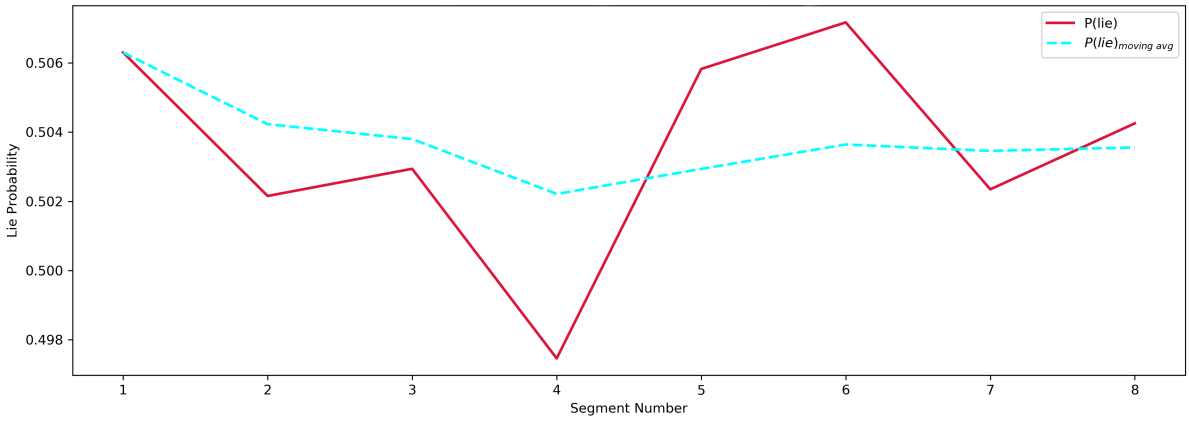
Figure 3.4: Qualitative analysis visualizations for the video *trial_lie_033* which is correctly classified as *deceptive* since the mean lie probability of segments is above 50%. (a) Change of lie probability over time, (b) 220., 230., 240. frames (left to right) from the 8th segment which produces the highest lie probability, (c) 280., 290., 300. frames (left to right) from the 10th segment which produces the lowest lie probability.

3.3.5.2 Failure Cases

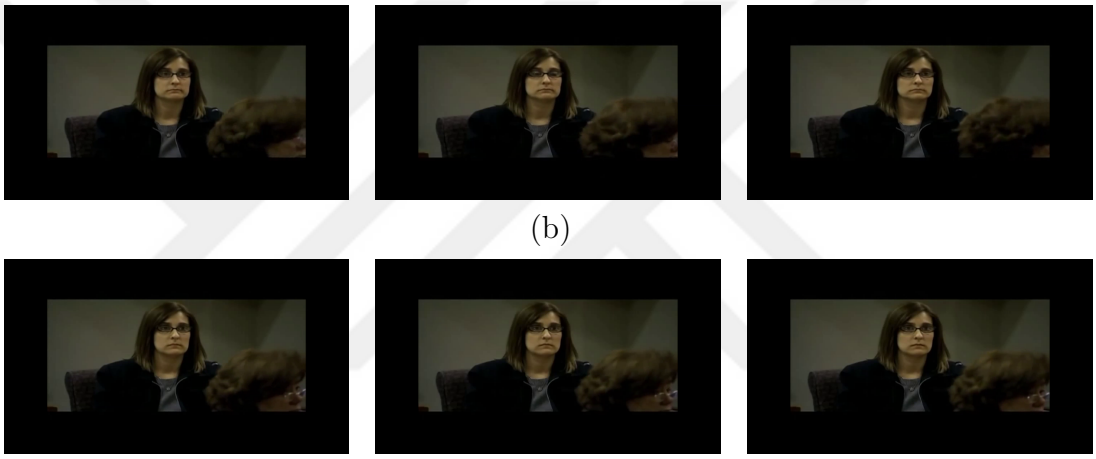
We conduct visual qualitative analysis on the failure cases and present the results of two misclassified samples (one from each class), which are found to be valuable to shed light on why the model fails. The corresponding sample frames are presented in Figure 3.5 and Figure 3.6.

