Deepfake Video Detection Based on Spatial, Spectral, and Temporal Inconsistencies Using Multimodal Deep Learning

John K. Lewis
Florida Southern College
jklewis99@gmail.com

Imad Eddine Toubal
University of Missouri
itoubal@mail.missouri.edu

Helen Chen
University of Maryland, College Park
hchen104@gmail.com

Vishal Sandesera
IQT Labs
vsandesara@iqt.org

Michael Lomnitz
IQT Labs
mllomnitz@gmail.com

Zigfried Hampel-Arias
IQT Labs
zhampel@iqt.org

Calyam Prasad
University of Missouri
calyamp@missouri.edu

Kannappan Palaniappan
University of Missouri
pal@missouri.edu

2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR) | 978-1-7281-8243-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/AIPR50011.2020.9425167

***Abstract*—Authentication of digital media has become an ever-pressing necessity for modern society. Since the introduction of Generative Adversarial Networks (GANs), synthetic media has become increasingly difficult to identify. Synthetic videos that contain altered faces and/or voices of a person are known as deepfakes and threaten trust and privacy in digital media. Deepfakes can be weaponized for political advantage, slander, and to undermine the reputation of public figures. Despite imperfections of deepfakes, people struggle to distinguish between authentic and manipulated images and videos. Consequently, it is important to have automated systems that accurately and efficiently classify the validity of digital content. Many recent deepfake detection methods use single frames of video and focus on the spatial information in the image to infer the authenticity of the video. Some promising approaches exploit the temporal inconsistencies of manipulated videos; however, research primarily focuses on spatial features. We propose a hybrid deep learning approach that uses spatial, spectral, and temporal content that is coupled in a consistent way to differentiate real and fake videos. We show that the Discrete Cosine transform can improve deepfake detection by capturing spectral features of individual frames. In this work, we build a multimodal network that explores new features to detect deepfake videos, achieving 61.95% accuracy on the Facebook Deepfake Detection Challenge (DFDC) dataset.**

***Index Terms*—deepfake detection, deep learning, multi-modal, computer vision**

I. Introduction

Recent advances in synthetic media have posed a great threat for individual privacy, trust, and transparency of media. Resources that were once used for harmless activities on Snapchat and Instagram can now be used to manipulate the words of politicians [1] or feature non-consenting individuals in pornographic content [2]. While the production cost to create a photo-realistic product was once very expensive, advancements in artificial intelligence and computer vision have brought powerful video editing to the fingertips of curious individuals. When these technologies are used to swap the image of one person onto the body of another or to generate photo-realistic facial expressions and realistic dialogue, they create what are known as deepfakes. Generative Adversarial Networks (GANs) and Variational Autoencoders are the primary technologies that make these deepfakes possible. Though some users hope to do some good with this technology [3], the most prominent platform of deepfake appearances comes in the form of pornography [2], and the nefarious uses do not stop there. Fortunately, governments have become more aware of the dangers of this synthetic media on democracy and the spread of misinformation and the violations of civil liberties, but creating a deepfake is arguably easier than accurately classifying the authenticity of a video. As more tools become available and sophisticated for creating deepfakes, it is important to move deepfake detection tools in the same direction.

The recent uptick [2] in synthetic media online poses a threat of authentic content too. For example, amid health concerns and a lack of public appearances, the Gabonese President appeared in an address that appeared synthetic [4]. Following this video, a military coup was launched, some citing the presumed manipulated video as inspiration. However, the video has been evaluated by deepfake detection algorithms [4], and it has been classified as authentic. Moreover, some politicians have used the existence of deepfakes to discredit authentic content to avoid persecution and punishment for their actions [5].

As the technology has improved, evidence suggests that a human guess is not marginally better than a coin flip [6]. Though the artifacts created by some deepfake methods are sometimes easier to identify than others, these artifacts are becoming increasingly difficult for humans to detect [6]. Fortunately, some manipulation indications that the naked eye cannot detect can be identified by a computer [7]. Nonetheless,

the difficulty of detection will continue to increase, and it is important that deepfake detection methods stay up-to-date with successful deepfake generation techniques in order to achieve successful classification of a video's authenticity.

Our model is designed to achieve success in this generalization. In this work, we propose a multimodal network that combines recent promising approaches in deepfake detection to accurately classify content as real or fake. Our results are based on a subset of data sampled from the Facebook Deepfake Detection Challenge [8]. We process visual neural features, visual spectral features, audio spectral features, and utilize transfer learning with XceptionNet [9], LipNet [10], and DeepSpeech2 [11]. We introduce a new method of spectral feature analysis with the Discrete Cosine transform, and show that it can be effective in improving deepfake detection. Our model is segmented into multiple sub-networks, which we test independently to evaluate each sub-network's significance. We consider speed regarding our methods to process and extract meaningful features, and accuracy in evaluating the success of our model. We compare our model to the winner of the Deepfake Detection Challenge on the Deepfake Detection Challenge Dataset. Our code has been made publicly available [12].

II. Related Work

A. Face Manipulation and Generative Adversarial Networks

Ever since the introduction of Generative Adversarial Networks (GANs) [13], researchers in deep learning have increasingly focused on this area of research; most notably, in computer vision applications [14]. GANs introduced a method of competitively training two separate neural networks in which the goal of one network, the generator, is to create synthetic data that causes inaccurate classification by the other network, the discriminator. Thanks to the novel architecture of GANs, significant advancements have been made in areas like semantic segmentation [15], [16], [4], style transfer [17], [18], [19], [20], realistic image generation [21], [22], [23], image super-resolution [24], [25], [26], and image completion [27], [28], [29]. Most notably, though, these methods can be used to generate realistic deepfakes.

