The Princeton Journal of Interdisciplinary Research
ISSN 3069-8200

Unveiling Truth Through Gestures: A Multimodal Deepfake Detection System

Vishruth I. Rao

Eastlake High School

Abstract

Over the past few years, generative artificial intelligence and other deepfake technologies have been improving every day. Although these technologies have many useful applications, they are also being used maliciously to spread fake news, blackmail, scam, and cause cyberattacks. Thus, it is important to accurately detect if these videos are real or if they are the creation of artificial intelligence. Much of the recent work on these uses face images and Generative Adversarial Network (GAN) artifacts to detect deepfakes and generated videos. Instead, we propose to use a multimodal architecture that uses gesture information and the whole body in order to detect deepfakes. In addition, this model will also use EfficientNet to extract deep features from an image and assist in deepfake detection. On the Deepfake Dataset Challenge (DFDC) and the FakeAVCeleb dataset, the system has achieved an AUC of 0.9658 and 0.9714 respectively.

Keywords: deepfake detection, gestures, keypoints, artifacts, deepfakes

The Princeton Journal of Interdisciplinary Research
ISSN 3069-8200

Introduction

There have been many advancements in generative Artificial Intelligence (AI) technology of late. The term deepfake refers to a video or image that was generated or modified by AI. Deepfakes have widely been considered a threat to society (Chesney and Citron, 2019; Rana et al., 2022; Zheng et al., 2019). They can be used to alter videos quite easily without human input, as seen by the number of large deepfake datasets (Dolhansky et al., 2020; Jiang et al., 2020; Khalid et al., 2021; Rossler et al., 2019). These datasets contain thousands of deepfake videos generated using a multitude of techniques. Deepfakes can be used for many malicious purposes. Bad actors can use generated and manipulated videos for disinformation, societal polarization, blackmail, scamming, cyberbullying, embarrassment, and privacy violations. One of the most viable ways of combating deepfakes is by detecting and flagging them.

There are multiple methods to detect deepfakes. For example, some methods analyze the noise of a video (Hasan and Salah, 2019) while others check meta-data like information to identify these deepfakes (Koopman et al., 2018). However, the most common methods use unimodal deep learning to detect deepfakes. Many of these methods are purely based on detecting the artifacts present in deepfakes (Afchar et al., 2018; Coccomini et al., 2022; Montserrat et al., 2020). However, as generative technology continues to grow, this becomes more and more difficult to do. Recently, some new multimodal deepfake detection methods (Mittal et al., 2020; Becattini et al., 2024) have been proposed which have been shown to have improved results compared to the unimodal methods. Unfortunately, most multimodal methods continue to focus on detecting artifacts and anomalies in the face rather than focusing on the whole body due to its challenges such as the high dimensionality and the complexity of the body.

It has been shown that GANs struggle to generate realistic gestures (Kontogiannis et al., 2024; Zhang et al., 2019).

In this paper, we propose a multimodal framework that uses both face information and body information to accurately detect deepfakes. The face information that is used is the image of the face and the facial keypoints. Keypoints are a series of coordinates that can represent gestures. The body information that is utilized is the image of the body and the body keypoints and hand keypoints. The system processes the raw images to detect artifacts and processes the keypoints to detect inaccuracies and inconsistencies. This information is used to accurately detect a deepfake.

The contributions of the paper can be summarized as follows:

- We present a multimodal architecture that combines facial expressions and body gestures with extracted artifacts to detect deepfakes from video information.
- We evaluate our model on multiple datasets, and we demonstrate that our method works well on both the DFDC and FakeAVCeleb datasets.
- To the best of our knowledge, we are the first work that uses hand and body gestures to detect deepfakes.

Related Work

Deepfakes and Deepfake Detection is a relatively new field, first coming into prominence around five years ago. Deepfake detection started with the need to identify deepfake videos from

non-deepfake videos. Thus, before one can talk about deepfake detection, it is good to first explore how deepfakes are generated to better understand how deepfakes work. Then, one can explore more about two different types of deepfake detection: Unimodal detection, which typically runs one image through an autoencoder to identify deepfakes, and multimodal detection, which adds additional information such as audio, head position, and eye blinking to more accurately detect deepfakes.

Generation

To look at detection, we first have to talk about generation. Most of the work tries to detect artifacts left behind in the generation process. Fake images are created by either taking an original image and modifying it in some way or generating a completely new image. Modifying images is done by swapping the original image face with a target face, as shown in these works (Chen et al., 2020; Korshunova et al., 2017; Perov et al., 2020). The other method works by generating the pixels from scratch. This usually happens by transforming an image from random noise (Karras, 2019) or from a text-based input (Li et al., 2019; Rombach et al., 2022). However, both methods typically make use of GANs. These networks typically leave behind artifacts in a generated image and typically struggle to generate realistic gestures (Zhang et al., 2019; Kontogiannis et al., 2024).