When we inspect the visuals of the first failure case presented in Figure 3.5, it is noticeable that the lie probabilities are very close to 50% for all segments. In addition, the video length is too short (i.e., $8 \times 1.5 = 12$ seconds). Figure 3.5(b) and Figure 3.5(c) show the sample frames taken from the lowest (4^{th}) and the highest (6^{th}) lie probability segments, respectively. We realise that the subject speaks total of only 4 seconds in pieces out of 12 seconds. The reason for this is that there is another person who is asking questions to the subject and the subject is replying in short answers. In both 4^{th} and 6^{th} segments, subject silently stands, however, when this silence lasts longer, the model suspects that the subject is being deceptive. We believe that the shortage in speaking in such a short video can be the reason why model fails in this particular case. Note that the predictions are not significantly above or below 50% which indicates that the model is not highly confident in its classification. When we analyze other videos, we see that having obstacles momentarily in the subjects face (such as a moving text as shown in the top right of Figure 3.2(b) taken from the video trial_lie_007) causes to peaks in the lie probability scores. These sudden peaks increase the mean lie probability score drastically.

Similarly, visualizations of our quantitative analysis for another failure case for a deceptive video is given in Figure 3.6. Interestingly, the lowest lie probability is obtained when the subject looks downward similar to the case in Figure 3.3.b. This video is taken from a TV show and it contains significant scene changes as can be seen in Figure 3.6(c) as well as interrupts from the announcer of the TV program (this is why the last frame in Figure 3.6(c) is taken from the 437^{th} frame instead of the 383^{rd} frame). These scene changes cause to significant color differences in the subject's face and severe transitions from one frame to another,



(a)