Korshunov et al. [30] utilize advancements in style transfer methods with generative models (GANs [13] and Convolutional Autoencoders [18]) and incorporate their own network with a multi-image style loss for face-swapping. They also match lighting conditions of the target image and the synthetically produced image. Their networks are able to produce face-swapped images in near-real to real time. Averbuch-Elor et al. [15] approaches face-swapping differently by requiring only one image of the target subject. Additionally, this method is effective in animating videos of a subject. The First Order Motion Model [31] introduced a similar method of single target animation but extends beyond just facial manipulation to full body reenactment. Using a Taylor expansion approximation, they define the key points for movement and apprehend to those key points local affine transformations which, paired together with the image source, are sent through a generator network to produce the animated video. To improve their model, they incorporate an occlusion-aware generator that infers hidden elements based on image context. Face2Face [32] introduces real-time facial reenactment using a single target video. To accomplish this, target and source actors are tracked with a pixel-accurate photometric energy minimization technique. Then, the expressions of the source actor are rendered on the target actor using subspace deformation transfer as introduced by [33] to build a modified template of the target actor which can then be rendered on the original image. The mouth of the subject is also refitted using a similarity metric and frame-to-cluster matching strategy and is temporally smoothed by locating an accurate mouth shape between the target frame and last rendered frame. The maintenance of target mouth shape helps improve the photo-realism of the video sequence.

Because of the variety of techniques used in deepfake generation, developing a network to identify the distinguishing features can be difficult to generalize without data that consists of these many techniques. We recognize the unique qualities of deepfake generation methods and train our model on a dataset with deepfake videos created using various generation methods.

B. Audio Manipulation Methods

The production of seamlessly integrated, separate source video and audio streams is a relevant challenge for researchers focusing on audio-driven facial reenactment. Though this subject has been of interest for the last few decades, recent advances have become most relevant. Suwajanakorn et al. [34] trained a recurrent neural network on the mouth shape of raw audio on President Obama to recreate a synthetic, photo-realistic videos of him based on the audio input of his voice. The Speech2Vid [35] model is used to render audio on a target subject for both still photos and video sequences without requiring the target to be part of the training data. Vougioukas et al. [36] expand on the usage of still images and subject independence by creating a realistic video of a talking head based on audio input. They use a temporal GAN with a frame discriminator and a temporal discriminator to add natural facial expressions and blinks to the subject image. The same researchers expanded on their own work in [36] by adding a second temporal discriminator to ensure both audio-visual correspondence and facial expressions are captured in the video. Neural Voice Puppetry, as introduced in [37], uses a deep neural network to produce photo-realistic synthesizing of audio onto any target subject. They develop a latent audio expression space to generalize the expression of lips for given phonemes and speech style, allowing for the rendering of photo-realistic videos using input speech or text.

Because audio and visual elements are often separated in these generative models, there are likely to be subtle misalignments in the audio and visual features, whether in the streams themselves or in mismatches between phonemes and visemes. Often, the movement of the mouth does not reflect natural human mouth motion, mostly due to frame by

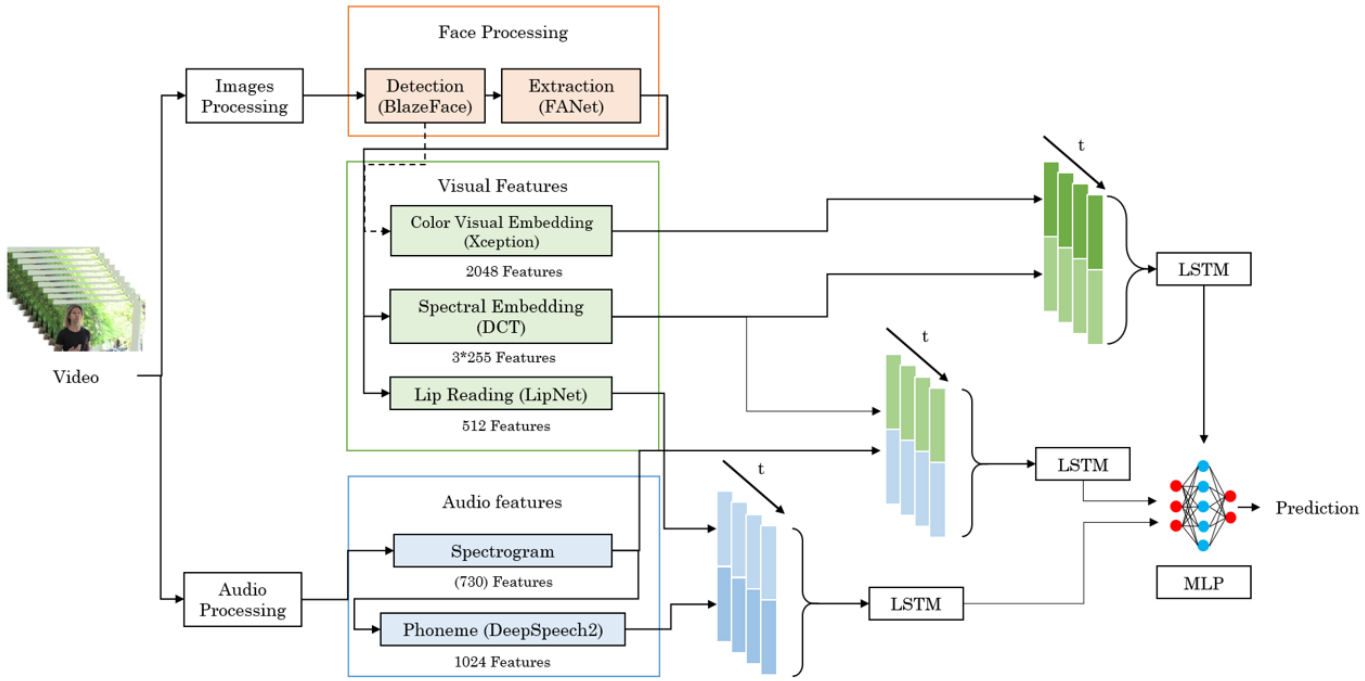


Fig. 1: Flowchart of the proposed multimodal network. Each video goes through a series of pre-processing steps and pre-trained models. We use these models' final layer of features concatenated with spectrogram features and visual spectral features in a complex fusion into multiple Long Short Term Memory Networks and through a final classification layer to determine whether a video is real or fake.

frame synthesis. These methods inspired consideration for our LipSpeech model, as defined in Section IV-C.