Detection

There are multiple ways to detect deepfakes. Traditionally, most detectors used an MLP (Multi- processing Perceptron) to efficiently detect a manipulated video (Sahla Habeeba et al., 2021). There are also methods that use the metadata-like information and noise of a video to

detect if it is a deepfake (Koopman et al., 2018; Hasan and Salah, 2019). However, the most popular and promising way to detect deepfakes is by using deep learning methods (Rana et al., 2022).

Unimodal Detection

One of the easiest ways to identify a deepfake is to run a frame-by-frame analysis and find the artifacts left behind by the GAN in the generation process. To do this, most detectors utilize a convolutional autoencoder to extract deep features such as these artifacts (Bonettini et al., 2021; Coccomini et al., 2022; Montserrat et al., 2020). One of the first iterations of this was MesoNet (Afchar et al., 2018), which used a relatively shallow CNN to try to detect these artifacts. Nowadays, however, a more complex autoencoder such as EfficientNet (Tan and Le, 2019) is used. This is because EfficientNet is pretrained on the ImageNet dataset (Deng et al., 2009) and has already learned many of the low level features found in images. Additionally, EfficientNet's ability to automatically scale its Convolutional Neural Networks (CNNs) allow for it to have better efficiency and accuracy than other convolutional autoencoders. Typically, EfficientNet layers are also modified to allow for better performance. For example, (Bonettini et. al., 2021) modified the EfficientNetB4 and used model assembling to add an attention mechanism and Siamese training strategies to better detect deepfakes. In (Montserrat et al., 2020), deepfakes were detected by combining EfficientNet with a Gated Recurrent Unit and using that to detect spatio-temporal anomalies and artifacts. (De Lima et al., 2020) also attempted to detect deepfakes using spatio-temporal features by using a 3DCNN network. In (Amerini et al., 2019), deepfakes were detected by exploiting optical flow to detect video glitches. Finally, in

(Coccomini et al., 2022), EfficientNet is combined with an Efficient Vision Transformer and a Convolutional Cross Vision Transformer to better detect artifacts.