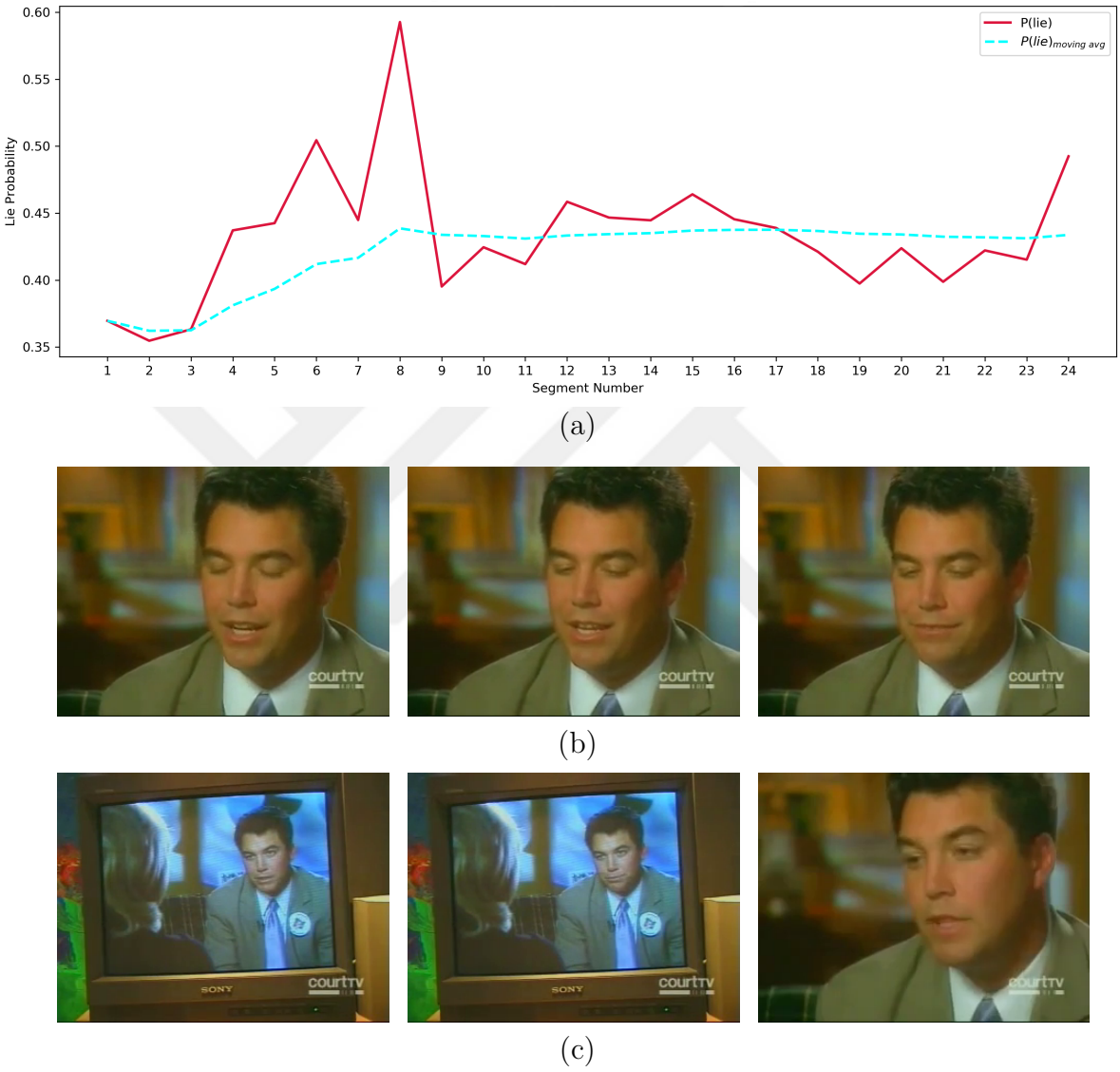


(b)

(c)

Figure 3.5: Qualitative analysis visualizations for the video *trial_truth_003* which is *misclassified* as *deceptive* since the mean lie probability of segments is above 50%. (a) Change of lie probability over time, (b) 100., 110., 120. frames (left to right) from the 4^{th} segment which produces the highest lie probability, (c) 160., 170., 180. frames (left to right) from the 6^{th} segment which produces the lowest lie probability.

which might confuse the model's interpretation. We also observe that in some videos (e.g., trial_lie_041) where the yaw rotation of the subject's face is in higher degrees, our facial normalization step causes defects in the normalized images (e.g., either left or right side of the face is not seen at all) throughout the entire video. This may possibly prevent the model learning informative features. In summary, having noise-free (e.g., without having subtitles and scene changes) and not occluded videos is vital for successful classification.



(a)

(b)

(c)

Figure 3.6: Qualitative analysis visualizations for the video *trial_lie_052* which is *misclassified* as *truthful* since the mean lie probability of segments is below 50%. (a) Change of lie probability over time, (b) 40., 50., 60. frames (left to right) from the 2nd segment which produces the lowest lie probability, (c) 363., 373., 437. frames (left to right) from the 8th segment which produces the highest lie probability.

Chapter 4

Conclusion

We have proposed a fully automated deception detection model in this thesis. Our model follows a multimodal approach, which uses both visual and acoustic features in spatiotemporal models. The visual focus of the proposed system have only been into the facial boundary of subjects. Therefore, the system first detects 68 facial landmarks and uses them to normalize facial images. Each normalized face image have tip of nose, centers of eyes and lips at the fixed locations in terms of pixels regardless of the subject's facial anatomy in the image. Extraction of visual features has been made through a deep convolutional neural network (CNN), i.e., 3D-ResNext [70]. Moreover, we have brought a new perspective into deception detection. To our knowledge, we are the first to consider subject's personality during deception detection. To detect the subject's personality features, our system again uses facial deep CNN features using [71]'s visual model. Obtained personality features are concatenated with the features of personality-unaware deception module after having passed through an MLP that contains two blocks of BN, 1×1 convolutional layer and ReLU activation function. The audio features have been extracted using the log mel-frequency spectrogram coefficients. The system uses coarse-to-fine grained audio features by concatenating 3 different spectrograms each of which uses different window sizes during their spectrogram calculations. In addition to visual and audio models, the proposed system also contains a fusion model which is a bi-directional gated recurrent units (BiGRU)

based recurrent neural network. The aim of fusion model is to boost the overall performance using visual and audio modalities.

In our thorough analyses, we have shown that integration of personality features improves the accuracy of visual models up to 20% (relative). We have shown that integrating the personality features into visual deception features not only boost the performance of our models but also the performance of other works in the literature. Furthermore, how to integrate these features into the models is found to be crucial in our experiments and it's best to pass them through 1×1 convolutional layers before using with deep models. Besides, our motivation for using the proposed spatiotemporal model, namely 2D-ResNext+BiGRU, in lieu of sole recurrent model is hold. According to our audio experiments, there's a 4.5% (relative) performance enhancement with 2D-ResNext+BiGRU compared to usual BiGRU model. Moreover, forming coarse-to-fine grained audio features is shown to be favourable by bringing additional 5.7% (relative) improvement in our experiments. Furthermore, when our personality integrated deception model fused with our audio model in the fusion model, we have achieved an accuracy of 64.66% which is 15.4% and 1.4% (both relative) performance enhancement compared to visual and audio modalities, respectively.