C. Deepfake Detection Strategies

It has been noted that many deepfakes possess notable imperfections as a result of harsh shadows [4], face occlusion [34], and inaccurate geometry estimates [38]. Some imperfections that result from deepfakes are heterochromia, which is the different coloring of irises, or specular reflection [38]. Deepfakes can be exposed based on inconsistencies of the position of central facial landmarks in relation to a face's outer contour landmarks [39]. Additionally, it may be common for deepfakes to show patterns of infrequent blinking [40]. One imperfection noted by Agarwal et al. [41] is related to the inconsistencies between the visemes and phonemes in videos involving the manipulation of the mouth. Li and Lyu [42] propose a faster and less data-dependent algorithm that focuses on the artifacts, specifically resolution inconsistencies, caused by the affine transformation when a face is rendered on top of another. However, some of these spatial artifacts may be indistinguishable from artifacts caused by video compression [6], which is why it is important to continue researching some other indications of synthetic media that may not be in the spatial domain. A recent method proposed by Durall et al. [7] achieves good results by identifying the differences between the frequency domain analysis of real and deepfake subjects. The success was heavily favored toward higher-resolution sources, however, and, as mentioned, manipulated media does not need to be in high definition to be effective. We approach our feature analysis similarly, but we use the Discrete Cosine transform instead of the Fourier transform.

III. Data

In 2019, Facebook introduced a new dataset for the Facebook Deepfake Detection Challenge [8]. This dataset consists of over 128 thousand videos, 83% of which are fake videos, with 3,426 paid and consenting actors that, in its raw form, totaled 25 terabytes of data. This data involved actors of different races, age, gender, and mannerisms and included distractors, occlusions, and variations in movements, frame rate, audio sample rates, orientation, number of actors per video, and sound environments. Additionally, the methods used to create the deepfakes varied and were not labeled.

To train and test our model, we sub-sampled 5,000 videos from this dataset with balanced labels: 50% real videos and 50% fake videos. We split the data into training and testing data with a 90%:10% split, respectively, resulting in 4,500 videos for training and 500 videos for testing. Each of these videos were approximately 10 seconds in length.

IV. Methods

A. Multimodal System Overview

In order to accurately distribute the relevant data to each model in our network, we must pre-process our input video into 2 forms. Our model, NOLANet, as shown in Fig. 1, takes

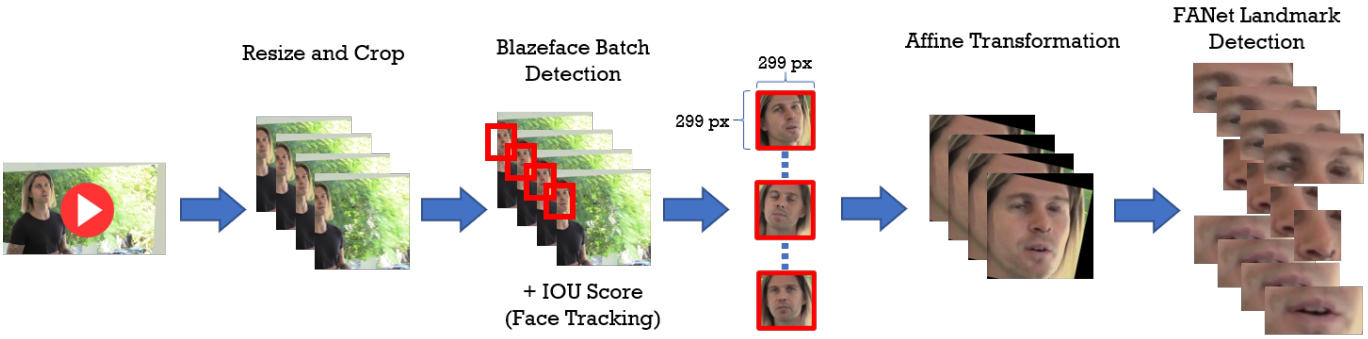


Fig. 2: Extraction of the key landmarks from the face

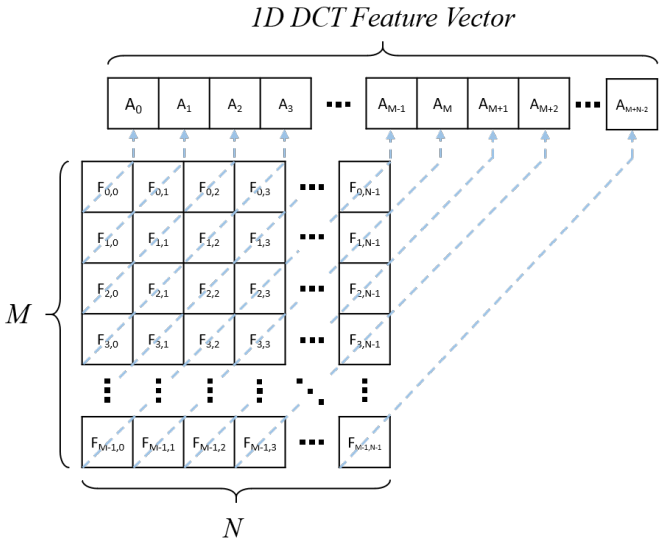
an input video at a normalized frame rate of 30 frames per second and extracts each individual frame. NOLANet takes the audio from the input video in its raw form. These features are concatenated to form the input features into multiple Long Short Term Memory (LSTM) networks, which serve to encompass the visually embedded features and the aligned audio and visual features. We detail these methods in the following sections.

B. Feature Extraction

1) Face Detection: The primary region of focus for creating deepfakes is the area encompassing the head and face, so for deepfake detection, all features outside of this region can be disregarded. We use a pretrained BlazeFace model [43] to extract this face region. Each face region is cropped and resized to 128x128 pixels, maintaining aspect ratio with a buffer area around the face the make it square. Because some videos contain more than one face, we include an Intersection Over Union (IOU) score using the bounding boxes around the face to ensure the same face is detected in each frame. If the IOU score is 0, we assume the face is a new one and this new face is disregarded. Only faces with an IOU score greater than 0 are considered (except for the first frame). For each video, every face is saved with a label of its frame index, resized to 299x299 pixels.

2) Facial Landmark Extraction: After faces are detected and saved, we use the Face Alignment Network (FANet) [44], [45] to detect the facial landmarks. Following, we perform an affine transformation to normalize the position of landmarks. Three bounding boxes are extracted: one surrounding the eyes and eyebrows, one surrounding the nose, and one surrounding the mouth. Using the center of these landmarks for each landmark, we create a buffer in each x and y direction, establishing a 2:1 ratio for the eyes/eyebrows and the mouth and 1:1 ratio for the nose. This entire procedure, in tandem with face detection, is shown in Fig. 2.