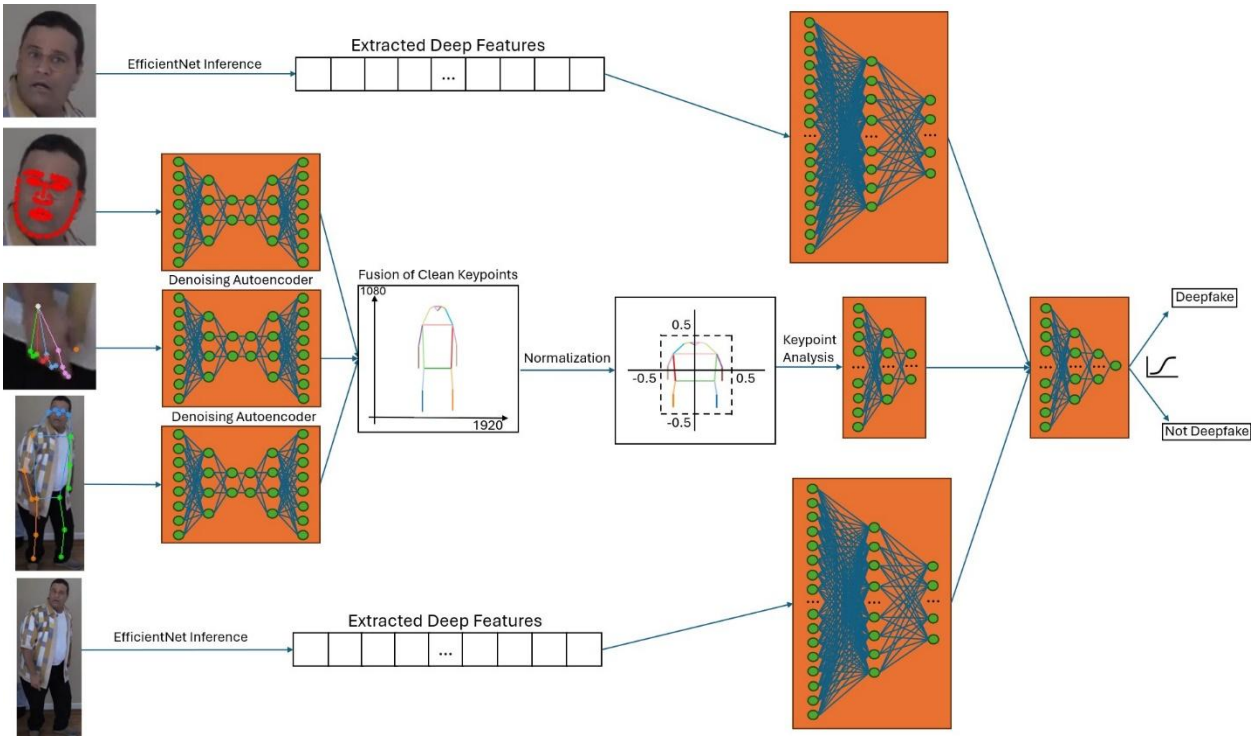
Multimodal Detection

Recent deepfake detection methods leverage multimodal and hand-crafted features to better utilize video data. These multimodal architectures rely on images and audio to assist in detecting deepfakes. For example, (Mittal et al., 2020) proposes an architecture that can extract the similarities found in the audio and visual modalities of a video. It also extracts the perceived emotion within the video. By combining these two pieces of information, it can detect deepfakes. This work outperformed many of the works in its time, including MesoNet (Afchar et al., 2018). (Korshunov and Marcel 2018) focuses on detecting audio manipulations in a video and combining this information with lip-sync and lip dubbing inconsistencies in order to detect deepfakes. In (Salvi et al., 2023), combined audio-visual features are analyzed by a time-aware neural network in order to find time accuracy and semantic inconsistencies that can be used to detect deepfakes. Hand crafted features have also been used to improve deepfake detection before. A recent work (Becattini et al., 2024) extracts the head pose of the person in the video. The head poses are run through a machine learning algorithm that uses Dynamic Time Warping to determine whether the temporal patterns in the video indicate a deepfake. Finally, (Li et al., 2018) takes advantage of generative technologies' inability to generate videos with realistic accurate eye blinking. By using a CNN based classifier and Long-Term Recurrent Convolutional Networks (LRCNs), the length and the frequency of eye blinks can be judged to see if they are realistic or the product of a deepfake. It seems that multimodal architectures are a promising way to detect deepfakes but are mostly unexplored, as all of these methods rely on using the face to

detect deepfakes. However, not all artifacts are on the face. As discussed previously, GANs struggle to create realistic gestures, and no one has utilized gestures to detect deepfakes due to its challenges. Thus, our system will be leveraging gesture information and whole body images to detect deepfakes.

Figure 1

The Proposed Model Pipeline for Detecting Deepfakes.



The body image and face image both go through the EfficientNet encoder and multiple linear projections, while the keypoints first need to be denoised and normalized before being further analyzed. Finally, the predictions from all the branches are used to create a final prediction for this image. The image predictions later get compiled down into a video prediction**.**

Methodology

In this paper, we are proposing a multimodal methodology that combines face and body information to accurately detect deepfakes, which has not yet been worked on. Our proposed methodology splits a video into frames and analyzes each of the frames separately. It then extracts the gesture information from the image in the form of facial landmarks and body keypoints. The image is run through a convolutional autoencoder in order to extract and analyze deep features that will help detect if the image came from a deepfake video. The facial landmarks and body keypoints are normalized and analyzed by our keypoint analyzer to detect any abnormalities that could show signs of a deepfake. The results from the individual frames of a video are then combined in order to produce a prediction for the video as a whole. See Fig. 1 for the model diagram.

Datasets

In order to talk about the model and preprocessing, one must first discuss the datasets used. The main dataset that was used was the DFDC (Deepfake Detection Challenge) dataset (Dolhansky et al., 2020). The DFDC dataset is one of the largest and most high quality datasets for deepfake detection. The dataset consists of over 10,000 short videos of 10-30 seconds in length. Additionally, each deepfake video is unique and is generated by 8 different deepfake generation algorithms. In total, the full dataset contains over 500 gigabytes of dataset. The DFDC dataset is also a difficult dataset to detect deepfakes on, as they utilize post-processing modifications such as noise, low light conditions, video compressions, and audio compressions, which have been proven to reduce the chance of a modified video being detect as a deepfake.

This dataset also has one more important attribute: it shows the whole body in the video, which we will need to utilize in order to detect gestures.

The second dataset that was used was FakeAVCeleb (Khalid et al., 2021), another large audio-visual dataset. This dataset contains thousands of deepfaked videos made from real videos of celebrities. Because of this, there are many high quality videos to generate deepfakes from. Additionally, the FakeAVCeleb dataset uses more people in their real videos than the DFDC dataset, which can allow for a higher quality dataset. This dataset also includes much of the post-processing that the DFDC does, include video and audio compression. Again, this dataset shows the whole body in the videos, which we use to extract gestures.