One thing to put emphasis on is that relying heavily on automated systems may be troublesome in real life applications. Specifically, ethical issues and biases can be observed in such systems. Thus, they should be carefully designed to mitigate any kind of unfair bias or prejudice towards a group or ethnicity if they will be employed in real life cases. True positive and false positive rates of such systems depend on the training data and these metrics could change drastically when the data distribution changes. Therefore, such automated systems can only act as an intermediate for decision makers and should only be used with utmost care as an auxiliary service.

Lastly, as for the limitations of our work, our system is only validated for high-stake scenarios where the consequence of telling a lie has severe penalties as in the case of court trials. More experiments are required to validate its robustness also in low-stake scenarios such as telling lies in daily life. Furthermore, an elaborate

study can be conducted to annotate Big Five personality scores of subjects in the dataset so that the personality module can be finetuned to extract possibly better personality features specifically tailored for the deception dataset. In addition, possible biases of our model is not thoroughly examined such as identity bias, e.g., skin reflectance and 3D facial geometry. It's necessary to set aside ample time for detecting possible biases of the system and revise the model accordingly as in the case of [108].

Bibliography

[1] B. Whaley, "Toward a general theory of deception," *The Journal of Strategic Studies*, vol. 5, no. 1, pp. 178–192, 1982.

[2] P. Ekman, "Telling lies: Clues to deceit in the marketplace, politics, and marriage.," 1991.

[3] E. E. Sweetser, "The definition of lie: An examination of the folk models underlying a semantic prototype," *Cultural Models in Language and Thought*, pp. 43–66, 1987.

[4] M. I. Handel, "Intelligence and deception," *The Journal of Strategic Studies*, vol. 5, no. 1, pp. 122–154, 1982.

[5] S. R. Kessler, A. C. Bandelli, P. E. Spector, W. C. Borman, C. E. Nelson, and L. M. Penney, "Re-examining machiavelli: A three-dimensional model of machiavellianism in the workplace," *Journal of Applied Social Psychology*, vol. 40, no. 8, pp. 1868–1896, 2010.

[6] A. Vrij, K. Edward, K. P. Roberts, and R. Bull, "Detecting deceit via analysis of verbal and nonverbal behavior," *Journal of Nonverbal Behavior*, vol. 24, no. 4, pp. 239–263, 2000.

[7] C. F. Bond Jr and B. M. DePaulo, "Accuracy of deception judgments," *Personality and Social Psychology Review*, vol. 10, no. 3, pp. 214–234, 2006.

[8] M. Zuckerman, B. M. DePaulo, and R. Rosenthal, "Verbal and nonverbal communication of deception," in *Advances in Experimental Social Psychology*, vol. 14, pp. 1–59, Elsevier, 1981.

[9] S. A. McCornack and M. R. Parks, "Deception detection and relationship development: The other side of trust," *Annals of the International Communication Association*, vol. 9, no. 1, pp. 377–389, 1986.

[10] E. L. Thorndike, "A constant error in psychological ratings," *Journal of Applied Psychology*, vol. 4, no. 1, pp. 25–29, 1920.

[11] S. Porter, L. Ten Brinke, and B. Wallace, "Secrets and lies: Involuntary leakage in deceptive facial expressions as a function of emotional intensity," *Journal of Nonverbal Behavior*, vol. 36, no. 1, pp. 23–37, 2012.

[12] L. H. Levy and R. D. Dugan, "A constant error approach to the study of dimensions of social perception.," *The Journal of Abnormal and Social Psychology*, vol. 61, no. 1, p. 21, 1960.

[13] S. J. Lachman and A. R. Bass, "A direct study of halo effect," *The Journal of Psychology*, vol. 119, no. 6, pp. 535–540, 1985.

[14] A. G. Greenwald and M. R. Banaji, "Implicit social cognition: attitudes, self-esteem, and stereotypes.," *Psychological Review*, vol. 102, no. 1, p. 4, 1995.

[15] R. Adelson, "Detecting deception," *APA Monitor on Psychology*, vol. 35, no. 7, pp. 70–73, 2004.

[16] A. Vrij and S. Mann, "Police use of nonverbal behavior as indicators of deception.," 2005.