3) Discrete Cosine Transform: Fig. 3 details our approach for analysis in the frequency domain (spectral features) of landmark images. We adopt a similar approach to the method used in [7] to extract spectral features, but we choose to utilize the Discrete Cosine transform (DCT) instead of the Discrete Fourier transform (DFT). The intuition behind this decision



$$A_k = \frac{1}{k-\delta} \sum_{i=\gamma}^{k-\delta} F(k-i, i)$$

$$\gamma = \begin{cases} 0, & if\ u < M \\ u - M + 1, & else \end{cases}$$

$$\delta = \begin{cases} 0, & if\ u < N \\ u - N + 1, & else \end{cases}$$

Fig. 3: Anti-diagonal Average

is based on the identifying components in the imaginary parts of the DFT. The DFT keeps all the image information in a complex map (that can be split into a real magnitude map and a real phase map), but most of the relevant visual information that can be used to reconstruct the original image is present in the phase map. DCT, however, keeps all the spatial information in a single real map that provides a more meaningful aggregation and guarantees better preservation of information. The DCT is calculated at each landmark bounding

box with a minimum dimension of 128 pixels. For the deep learning steps that use this DCT information, it is required that these features be one-dimensional instead of two-dimensional. To reduce this dimensionality, we calculate the anti-diagonal average of the DCT map 3. To visualize this DCT embedding, we use a DCT-Spectrogram shown in Fig. 4.

4) Spectrogram: The first step for audio processing is the generation of spectrogram features. To ensure the audio features temporally align with visual features, the sampling windows must be the same, that is, the window size for the spectrogram input sample must match the time each frame is present in the video. The window size, w, in milliseconds, is calculated as follows:

$$w = \frac{1}{FPS}$$

For our training examples, the most prevalent frame-rate was 30 FPS. We use this frame-rate as the basis for calculating the window size $w = 33ms$. If we assume an input discrete audio signal to the Short-Time Fourier transform to be

$$\vec{s_i} = [s_{i,1}, s_{i,2}, ...s_{i,k}]$$

where the number of samples $k = w \times f_s$ and f_s being the audio sampling rate, then the Short Time Fourier transform of s_i:

$$\vec{S_i} = STFT(\vec{s_i}) = [S_{i,1}, S_{i,1}, ..., S_{i,1}]$$

is calculated as an audio feature vector of the given window.

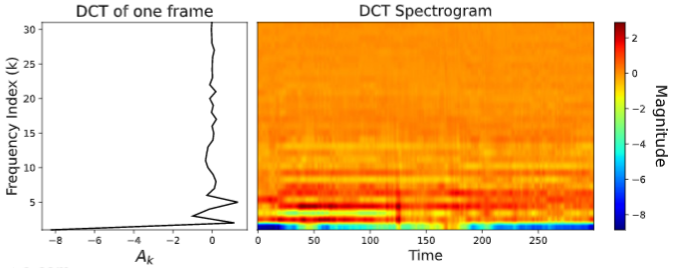


Fig. 4: Example DCT Spectrogram.

5) Transfer Learning:

XceptionNet: XceptionNet [9] has proven its efficiency and accuracy in image classification tasks. More specifically it has also been used successfully for deepfake detection [6]. We use XceptionNet pre-trained on the ImageNet [46] dataset as a feature extractor for the individual frames of each face. Let's assume a sequence of images $\boldsymbol{I} = \{I_1, I_2, ..., I_q\}$ of length q, with each image $I_i \in \mathbb{R}^{299 \times 299 \times 3}$. The output of this image sequence in XceptionNet is a sequence of features $\boldsymbol{X} = \{\vec{X_1}, \vec{X_2}, ..., \vec{X_q}\}$ with $X_i \in \mathbb{R}^{2048}$. We process the XceptionNet feature sequence in a 2-layer LSTM network that will be thoroughly described later in this paper. For our experiments, we choose the input sequence size $q = 30$, which is equal to one second of a video.

LipNet: LipNet [10] is used as a lip reading network. This network outputs a transcript by decoding text from the movement of a speaker's mouth. Similar to XceptionNet, we use this network on our face image sequence $\boldsymbol{I}$ as a feature extractor to produce an output sequence $\boldsymbol{L} = \{L_1, L_2, ..., L_q\}$ with $L_i \in \mathbb{R}^{512}$. We hypothesize that the output of this network contains the necessary lip movement information that we can further use in tandem with our audio features to detect possible discrepancies.

DeepSpeech2: Using a pre-trained implementation of DeepSpeech2 ([11], [47]) proved challenging because of its input size. The model is based on a spectrogram that contains a window size of 20 ms and a window stride of 10 ms. All other temporal input data in our models is a window size of 33 ms (1 frame of video at 30 fps), so aligning the output features of DeepSpeech2 and all other features of our model cannot be done simply. To overcome this challenge, we prepare one LSTM for all temporally-aligned visual features and one LSTM for the DeepSpeech2 features. We achieve alignment with sequentially appended inputs for each LSTM. The visual LSTM uses thirty sequential frames, which achieves a time length of 1 second, and the DeepSpeech2 LSTM uses 50 sequential input features, which achieves a matching time length of 1 second. These output features are then fused as input into another LSTM that outputs a binary classification of "real" or "fake".

6) Feature Alignment: When working with a multimodal network that processes both visual and audio information sequentially, it is important to take into consideration the video-audio temporal consistency. For this reason, we propose an alignment technique (Fig. 5) that ensures a near-perfect temporal alignment. This ensures that our methods will use any misalignment in an input video as a potential indication of synthetically generated media.

C. Sub-Networks

LipSpeech: Inspired by the work in [41] and [48], we design a sub-network to evaluate the alignment of visual data and audio data streams, as shown in Fig. 7. Specifically, we compare the output of text predictors, LipNet and DeepSpeech2. LipNet [10] is designed to predict text based on a lip reading model, and DeepSpeech2 [11] is designed to predict speech based on the audio. The idea of this sub-network is based on the assumption that these two pre-trained models are inherently aware of visemes and phonemes, respectively. The inputs to each of these models are sequences of 1 second, resulting in an output feature vector representing one second of translated text based on the audio and one second of translated text based on the mouth movements. We extract the feature vectors before the text translation from each of these models and feed these as the input into an LSTM (Fig. 6).