Preprocessing

The first preprocessing step is to split the videos into frames. However, it is not necessary to evaluate every frame in the video. Because there is very little gesture movement and changes between frames, it is not necessary to extract every frame in a video. Instead, the system extracts every tenth frame in order to achieve a balance between data size and data quality. Additionally, data augmentations were used in order to generate more data while training our model. Common augmentations such as rotations, cropping, noise, etc were utilized. This forced the model to not overfit on the training data and allowed it to better detect post-processed deepfakes.

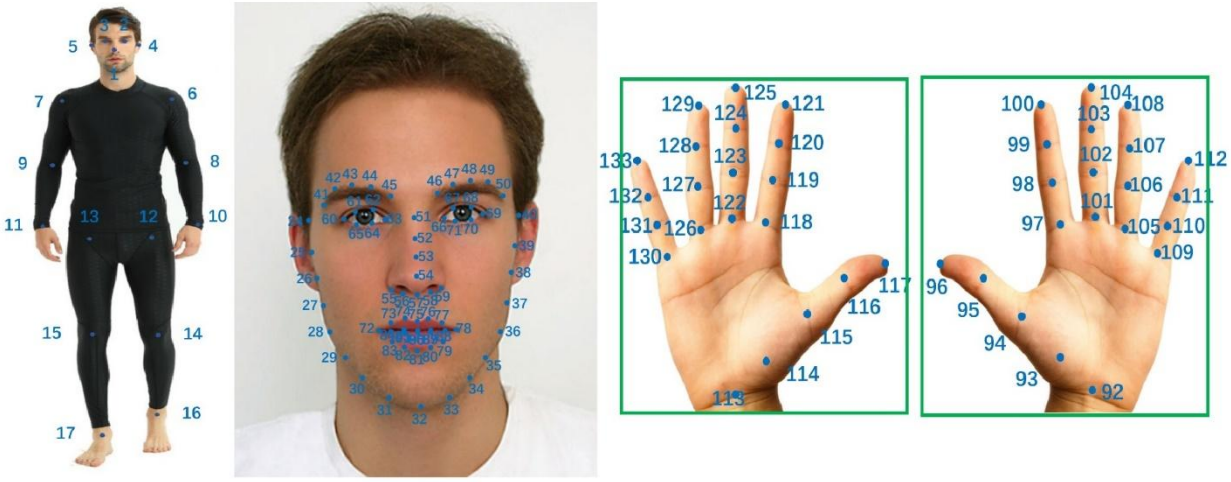
Keypoint Extraction

The keypoints that describe the gestures of the person are also extracted in this step. The Sapiens (Khirodkar et al., 2024) model with two billion parameters was used for this task. This model has shown a high accuracy at detecting keypoints on the face, hands, and body. It was

trained on the Common Objects in Context (COCO) Whole body dataset (Jin et al., 2020). This dataset contains 133 keypoints on the face, body, hands, and feet. 68 of the keypoints describe the face, 17 of them describe the body, and 21 of them describe each hand. The feet keypoints were not used due to the fact that the feet keypoints are not very useful in describing gestures. Thus, the system only uses127 of the 133 keypoints. Please see Fig. 2 for a visual description of the keypoints.

Figure 2

The Keypoints that are Used are Defined Above.



We extract 17 body keypoints, 46 face keypoints, and 42 hand keypoints as per the COCO Whole body dataset. However, we do not use the feet keypoints as they contain very little information.

Deep feature analysis

Although using gestures to detect deepfakes is the main focus of the system, a portion of our model still analyzes the image to find artifacts and assist in deepfake detection. This is important as sometimes the body is not fully visible in a deepfake video and it is thus necessary to utilize these more traditional methods of deepfake detection.

Thedesign for this is relatively simple. Most importantly, the image is run through a convolutional autoencoder to extract features out of the image. This is achieved by using the EfficientNetv2 (Tan and Le, 2021) autoencoder. EfficientNet is a family of autoencoders that can automatically scale their CNNs up and down as needed. This allows them to be more efficient and accurate than other convolutional autoencoders. This automatic scaling leads to shorter training times, which allows rapid development of the model. EfficientNetv2 uses many of the same principles as the original EfficientNet while being more efficient, faster, and accurate. Additionally, EfficientNetv2 was pretrained on the ImageNet dataset (Deng et al., 2009) and has already learned many of the low level features found in images. This allows it to be a good feature extractor for images. The specific model that was used was the EfficientNetv2XL model that was pretrained on the ImageNet 21k dataset and fine tuned on the ImageNet 1k dataset.