[17] E. Svetieva and M. G. Frank, "Empathy, emotion dysregulation, and enhanced microexpression recognition ability," *Motivation and Emotion*, vol. 40, no. 2, pp. 309–320, 2016.

[18] C. Darwin, *The expression of the emotions in man and animals*. University of Chicago press, 1872.

[19] A. P. Association *et al.*, "The truth about lie detectors (aka polygraph tests)," *Retrieved June*, vol. 20, p. 2011, 2004.

[20] E. Rusconi and T. Mitchener-Nissen, "Prospects of functional magnetic resonance imaging as lie detector," *Frontiers in Human Neuroscience*, vol. 7, p. 594, 2013.

[21] C.-W. Hsu, C. Begliomini, T. Dall'Acqua, and G. Ganis, "The effect of mental countermeasures on neuroimaging-based concealed information tests," *Human Brain Mapping*, vol. 40, no. 10, pp. 2899–2916, 2019.

[22] P. Ekman and W. V. Friesen, "Nonverbal leakage and clues to deception," *Psychiatry*, vol. 32, no. 1, pp. 88–106, 1969.

[23] B. M. DePaulo, J. J. Lindsay, B. E. Malone, L. Muhlenbruck, K. Charlton, and H. Cooper, "Cues to deception.," *Psychological Bulletin*, vol. 129, no. 1, p. 74, 2003.

[24] S. Porter and L. ten Brinke, "The truth about lies: What works in detecting high-stakes deception?," *Legal and Criminological Psychology*, vol. 15, no. 1, pp. 57–75, 2010.

[25] S. Freud, "Fragment of an analysis of a case of hysteria (1905 [1901])," in *The Standard Edition of the Complete Psychological Works of Sigmund Freud, Volume VII (1901-1905): A Case of Hysteria, Three Essays on Sexuality and Other Works*, pp. 1–122, 1953.

[26] V. Pérez-Rosas, M. Abouelenien, R. Mihalcea, and M. Burzo, "Deception detection using real-life trial data," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pp. 59–66, 2015.

[27] B. M. DePaulo, D. A. Kashy, S. E. Kirkendol, M. M. Wyer, and J. A. Epstein, "Lying in everyday life.," *Journal of Personality and Social Psychology*, vol. 70, no. 5, p. 979, 1996.

[28] M. Lyons, N. Healy, and D. Bruno, "It takes one to know one: Relationship between lie detection and psychopathy," *Personality and individual differences*, vol. 55, no. 6, pp. 676–679, 2013.

[29] M. T. Le, *Language of low-stakes and high-stakes deception: Differences within individuals*. PhD thesis, University of British Columbia, 2016.

[30] D. P. Twitchell, J. F. Nunamaker, and J. K. Burgoon, "Using speech act profiling for deception detection," in *International Conference on Intelligence and Security Informatics*, pp. 403–410, Springer, 2004.

[31] T. Qin, J. K. Burgoon, J. P. Blair, and J. F. Nunamaker, "Modality effects in deception detection and applications in automatic-deception-detection," in *Proceedings of the Hawaii International Conference on System Sciences*, pp. 23b–23b, IEEE, 2005.

[32] L. Zhou and D. Zhang, "Following linguistic footprints: Automatic deception detection in online communication," *Communications of the ACM*, vol. 51, no. 9, pp. 119–122, 2008.

[33] T. O. Meservy, M. L. Jensen, J. Kruse, J. K. Burgoon, J. F. Nunamaker, D. P. Twitchell, G. Tsechpenakis, and D. N. Metaxas, "Deception detection through automatic, unobtrusive analysis of nonverbal behavior," *IEEE Intelligent Systems*, vol. 20, no. 5, pp. 36–43, 2005.

[34] S. Lu, G. Tsechpenakis, D. N. Metaxas, M. L. Jensen, and J. Kruse, "Blob analysis of the head and hands: A method for deception detection," in *Proceedings of the Hawaii International Conference on System Sciences*, pp. 20c–20c, IEEE, 2005.

[35] F. Xia, H. Wang, and J. Huang, "Deception detection via blob motion pattern analysis," in *International Conference on Affective Computing and Intelligent Interaction*, pp. 727–728, Springer, 2007.