FourierNet: The spectral feature-focused sub-network of NOLANet evaluates the significance of the spectral information of visual features as calculated by the DCT of the landmarks and the spectral information of the audio with the Short Time Fourier transform of the audio sample. Each of

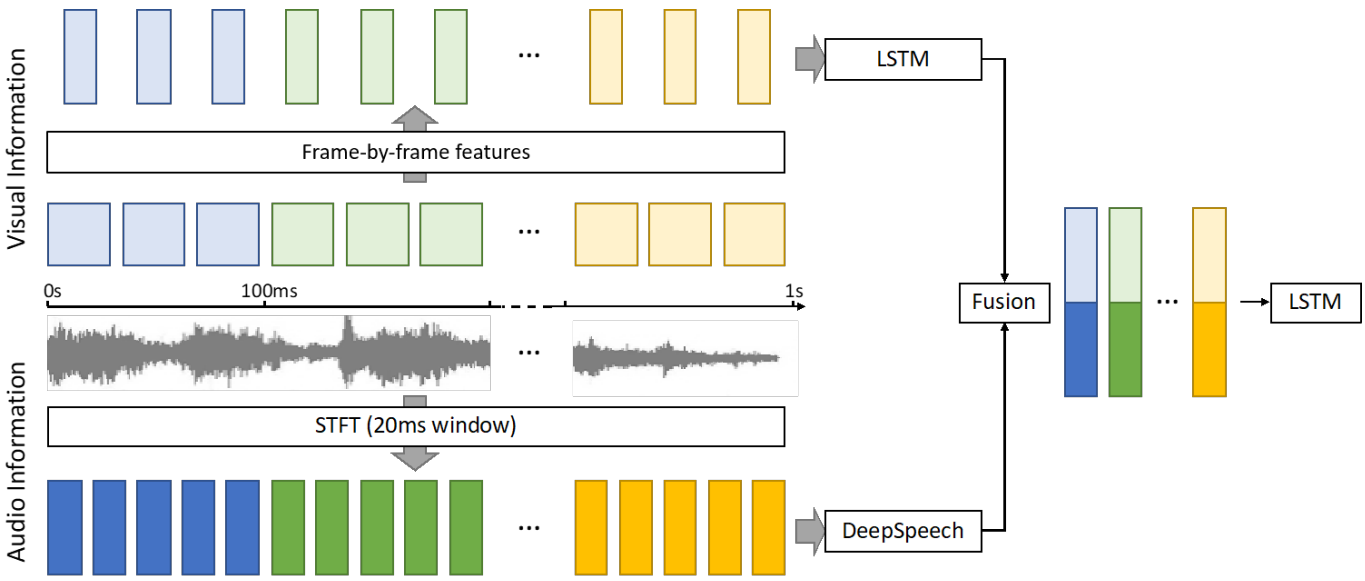


Fig. 5: Aligning visual and audio features temporally

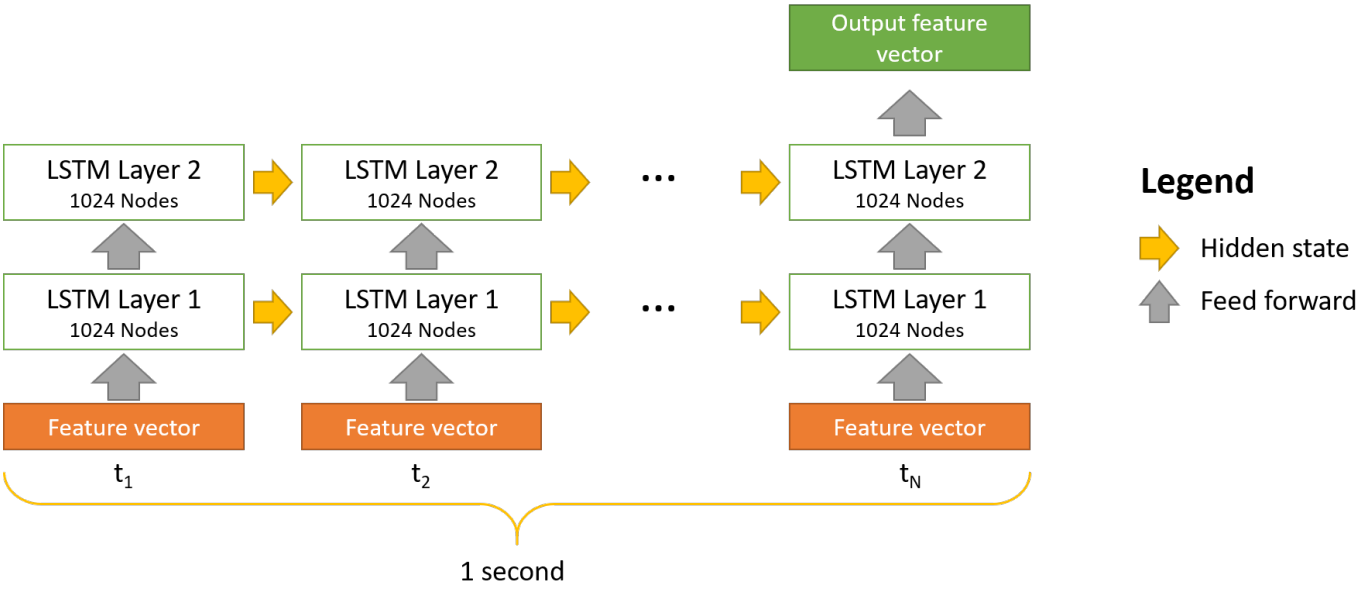


Fig. 6: Architecture of the LSTM used in each sub-network.