Before running the image through an auto encoder, it is helpful to first crop the image. Two different images need to be run through the autoencoder: an image of the face and an image of the whole body. In order to find a image of the face, a multi-task cascaded convolutional network (MTCNN) (Zhang et al., 2016) model was used. The MTCNN model is a series of cascading CNNs specifically made for finding faces in images. In order to extract the body image, a pretrained You Only Look Once (YOLO)v8 (Varghese and Sambath, 2024) model was used. YOLOv8 is a single shot object detection model that is extremely efficient. By pretraining

it on the COCO dataset, it can quickly identify the bounding body of the body. Once the bounding box of both the face and the body have been identified, a crop of the image needs to be taken. While making a crop, it is important to leave a bit of margin so that the EfficientNet autoencoder has context to work with along with the face/body image. The results that the autoencoder gives are then flattened, pooled, and passed through a linear projection to predict the chance of this image being a deepfake.

Gesture analysis

As mentioned in section 3, the gestures in a video are represented by a series of keypoints and landmarks. In section 3.2, we split these keypoints into 3 categories: face landmarks, body keypoints, and hand keypoints. There are 68 face landmarks, 17 body keypoints, and 42 hand keypoints, which means 127 keypoints must be analyzed to determine if this image is from a deepfake video or not.

Denoising Autoencoder

The first step in processing the keypoints is to denoise them. To understand why this is necessary, it is important to look at how these keypoints were generated. They were generated from a potentially modified or generated video that had lots of post-processing applied to it to make it harder to detect if it was a deepfake. This post-processing could have also made it harder to find the locations of keypoints. After this post-processing, the keypoints were found by using another AI model that could have generated even more noisy keypoints due to the bad quality of input data. If these unprocessed keypoints were fed into our model, it would have a difficult time

attempting to find patterns between deepfake keypoints and normal keypoints. For this reason, it is important that as much noise as possible is removed from these keypoints.

A denoising autoencoder was used to denoise the keypoints. A denoising autoencoder is typically used to denoise and restore images, but it can also denoise other types of media as well. This special type of autoencoder follows the encoder-decoder model. The encoder encodes features into a smaller latent space while the decoder takes these features from the latent space and attempts to rebuild the keypoints. By using supervised training, the denoising autoencoder can learn how to rebuild the keypoints without the noise. After some testing, it was realized that the most optimal setup is to create three separate denoising autoencoders for the different types of keypoints. This is because certain keypoints, such as the hand keypoints, were more prone to noise than the body keypoints.

Normalization

When keypoints are extracted, pixel coordinates are given that describe the location of these keypoints. However, every video and person is different. The person might be located at different parts of the video at different times. The person might have a different body shape than others. These are all reasons why normalization is necessary. Normalization transforms the keypoints into relative keypoints so that they are unaffected by outside factors such as the pixel position of a person on the screen on the person's physical characteristics (e.g. being tall or short). This is especially important for gestures as gestures do not depend on the person's position or body shape, so having this information could cause the model to get confused. Multiple gesture recognition systems (Arwoko et al., 2022; Dang et al., 2022; Schneider et al., 2019) use normalization to improve performance of their models. Normalization solves the

position problem by resetting the origin of the keypoints at a dedicated center. Normalization then solves the body shape problem by having a scale on these keypoints such that the maximum distance a keypoint can be from the origin is one. Normalized keypoints can be defined by equations 1 and 2:

$$x_i = \frac{x_i - C_x}{s} \quad (1)$$

$$y_i = \frac{y_i - C_y}{s} \quad (2)$$

Where x and y are the coordinates of the keypoint, C describes the center coordinate, and s describes the scaling. The difficult part about normalization is selecting where to put the center and what to use as your scale. Three different normalizations were used in this paper: one for the face keypoints, one for the body keypoints, and one for the hands keypoints. For the face keypoints, keypoint 57 (see Fig. 2) was used as the center and the scaling is the distance between keypoints 24 and 40. This achieves an optimal normalization. For the body, the x center is the average keypoints 6 and 7 and the y center between is between keypoint 6 and keypoint 1. For the scale, the distance between keypoints 6 and 7 was used. This was based on the work done in (Schneider et al., 2019). For the hands, the center is the average of all the hand coordinates and the spread is the maximum distance between the wrist keypoint and every other keypoint. This is based off of the work done in (Dang et al., 2022).

Keypoint Analyzer

After normalization, the keypoints are ready for analysis. The system first calculates the relative distances and angles of keypoints, as this will also be passed into the keypoint analyzer.