[36] I. Nwogu, M. Frank, and V. Govindaraju, "An automated process for deceit detection," in *Biometric Technology for Human Identification VII*, vol. 7667, p. 76670R, International Society for Optics and Photonics, 2010.

[37] R. Wiseman, C. Watt, L. ten Brinke, S. Porter, S.-L. Couper, and C. Rankin, "The eyes don't have it: Lie detection and neuro-linguistic programming," *PloS one*, vol. 7, no. 7, p. e40259, 2012.

[38] I. Pavlidis and J. Levine, "Thermal facial screening for deception detection," in *Proceedings of the Second Joint Conference and the Fall Meeting of the Biomedical Engineering Society*, vol. 2, pp. 1143–1144, IEEE, 2002.

[39] P. Buddharaju, J. Dowdall, P. Tsiamyrtzis, D. Shastri, I. Pavlidis, and M. Frank, "Automatic thermal monitoring system (athemos) for deception detection," in *CVPR'05*, vol. 2, pp. 1179–vol, 2005.

[40] P. Tsiamyrtzis, J. Dowdall, D. Shastri, I. T. Pavlidis, M. Frank, and P. Ekman, "Imaging facial physiology for the detection of deceit," *International Journal of Computer Vision*, vol. 71, no. 2, pp. 197–214, 2007.

[41] Z. Zhang, V. Singh, T. E. Slowe, S. Tulyakov, and V. Govindaraju, "Real-time automatic deceit detection from involuntary facial expressions," in *Conference on Computer Vision and Pattern Recognition*, pp. 1–6, 2007.

[42] P. Ekman, "Facial action coding system (facs)," *A human face*, 2002.

[43] N. Michael, M. Dilsizian, D. Metaxas, and J. K. Burgoon, "Motion profiles for deception detection using visual cues," in *European Conference on Computer Vision*, pp. 462–475, Springer, 2010.

[44] N. E. Dunbar, M. L. Jensen, J. K. Burgoon, B. A. K. J. Robertson, K. M. Kelley, and L. Harvill, "A dyadic approach to the detection of deception," in *Quality in Government and Business Symposium*, p. 17, Citeseer, 2011.

[45] S. Brault, B. Bideau, R. Kulpa, and C. M. Craig, "Detecting deception in movement: the case of the side-step in rugby," *PloS one*, vol. 7, no. 6, p. e37494, 2012.

[46] R. Poppe, S. Van der Zee, P. J. Taylor, R. Anderson, and R. C. Veltkamp, "Mining bodily cues to deception," in *Proceedings of the Rapid Screening Technologies, Deception Detection and Credibility Assessment Symposium at the Hawaii International Conference on System Sciences (HICSS)*, 2015.

[47] H. Bouma, G. Burghouts, R. den Hollander, S. Van Der Zee, J. Baan, J.-M. ten Hove, S. van Diepen, P. van den Haak, and J. van Rest, "Measuring cues for stand-off deception detection based on full-body nonverbal features in body-worn cameras," in *Optics and Photonics for Counterterrorism, Crime Fighting, and Defence XII*, vol. 9995, p. 99950N, International Society for Optics and Photonics, 2016.

[48] S. Van Der Zee, R. Poppe, P. J. Taylor, and R. Anderson, "To freeze or not to freeze: A culture-sensitive motion capture approach to detecting deceit," *PloS one*, vol. 14, no. 4, p. e0215000, 2019.

[49] T. Fornaciari and M. Poesio, "Automatic deception detection in italian court cases," *Artificial Intelligence and Law*, vol. 21, no. 3, pp. 303–340, 2013.

[50] V. Pérez-Rosas and R. Mihalcea, "Cross-cultural deception detection," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 440–445, 2014.

[51] M. Abouelenien, V. Pérez-Rosas, R. Mihalcea, and M. Burzo, "Deception detection using a multimodal approach," in *Proceedings of the 16th international conference on multimodal interaction*, pp. 58–65, 2014.

[52] M. Jaiswal, S. Tabibu, and R. Bajpai, "The truth and nothing but the truth: Multimodal analysis for deception detection," in *16th International Conference on Data Mining Workshops (ICDMW)*, pp. 938–943, 2016.

[53] M. R. Morales, S. Scherer, and R. Levitan, "Openmm: An open-source multimodal feature extraction tool," in *Proc. Interspeech 2017*, pp. 3354–3358, 2017.