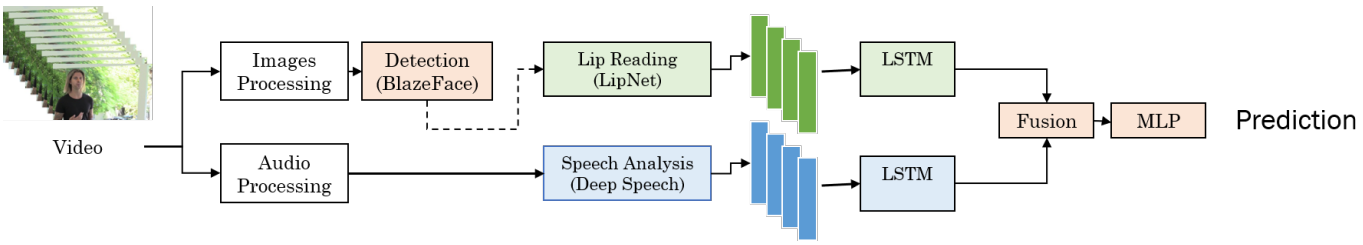


Fig. 7: The pipeline of the LipSpeech sub-network. A mouth sequence of 1 second is input in the LipNet model, and a spectrogram sequence of 1 second is input in the DeepSpeech2 model. The output features of these models are sent through individual LSTMs and fused together as input to a multilayer perceptron

these inputs span temporally for 1 second. These features are concatenated, giving them a total feature size of 1495, and sent through an LSTM (Fig 6). The entire network of FourierNet is outlined in Fig 8.

VSNet: The image-based sub-network of NOLANet evaluates only the information provided by the visual features of a video. This sub-network evaluates the temporal features from the Xception Network features and the 1D DCT Features. The XceptionNet output features are of size 2048, and the sum of each 1D DCT of the three landmark bounding box totals 765. These features are concatenated, making the total input size 2,324 features, and sent through an LSTM (Fig. 6) with a temporal window of 1 second. This model is shown in 9

D. Training Configuration

In order to get an initial validation of our multimodal network design, we train a subnetwork that consists of XceptionNet and our final 2-layer LSTM network. We use XceptionNet re-trained on ImageNet as our feature extractor and we do not perform any fine-tuning on it. The LSTM network consists of two hidden LSTM layers of size 1024 followed by two fully connected layers of of 1024 nodes followed by a final output layer of 2 nodes. The videos are randomly sampled from the original DFDC dataset to ensure a decent distribution in terms of subjects, lighting, and other video settings.

Dataset: We train the LSTM network on a fraction of DFDC [49] dataset of 5,000 videos (approximately 1.5M frames. This dataset is composed of 50% deepfake videos and 50% real videos. We use 90% of the data for training and 10% for validation.

Hyperparameters: As mentioned before, we train only the LSTM network by optimizing cross-entropy using Adam optimizer [50] loss with learning rate of $\alpha = 10^{-5}$. We run the training for 1,000 epochs with a mini-batch size of 500.

V. Results

We tested each sub-network to analyze its individual contributions and determine whether it should stay within the overall model. The results are described for each sub-network, and can be seen in Table I.

TABLE I: Training and Validation Results

Network	Evaluation dataset	Accuracy
Xception + LSTM	DFDC Partial training (4,500 training and 500 validation)	54.82%
VSNet		61.95%
LipSpeech		59.21%
FourierNet		50.20%
Selim Seferbekov (DFDC Winner)	Full DFDC training	82%
	DFDC hidden test set	65.18%

LipSpeech

LipSpeech, which evaluates the translated text based on the mouth movements and spectrogram of the audio, does not appear be very meaningful in deepfake detection on the DFDC dataset. Testing accuracy was 59.21%. Though these results show limited improvement to random guessing, they support the claim that audio-visual disharmony is present in deepfakes, as proposed in [41] and [48]. With advancements in lipreading models, better results are likely to follow.

FourierNet

FourierNet, which evaluates the significance of features from the frequency domain of visual and audio, suggests to not be very meaningful in deepfake detection on the DFDC dataset. Testing accuracy was only 50.20%. These results suggest there is no correlation between spectral features in the audio domain and spectral features in the visual domain. Given that audio-visual disharmony in the spectral domain are not related to the believability of a deepfake, these results are unsurprising. However, we believe this method could be expanded to specifically audio-driven generation of deepfakes, in which audio is dubbed to match the movement of the mouth in a video, as this method of deepfaking, if perform well, could have the potential to undermine the authenticity of words of any public figure. Should this method become prevalent in deepfakes, there are likely to be artifacts in the spectral domain of audio, and there may be a relationship to artifacts in the visual domain.

XceptionNet

We tested XceptionNet features with an added LSTM component (Xception + LSTM) in isolation for deepfake detection. The results on the validation set achieved accuracy of 54.82%. The visual features that XceptionNet extracts from the RGB information on individual frames are proven to be very effective in image classification [9]. However, we found that these features by themselves do not result in a good detector of deepfakes. XceptionNet focuses on high level features to make a frame-based classification of the general class of an image; hence, it does not focus too much on more nuanced features and discrepancies. When we combine XceptionNet with other features, it performs better.

VSNet

For the sub-network VSNet, which consisted of XceptionNet features concatenated with the 1-D DCT of facial landmarks, we improved the results on the isolated XceptionNet Features. This sub-network achieved accuracy of 61.95% on the testing data. These results suggest that the addition of spectral features to strictly visual neural features can improve deepfake detection. Moreover, these results suggest that the DCT can effectively capture the relevant features in the spectral domain, as opposed to the Fourier transform.

VI. Conclusion

In this work, we have designed a multimodal network to detect deepfake videos called NOLANet. We have shown that we have been able to widely explore the Deepfake Detection Challenge dataset by extracting a wide spectrum of features. We introduced new methods for extraction of features in the frequency domain and a new model hypothesis for detecting misalignment between audio and visual features in deepfakes. In testing each sub-network, we determined that the only

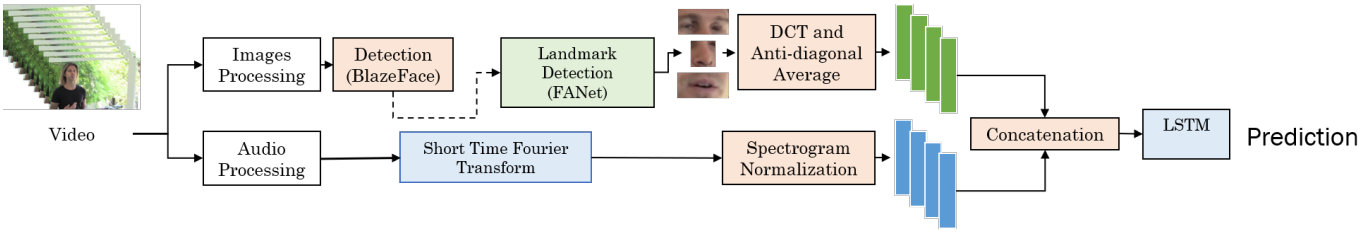


Fig. 8: The pipeline of the FourierNet sub-network. The Discrete Cosine transform is calculated for each landmark bounding box: eyes/eyebrows, nose, mouth. This results in three elements each with a feature size of 255. The Short Time Fourier transform is calculated for the audio sample in the video to get the spectrogram, which is then normalized for a feature size of 730. These input sequences span 1 second. The features are then concatenated and sent through a Long Short Term Memory network which outputs a prediction of real or fake.