To do this, a neural network based on the research done in the gesture recognition field was created. Most of the work done in this field specifically focuses on analyzing and extracting information and features out of keypoints. The model was based on the work done these key studies: (Arwoko et al., 2022; Dang et al., 2022; Schneider et al., 2019; Wu and Shao, 2014). The network contains several linear projections and Rectified Linear Unit (ReLU) activation functions that are layered together to create a prediction for the deepfake based on the patterns found in the keypoints and gestures. This is similar to the other works done in the field. It is also important to notice that this only works because of the keypoint denoising and normalization that was done previously. Without those steps, this architecture fails to accurately predict deepfakes.

Experimental Evaluation

In order to determine how well our method for deepfake detection works, our model was train and and used to compare our method to other deepfake detection methods on both the DFDC and FakeAVCeleb datasets. the cross-dataset performance was also compared to see how well the model can detect types of deepfakes that it has never seen before.

Training

Our model was trained over the course of multiple days on a cloud server with an NVIDIA V100 with NVIDIA drivers installed. Multiple experiments were done to determine the best hyper-parameters to use. The dataset was preprocessed before starting the training in order to improve the training times. 75% of the dataset was used for training, 15% of the dataset was used for validation, and 10% of the dataset was used for testing. Since there are very few real videos in the dataset due to the large amount of deepfakes, the real videos were oversampled in

order to maintain an equal split of real and fake videos in the dataset. The denoising autoencoders were trained by injecting Gaussian noise into clean keypoints. The amount of noise injected differs based on the type of keypoint.

Comparison

Table 1

Comparison of Methods on the DFDC Dataset

Method	AUC
Ours	0.9658
Cross Efficient Vision Transformer (Coccomini et al., 2022)	0.951
Efficient Vision Transformer (Coccomini et al., 2022)	0.919
Vision Transformer with distillation (Heo et al., 2021)	**0.978**
Convolutional ViT (Wodajo and Atnafu, 2021)	0.915
CNN and RNN-GRU (Montserrat et al., 2020)	0.9188
EfficientNetB4 + EfficientNetB4ST + B4Att (Bonettini et al., 2021)	0.8813

Note. This table is a comparison of different methods on DFDC dataset. All methods were both trained and evaluated on the DFDC dataset. Notice our strong performance on this dataset.

Table 1 shows the comparison of our fully trained model to other methods for deepfake detection. Our model was both trained and tested on the DFDC dataset, as were the other models. Our method has strong accuracy compared to the many of the other top deepfake detection methods for the DFDC dataset. This is likely due to our use of gestures to detect deepfakes, as the purely EfficientNet method achieves a much lower AUC (area under curve). Our method has also surpassed most, but not all, of the vision transformer based models.

Table 2

Comparison of Methods on the FakeAVCeleb Dataset

Method	AUC
Ours	0.9714
FACTOR (Reiss et al., 2023)	**0.974**
FTCN (Zheng et al., 2021)	0.931
Lip Forensics (Bonettini et al., 2021)	0.911
Real Forensics (Wodajo and Atnafu, 2021)	0.971
XCeption (Montserrat et al., 2020)	0.853

Note. This table is a comparison of different methods on the FakeAVCeleb dataset. All methods were both trained and evaluated on the FakeAVCeleb dataset. Notice our strong and comparable performance on this dataset.

Table 2 shows the comparison between our model to other methods for deepfake detection. Our model was trained and tested on the FakeAVCeleb dataset, as were the other models. Because this is an audio-visual dataset, there are more multimodal deepfake detection methods made for this dataset. Our model is comparable to other models for this dataset, with the FACTOR model being the model with the highest AUC. However, our method comes in at a close second. The table also proves the relative strength of multimodal deepfake detection. For example, FACTOR and Real Forensics are multimodal deepfake detection frameworks.

Table 3

Cross-Dataset Comparison

Method	AUC
Ours (Cross-Dataset)	0.9487
FACTOR (Reiss et al., 2023)	0.974
FTCN (Zheng et al., 2021)	0.931
Lip Forensics (Bonettini et al., 2021)	0.911
Real Forensics (Wodajo and Atnafu, 2021)	0.971
XCeption (Montserrat et al., 2020)	0.853

Note. This is the result of our method trained on the DFDC dataset and tested on the FakeAVCeleb dataset compared to other methods trained and tested on the FakeAVCeleb dataset. Note the strong performance of our method even though it was not trained on the FakeAVCeleb dataset.

Finally, Table 3 shows a comparison of our cross-dataset method to other methods for deepfake detection. Our model was trained on the DFDC dataset while being evaluating on the FakeAVCeleb dataset. Notice the strong performance of the model for this cross-dataset validation. While a cross-dataset validation is a difficult challenge for any model to complete, it also shows that our model has generally learned how to extract artifacts that are common in most GANs. The model has generally learned what generated gestures look like in most GANs. This cross-dataset validation is especially important in deepfake detection as most deepfakes are generated through a variability of new and consistently improving methods. The cross-dataset test shows that the model has been able to detect deepfakes generated by an algorithm that it might not have seen before with a good level of success.