[54] Z. Wu, B. Singh, L. Davis, and V. Subrahmanian, "Deception detection in videos," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[55] U. M. Sen, V. Perez-Rosas, B. Yanikoglu, M. Abouelenien, M. Burzo, and R. Mihalcea, "Multimodal deception detection using real-life trial data," *IEEE Transactions on Affective Computing*, 2020.

[56] D. Avola, L. Cinque, M. De Marsico, A. Fagioli, and G. L. Foresti, "Lietome: Preliminary study on hand gestures for deception detection via fisher-lstm," *Pattern Recognition Letters*, vol. 138, pp. 455–461, 2020.

[57] H. E. Cattell and A. D. Mead, "The sixteen personality factor questionnaire (16pf).," 2008.

[58] R. R. McCrae and O. P. John, "An introduction to the five-factor model and its applications," *Journal of Personality*, vol. 60, no. 2, pp. 175–215, 1992.

[59] M. C. Ashton and K. Lee, "Empirical, theoretical, and practical advantages of the hexaco model of personality structure," *Personality and Social Psychology Review*, vol. 11, no. 2, pp. 150–166, 2007.

[60] R. E. Riggio and H. S. Friedman, "Individual differences and cues to deception.," *Journal of Personality and Social Psychology*, vol. 45, no. 4, p. 899, 1983.

[61] R. E. Riggio, C. Salinas, and J. Tucker, "Personality and deception ability," *Personality and Individual Differences*, vol. 9, no. 1, pp. 189–191, 1988.

[62] G. H. Gudjonsson and S. Young, "Personality and deception. are suggestibility, compliance and acquiescence related to socially desirable responding?," *Personality and Individual Differences*, vol. 50, no. 2, pp. 192–195, 2011.

[63] E. A. Giammarco, B. Atkinson, H. M. Baughman, L. Veselka, and P. A. Vernon, "The relation between antisocial personality and the perceived ability to deceive," *Personality and Individual Differences*, vol. 54, no. 2, pp. 246–250, 2013.

[64] S. Porter, L. ten Brinke, A. Baker, and B. Wallace, "Would i lie to you?"leakage" in deceptive facial expressions relates to psychopathy and emotional intelligence," *Personality and Individual Differences*, vol. 51, no. 2, pp. 133–137, 2011.

[65] J. Sarzyńska, M. Falkiewicz, M. Riegel, J. Babula, D. S. Margulies, E. Necka, A. Grabowska, and I. Szatkowska, "More intelligent extraverts are more likely to deceive," *PloS one*, vol. 12, no. 4, p. e0176591, 2017.

[66] J. A. Hall, "Gender effects in decoding nonverbal cues.," *Psychological Bulletin*, vol. 85, no. 4, p. 845, 1978.

[67] H. S. Friedman, L. M. Prince, R. E. Riggio, and M. R. DiMatteo, "Understanding and assessing nonverbal expressiveness: the affective communication test.," *Journal of Personality and Social Psychology*, vol. 39, no. 2, p. 333, 1980.

[68] D. L. Paulhus and K. M. Williams, "The dark triad of personality: Narcissism, machiavellianism, and psychopathy," *Journal of Research in Personality*, vol. 36, no. 6, pp. 556–563, 2002.

[69] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L.-P. Morency, "Openface 2.0: Facial behavior analysis toolkit," in *IEEE international conference on automatic face & gesture recognition (FG 2018)*, pp. 59–66, IEEE, 2018.

[70] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6546–6555, 2018.

[71] D. Giritlioğlu, B. Mandira, S. F. Yilmaz, C. U. Ertenli, B. F. Akgür, M. Kınıklıoğlu, A. G. Kurt, E. Mutlu, Ş. C. Gürel, and H. Dibeklioğlu, "Multimodal analysis of personality traits on videos of self-presentation and induced behavior," *Journal on Multimodal User Interfaces*, pp. 1–22, 2020.

[72] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th Python in Science Conference*, vol. 8, pp. 18–25, Citeseer, 2015.

[73] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017.

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

[75] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[76] J.-I. Biel, O. Aran, and D. Gatica-Perez, "You are known by how you vlog: Personality impressions and nonverbal behavior in youtube," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 5, 2011.

[77] O. Aran and D. Gatica-Perez, "Cross-domain personality prediction: from video blogs to small group meetings," in *Proceedings of the International Conference on Multimodal Interaction*, pp. 127–130, 2013.