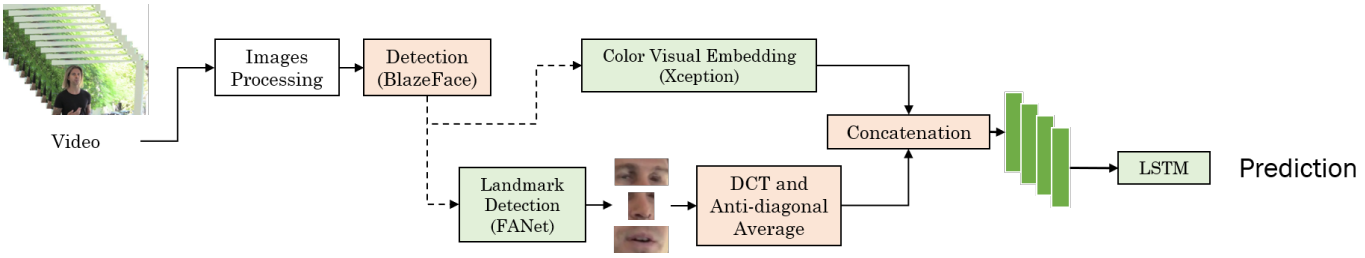


Fig. 9: The pipeline of the VSNet sub-network. The Discrete Cosine transform is calculated for thirty sequential frames for each landmark bounding box: eyes/eyebrows, nose, mouth. This results in three elements each with a feature size of 255 per frame. Each facial bounding box of the face in thirty sequential frames is sent through XceptionNet. These features are then concatenated and sent through a Long Short Term Memory network to make a prediction of real or fake.

relevant sub-networks were LipSpeech, which found audio and visual discrepancies with text prediction models, and VSNet, which combined 1-D Discrete Cosine transform features of the landmarks with the output feature vector from XceptionNet. This confirms the claim that the inclusion of spectral features can improve the accuracy of deepfake detection. It also has revealed that the use of the DCT can be an effective component in deepfake detection. We are, to the best of our knowledge, the first to use the DCT to capture spectral features in the spectral domain.

Despite better-than-random results for LipSpeech, this model could be tested on different models where mouth manipulation is guaranteed in the training data to determine if this approach is more accurate for specific methods of deepfaking. Additionally, an unsupervised method such as contrastive loss could be explored to find a separation in high-dimensional space between DeepSpeech2 output features and LipNet output features. Though FourierNet was no better than random, it too could be tested on a new dataset with confirmed audio manipulation; however, we do not suspect spectral disharmony will be a significant indicator of video manipulation. Training these networks end-to-end may also help the networks find more meaningful features.

Deepfake detection continues to be a challenging task for machine learning models and humans alike as models to create deepfakes improve. This work reveals this growing challenge, but also shows promise in adding features like the DCT, or using lipreading models in tandem with speech-to-text models.

ACKNOWLEDGEMENT

This material is based upon work funded by the National Science Foundation under Award Number CNS–1950873. Any opinions, findings, and conclusions and/or recommendations expressed in this publication are those of the authors and do no necessarily reflect the views of the National Science Foundation. We would like to thank the National Science Foundation for funding this research and the 2020 University of Missouri's Research Experience for Undergraduates in Consumer Networking Technologies for providing the resources and opportunities to conduct this research.

REFERENCES

[1] D. Mack, "This PSA About Fake News From Barack Obama Is Not What It Appears." https://www.buzzfeednews.com/article/davidmack/obama-fake-news-jordan-peele-psa-video-buzzfeed, Apr. 2018.

[2] H. Ajder, G. Patrini, F. Cavalli, and L. Cullen, "The State of Deepfakes: Landscape, Threats, and Impact," Sept. 2019.

[3] J. Rothkopf, "Deepfake Technology Enters the Documentary World," *The New York Times*, July 2020.

[4] S. Cahlan, "Analysis | How misinformation helped spark an attempted coup in Gabon," *Washington Post*, Feb. 2020.

[5] J. Blakkarly, "A gay sex tape is threatening to end the political careers of two men in Malaysia," *SBS News*, June 2019.

[6] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner, "FaceForensics++: Learning to Detect Manipulated Facial Images," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, (Seoul, Korea (South)), pp. 1–11, IEEE, Oct. 2019.

[7] R. Durall, M. Keuper, F.-J. Pfreundt, and J. Keuper, "Unmasking DeepFakes with simple Features," *arXiv:1911.00686 [cs, stat]*, Mar. 2020.

[8] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The deepfake detection challenge dataset," 2020.

[9] F. Chollet, "Xception: Deep Learning With Depthwise Separable Convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[10] Y. M. Assael, B. Shillingford, S. Whiteson, and N. de Freitas, "LipNet: End-to-End Sentence-level Lipreading," *arXiv:1611.01599 [cs]*, Dec. 2016.

[11] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen, J. Chen, J. Chen, Z. Chen, M. Chrzanowski, A. Coates, G. Diamos, K. Ding, N. Du, E. Elsen, J. Engel, W. Fang, L. Fan, C. Fougner, L. Gao, C. Gong, A. Hannun, T. Han, L. V. Johannes, B. Jiang, C. Ju, B. Jun, P. LeGresley, L. Lin, J. Liu, Y. Liu, W. Li, X. Li, D. Ma, S. Narang, A. Ng, S. Ozair, Y. Peng, R. Prenger, S. Qian, Z. Quan, J. Raiman, V. Rao, S. Satheesh, D. Seetapun, S. Sengupta, K. Srinet, A. Sriram, H. Tang, L. Tang, C. Wang, J. Wang, K. Wang, Y. Wang, Z. Wang, Z. Wang, S. Wu, L. Wei, B. Xiao, W. Xie, Y. Xie, D. Yogatama, B. Yuan, J. Zhan, and Z. Zhu, "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, p. 173–182, JMLR.org, 2016.