Conclusion

In this work, we proposed a multimodal deepfake detection system that uses gestures to detect deepfakes. We also showed that the system can achieve strong results on DFDC and FakeAVCeleb dataset. We further showed the importance of gestures and hand information as additional tools to assist in deepfake detection. However, the system does have some limitations. The main limitation of the method is that the accuracy goes down if the video is not a full body video as the system cannot utilize gesture information and can only rely on GAN artifacts Zto predict for a deepfake. In the future, we hope to continue our work and use different methods to better analyze deepfakes or incorporate more modalities to make our system more reliable. For example, temporal information could be given to the model to allow it to predict deepfakes better. Additional location-based features such as action units could also serve as a way to increase the accuracy of the model. Deepfake detection is a new and important field for preventing misinformation and abuse of generative technologies, and we hope our work attracts more studies to be done on multimodal deepfake detecting and usage of gestures in deepfake detection.

References

Afchar, D., Nozick, V., Yamagishi, J., Echizen, I. (2018). Mesonet: a compact facial video forgery detection network. In 2018 ieee international workshop on information forensics and security (wifs) (pp. 1–7).

Amerini, I., Galteri, L., Caldelli, R., Del Bimbo, A. (2019). Deepfake video detection through optical flow based cnn. In Proceedings of the ieee/cvf international conference on computer vision workshops (pp. 0–0).

Arwoko, H., Yuniarno, E. M., Purnomo, M. H. (2022). Hand gesture recognition based on keypoint vector. In 2022 international electronics symposium (ies) (pp. 530–533).

Becattini, F., Bisogni, C., Loia, V., Pero, C., Hao, F. (2024). Head pose estimation patterns as deepfake detectors. ACM Transactions on Multimedia Computing, Communications and Applications, 20 (11), 1–24.

Bonettini, N., Cannas, E. D., Mandelli, S., Bondi, L., Bestagini, P., Tubaro, S. (2021). Video face manipulation detection through ensemble of cnns. In 2020 25th international conference on pattern recognition (icpr) (pp. 5012–5019).

Chen, R., Chen, X., Ni, B., Ge, Y. (2020). Simswap: An efficient framework for high fidelity face swapping. In Proceedings of the 28th acm international conference on multimedia (pp. 2003–2011).

Chesney, B., Citron, D. (2019). Deep fakes: A looming challenge for privacy, democracy, and national security. Calif. L. Rev., 107 , 1753.

Coccomini, D. A., Messina, N., Gennaro, C., Falchi, F. (2022). Combining efficientnet and vision transformers for video deepfake detection. In International conference on image analysis and processing (pp. 219–229).

Dang, T. L., Tran, S. D., Nguyen, T. H., Kim, S., Monet, N. (2022). An improved hand gesture recognition system using keypoints and hand bounding boxes. Array, 16 , 100251.

De Lima, O., Franklin, S., Basu, S., Karwoski, B., George, A. (2020). Deepfake detection using spatiotemporal convolutional networks. arXiv preprint arXiv:2006.14749 .

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In 2009 ieee conference on computer vision and pattern recognition (pp. 248–255).

Dolhansky, B., Bitton, J., Pflaum, B., Lu, J., Howes, R., Wang, M., Ferrer, C. C. (2020). The deepfake detection challenge (dfdc) dataset. arXiv preprint arXiv:2006.07397 .

Hasan, H. R., Salah, K. (2019). Combating deepfake videos using blockchain and smart contracts. Ieee Access, 7 , 41596–41606.

Jiang, L., Li, R., Wu, W., Qian, C., Loy, C. C. (2020). Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection. In Proceedings of the ieee/cvf conference on computer vision and pattern recognition (pp. 2889–2898).

Jin, S., Xu, L., Xu, J., Wang, C., Liu, W., Qian, C., . . . Luo, P. (2020). Whole-body human pose estimation in the wild. In Computer vision–eccv 2020: 16th european conference, glasgow, uk, august 23–28, 2020, proceedings, part ix 16 (pp. 196–214).

Karras, T. (2019). A style-based generator architecture for generative adversarial networks. arXiv preprint arXiv:1812.04948 .

Khalid, H., Tariq, S., Kim, M., Woo, S. S. (2021). Fakeavceleb: A novel audio-video multimodal deepfake dataset. arXiv preprint arXiv:2108.05080 .9

Khirodkar, R., Bagautdinov, T., Martinez, J., Zhaoen, S., James, A., Selednik, P., . . . Saito, S. (2025). Sapiens: Foundation for human vision models. In European conference on computer vision (pp. 206–228).