[78] O. Celiktutan and H. Gunes, "Continuous prediction of perceived traits and social dimensions in space and time," in *International Conference on Image Processing (ICIP)*, pp. 4196–4200, IEEE, 2014.

[79] Y. Güçlütürk, U. Güçlü, X. Baro, H. J. Escalante, I. Guyon, S. Escalera, M. A. Van Gerven, and R. Van Lier, "Multimodal first impression analysis with deep residual networks," *IEEE Transactions on Affective Computing*, vol. 9, no. 3, pp. 316–329, 2017.

[80] H. Kaya, F. Gurpinar, and A. Ali Salah, "Multi-modal score fusion and decision trees for explainable automatic job candidate screening from video cvs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–9, 2017.

[81] X.-S. Wei, C.-L. Zhang, H. Zhang, and J. Wu, "Deep bimodal regression of apparent personality traits from short video sequences," *IEEE Transactions on Affective Computing*, vol. 9, no. 3, pp. 303–315, 2017.

[82] H. J. Escalante, H. Kaya, A. A. Salah, S. Escalera, Y. Güç, U. Güçlü, X. Baró, I. Guyon, J. C. Jacques, M. Madadi, *et al.*, "Modeling, recognizing, and explaining apparent personality from videos," *IEEE Transactions on Affective Computing*, 2020.

[83] S. Yan, D. Huang, and M. Soleymani, "Mitigating biases in multimodal personality assessment," in *Proceedings of the International Conference on Multimodal Interaction*, pp. 361–369, 2020.

[84] J. C. J. Junior, A. Lapedriza, C. Palmero, X. Baró, and S. Escalera, "Person perception biases exposed: Revisiting the first impressions dataset.," in *WACV (Workshops)*, pp. 13–21, 2021.

[85] V. Ponce-López, B. Chen, M. Oliu, C. Corneanu, A. Clapés, I. Guyon, X. Baró, H. J. Escalante, and S. Escalera, "Chalearn lap 2016: First round challenge on first impressions-dataset and results," in *European Conference on Computer Vision*, pp. 400–418, Springer, 2016.

[86] L. Zhang, Z. Tan, J. Song, J. Chen, C. Bao, and K. Ma, "Scan: A scalable neural networks framework towards compact and efficient models," *arXiv preprint arXiv:1906.03951*, 2019.

[87] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," *arXiv preprint arXiv:1503.00075*, 2015.

[88] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4945–4949, IEEE, 2016.

[89] Z. Zhao, Q. Yang, D. Cai, X. He, and Y. Zhuang, "Video question answering via hierarchical spatio-temporal attention networks.," in *IJCAI*, pp. 3518–3524, 2017.

[90] J. Xu, T. Yao, Y. Zhang, and T. Mei, "Learning multimodal attention lstm networks for video captioning," in *Proceedings of the International Conference on Multimedia*, pp. 537–545, 2017.

[91] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, pp. 8026–8037, 2019.

[92] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *International conference on machine learning*, pp. 1139–1147, PMLR, 2013.

[93] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 648–656, 2015.

[94] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 6450–6459, 2018.

[95] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

[96] M. Monfort, A. Andonian, B. Zhou, K. Ramakrishnan, S. A. Bargal, T. Yan, L. Brown, Q. Fan, D. Gutfreund, C. Vondrick, *et al.*, "Moments in time dataset: one million videos for event understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 502–508, 2019.

[97] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[98] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[99] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[100] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using

rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[101] 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.

[102] S. Hochreiter, "Untersuchungen zu dynamischen neuronalen netzen," *Diploma, Technische Universität München*, vol. 91, no. 1, 1991.

[103] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[104] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *International Conference on Machine Learning*, pp. 1310–1318, PMLR, 2013.

[105] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: a review," *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[106] B. Zhao, H. Lu, S. Chen, J. Liu, and D. Wu, "Convolutional neural networks for time series classification," *Journal of Systems Engineering and Electronics*, vol. 28, no. 1, pp. 162–169, 2017.

[107] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, "Scikit-learn: Machine learning in python," *the Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[108] L. M. Ngo, W. Wang, B. Mandira, S. Karaoglu, H. Bouma, H. Dibeklioglu, and T. Gevers, "Identity unbiased deception detection by 2d-to-3d face reconstruction," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 145–154, 2021.