[12] J. K. Lewis and I. E. Toubal, "jklewis99/MultimodalDeepfakeDetection." https://github.com/jklewis99/MultimodalDeepfakeDetection, 2020.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems 27* (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.), pp. 2672–2680, Curran Associates, Inc., 2014.

[14] Z. Wang, Q. She, and T. E. Ward, "Generative Adversarial Networks in Computer Vision: A Survey and Taxonomy," *arXiv:1906.01529 [cs]*, June 2020.

[15] H. Averbuch-Elor, D. Cohen-Or, J. Kopf, and M. F. Cohen, "Bringing portraits to life," *ACM Transactions on Graphics*, vol. 36, pp. 1–13, Nov. 2017.

[16] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran, and M. Grundmann, "BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs," *arXiv:1907.05047 [cs]*, July 2019.

[17] X. Huang and S. Belongie, "Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, (Venice), pp. 1510–1519, IEEE, Oct. 2017.

[18] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image Style Transfer Using Convolutional Neural Networks," pp. 2414–2423, 2016.

[19] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-To-Image Translation With Conditional Adversarial Networks," pp. 1125–1134, 2017.

[20] C. Li and M. Wand, "Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks," in *Computer Vision – ECCV 2016* (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), Lecture Notes in Computer Science, (Cham), pp. 702–716, Springer International Publishing, 2016.

[21] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic Segmentation using Adversarial Networks," *arXiv:1611.08408 [cs]*, Nov. 2016.

[22] N. Souly, C. Spampinato, and M. Shah, "Semi Supervised Semantic Segmentation Using Generative Adversarial Network," in *2017 IEEE International Conference on Computer Vision (ICCV)*, (Venice), pp. 5689–5697, IEEE, Oct. 2017.

[23] W. Zhu, X. Xiang, T. D. Tran, and X. Xie, "Adversarial Deep Structural Networks for Mammographic Mass Segmentation," *arXiv:1612.05970 [cs]*, June 2017.

[24] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," *arXiv:1609.04802 [cs, stat]*, May 2017.

[25] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy, "ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks," in *Computer Vision – ECCV 2018 Workshops* (L. Leal-Taixé and S. Roth, eds.), vol. 11133, pp. 63–79, Springer International Publishing, 2019.

[26] X. Zhu, L. Zhang, L. Zhang, X. Liu, Y. Shen, and S. Zhao, "GAN-Based Image Super-Resolution with a Novel Quality Loss," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–12, Feb. 2020.

[27] Z. Chen, S. Nie, T. Wu, and C. G. Healey, "High Resolution Face Completion with Multiple Controllable Attributes via Fully End-to-End Progressive Generative Adversarial Networks," *arXiv:1801.07632 [cs]*, Jan. 2018.

[28] B. Dolhansky and C. C. Ferrer, "Eye In-Painting with Exemplar Generative Adversarial Networks," *arXiv:1712.03999 [cs, stat]*, Dec. 2017.

[29] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic Image Inpainting with Deep Generative Models," *arXiv:1607.07539 [cs]*, July 2017.

[30] I. Korshunova, W. Shi, J. Dambre, and L. Theis, "Fast Face-Swap Using Convolutional Neural Networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, (Venice), pp. 3697–3705, IEEE, Oct. 2017.

[31] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, "First Order Motion Model for Image Animation," in *Advances in Neural Information Processing Systems 32* (H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alché-Buc, E. Fox, and R. Garnett, eds.), pp. 7137–7147, Curran Associates, Inc., 2019.

[32] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Niessner, "Face2Face: Real-Time Face Capture and Reenactment of RGB Videos," in *Proceedings of the IEEE Conference on CVPR*, pp. 2387–2395, 2016.

[33] R. W. Sumner and J. Popović, "Deformation transfer for triangle meshes," *ACM Trans. Graph.*, vol. 23, pp. 399–405, Aug. 2004.

[34] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing Obama: learning lip sync from audio," *ACM Trans. Graph.*, vol. 36, pp. 95:1–95:13, July 2017.

[35] J. S. Chung, A. Jamaludin, and A. Zisserman, "You said that?," *arXiv:1705.02966 [cs]*, July 2017.

[36] K. Vougioukas, S. Petridis, and M. Pantic, "End-to-End Speech-Driven Realistic Facial Animation with Temporal GANs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[37] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, "Neural Voice Puppetry: Audio-driven Facial Reenactment," *arXiv:1912.05566 [cs]*, Dec. 2019.

[38] F. Matern, C. Riess, and M. Stamminger, "Exploiting Visual Artifacts to Expose Deepfakes and Face Manipulations," in *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, (Waikoloa Village, HI, USA), pp. 83–92, IEEE, Jan. 2019.

[39] X. Yang, Y. Li, and S. Lyu, "Exposing Deep Fakes Using Inconsistent Head Poses," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8261–8265, May 2019. ISSN: 2379-190X.

[40] Y. Li, M.-C. Chang, and S. Lyu, "In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking," in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–7, Dec. 2018. ISSN: 2157-4774.

[41] S. Agarwal, H. Farid, O. Fried, and M. Agrawala, "Detecting Deep-Fake Videos From Phoneme-Viseme Mismatches," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[42] Y. Li and S. Lyu, "Exposing DeepFake Videos By Detecting Face Warping Artifacts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[43] M. Hollemans, "hollance/BlazeFace-PyTorch." https://github.com/hollance/BlazeFace-PyTorch, 2020.

[44] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks)," in *International Conference on Computer Vision*, 2017.

[45] A. Bulat, "1adrianb/face-alignment." https://github.com/1adrianb/face-alignment, 2020.

[46] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.

[47] S. Naren, "SeanNaren/deepspeech.pytorch." https://github.com/SeanNaren/deepspeech.pytorch, 2020.

[48] K. Chugh, P. Gupta, A. Dhall, and R. Subramanian, "Not made for each other- Audio-Visual Dissonance-based Deepfake Detection and Localization," *arXiv:2005.14405 [cs]*, May 2020.

[49] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, and C. C. Ferrer, "The deepfake detection challenge (dfdc) preview dataset," 2019.

[50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014.