Kontogiannis, G., Tzamalis, P., Nikoletseas, S. (2024). Exploring the impact of synthetic data on human gesture recognition tasks using gans. In 2024 20th international conference on distributed computing in smart systems and the internet of things (dcoss-iot) (pp. 384–391).

Koopman, M., Rodriguez, A. M., Geradts, Z. (2018). Detection of deepfake video manipulation. In The 20th irish machine vision and image processing conference (imvip) (pp. 133–136).

Korshunov, P., Marcel, S. (2018). Speaker inconsistency detection in tampered video. In 2018 26th european signal processing conference (eusipco) (pp. 2375–2379).

Korshunova, I., Shi, W., Dambre, J., Theis, L. (2017). Fast face-swap using convolutional neural networks. In Proceedings of the ieee international conference on computer vision (pp. 3677–3685).

Li, B., Qi, X., Lukasiewicz, T., Torr, P. (2019). Controllable text-to-image generation. Advances in neural information processing systems, 32 .

Li, Y., Chang, M.-C., Lyu, S. (2018). 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).

Mittal, T., Bhattacharya, U., Chandra, R., Bera, A., Manocha, D. (2020). Emotions don't lie: An audio-visual deepfake detection method using affective cues. In Proceedings of the 28th acm international conference on multimedia (pp. 2823–2832).

Montserrat, D. M., Hao, H., Yarlagadda, S. K., Baireddy, S., Shao, R., Horv´ath, J., . . . others (2020). Deepfakes detection with automatic face weighting. In Proceedings of the ieee/cvf conference on computer vision and pattern recognition workshops (pp. 668–669).

Perov, I., Gao, D., Chervoniy, N., Liu, K., Marangonda, S., Um´e, C., . . . others (2020). Deepfacelab: Integrated, flexible and extensible face-swapping framework. arXiv preprint arXiv:2005.05535 .

Rana, M. S., Nobi, M. N., Murali, B., Sung, A. H. (2022). Deepfake detection: A systematic literature review. IEEE access, 10 , 25494–25513.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In Proceedings of the ieee/cvf conference on computer vision and pattern recognition (pp. 10684–10695).

Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., Nießner, M. (2019). Faceforensics++: Learning to detect manipulated facial images. In Proceedings of the ieee/cvf international conference on computer vision (pp. 1–11).

Sahla Habeeba, M., Lijiya, A., Chacko, A. M. (2021). Detection of deepfakes using visual artifacts and neural network classifier. In Innovations in electrical and electronic engineering: Proceedings of iceee 2020 (pp. 411–422).

Salvi, D., Liu, H., Mandelli, S., Bestagini, P., Zhou, W., Zhang, W., Tubaro, S. (2023). A robust approach to multimodal deepfake detection. Journal of Imaging, 9 (6), 122.

Schneider, P., Memmesheimer, R., Kramer, I., Paulus, D. (2019). Gesture recognition in rgb videos using human body keypoints and dynamic time warping. In Robocup 2019: Robot world cup xxiii 23 (pp. 281–293).

Tan, M., Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In International conference on machine learning (pp. 6105–6114).

Tan, M., Le, Q. (2021). Efficientnetv2: Smaller models and faster training. In International conference on machine learning (pp. 10096–10106).

Varghese, R., Sambath, M. (2024). Yolov8: A novel object detection algorithm with enhanced performance and robustness. In 2024 international conference on advances in data engineering and intelligent computing systems (adics) (pp. 1–6).

Wodajo, D., Atnafu, S. (2021). Deepfake video detection using convolutional vision transformer. arXiv preprint arXiv:2102.11126 .

Wu, D., Shao, L. (2014). Multimodal dynamic networks for gesture recognition. In Proceedings of the 22nd acm international conference on multimedia (pp. 945–948).

Zhang, K., Zhang, Z., Li, Z., Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. IEEE signal processing letters, 23 (10), 1499–1503.

Zhang, X., Karaman, S., Chang, S.-F. (2019). Detecting and simulating artifacts in gan fake images. In 2019 ieee international workshop on information forensics and security (wifs) (pp. 1–6).10

Zheng, L., Zhang, Y., Thing, V. L. (2019). A survey on image tampering and its detection in real-world photos. Journal of Visual Communication and Image Representation, 58 , 380–399.

Zheng, Y., Bao, J., Chen, D., Zeng, M., Wen, F. (2021). Exploring temporal coherence for more general video face forgery detection. In Proceedings of the ieee/cvf international conference on computer vision (pp. 15044–15054).11