DeepFake Slayer

Mahanand Adimulam
Golisano College of Computing
Rochester Institute of Technology
Rochester, USA
ma9333@rit.edu

Niranjan Ambekar
Golisano College of Computing
Rochester Institute of Technology
Rochester, USA
na2883@rit.edu

I. Goal

The goal of this project is to reimplement and evaluate one of the state-of-the-art Deepfake detection models on challenging datasets like the WildDeepfake dataset[2]. We will also conduct a deeper analysis of the pain points of the model which we have chosen and/or write a confidence interval wrapper on top of them.

II. Background

Deepfake are synthetic media in which a person in an existing image or video is replaced with someone else's likeness. While Deepfake can be used for harmless purposes, they can also be used to spread misinformation, damage someone's reputation, or even commit fraud. Deepfake detection models are designed to distinguish between real videos and Deepfake. However, Deepfake are becoming increasingly sophisticated, making it difficult for Deepfake detection models to keep up.

III. Scientific Merit

This project has the potential to make a scientific contribution to the field of Deepfake detection. By reimplementing and evaluating the model on challenging datasets, we can help to identify their strengths and weaknesses and develop new methods to improve their performance. Additionally, by writing a confidence interval wrapper on top of the models, we can help users to understand how confident the models are in their predictions. This is a challenging question worthy of an applied data science project because Deepfake detection is a critical problem that needs to be solved. Deepfake are becoming increasingly sophisticated and are being used to spread misinformation and disinformation, damage people's reputations, and even commit fraud.

This section could be better where you describe how your work is adding value more clearly.

IV. Broader Impacts

This project has the potential to have a broad impact on society by helping to protect people from the harmful effects of Deepfake. By developing more effective Deepfake detection models, we can help to reduce the spread of misinformation and disinformation, protect people's privacy, and preserve democracy. For example, our work could be used to develop new tools to detect Deepfake on social media platforms. This could help to reduce the spread of misinformation and disinformation during elections and other important events. Additionally, our work could be used to develop new tools to protect people's privacy from Deepfake.

V. Team Members Goals and tasks

- Conduct a background literature review and choose a model accordingly.
- Evaluate the model on the WildDeepfake dataset to identify its strengths and weaknesses.
- Improve the model by addressing its pain points.
- Build an uncertainty estimation of the model to quantify the uncertainty in its predictions.

VI. Advisors

Matthew Wright, Professor of Cybersecurity, Rochester Institute of Technology
Saniat Sohrawardi, Rochester Institute of Technology
Kelly Wu, Rochester Institute of Technology

VII. Related Work

Deepfakes, a form of synthetic media that can realistically replace one person's likeness with another, have become a cause for concern due to their potential to be misused for spreading misinformation, damaging reputations, and committing fraud [5]. Deepfakes harness deep learning capabilities to change people's facial expressions and emotions in videos [6]. A common misuse of deepfakes involves removing a victim's clothes for humiliation purposes. To combat this issue, researchers have developed various deepfake detection models. This literature review aims to provide a comprehensive overview of existing deepfake detection approaches and identify potential research gaps. Broadly speaking, there are three main categories of deepfake detection methods: machine/deep learning-based approaches, forensic analysis-based approaches, and facial recognition-based approaches. Deep learning-based approaches employ various CNN architectures to detect deepfakes. For instance, AlexNet and VGG16 have been used to identify fake content evidence [7].

Khodabakhsh et al. [8] explored the application of deep learning to detect forged faces generated with CGI and FakeApp. They trained popular CNN architectures, including AlexNet, Inception, and Xception, on the ImageNet dataset. Gowda et al. [9] employed CNN architectures, including ResNet, Xception, and Ensemble, to detect fake images on various social media platforms. Among these approaches, the ensemble method achieved the highest accuracy, reaching 80%. Ivanov et al. [10] introduced a novel approach to detect fake imagery by combining deep learning and super-resolution algorithms. Qurat et. al [11] compared various CNN architectures to detect forged faces. Afchar et al. [12] developed a promising deep learning model called MesoNet for detecting deepfakes in images and videos. Wang and Dantcheva [13] conducted a comparative study of three CNN models, I3D, ResNet 3D, and ResNeXt3D, to evaluate their performance in deepfake detection for videos. Mitra et al. [14] employed the XceptionNet architecture as a feature extractor for detecting deepfakes in videos. Their deep learning model was trained on videos compressed using the c23 compression level from the FaceForensics++ dataset. Rossler et. al [15] suggested an automation benchmark for facial manipulation detection. Nguyen et al. [16] proposed a capsule network-based approach for detecting various types of computer-generated videos. J. Cao et. al [17] use an innovative multi-scale graph reasoning module that combines encoder output and decoder features into bipartite graphs in a multi-scale fashion for reasoning for forgery detection. Zhiyuan Yan et. al [18] propose a novel approach to DeepFake detection overfitting issues using a disentanglement framework that decomposes image information into three distinct components: forgery irrelevant, method-specific forgery, and common forgery features. Yunsheng Ni et al. [19] introduced a novel framework called Consistent Representation Learning (CORE) to address overfitting issues in deepfake detection. CORE explicitly constrains the consistency of different representations obtained through data augmentation. Specifically, it first captures diverse representations using different augmentations and then regularizes the cosine distance between these representations to enhance their consistency. Yuchen Luo et. al [20] note that current CNN-based detectors tend to overfit to method-specific color textures and thus fail to generalize well. They test the generalizability of their approach on famous datasets such as FaceForensics++ and DeepFake Detection dataset.

So far, several models and architectures that were mentioned earlier have been evaluated on classic DeepFake detection datasets such as DeepFake Detection [21] and the Face Forensics Dataset [22]. Zhiyuan Yan et. al [2] points out the unfair comparisons between different detection models due to lack of uniformity in experimental settings. For this, they propose 'DeepFake Bench', which consists of a unified data modeling system and standardized evaluation metrics. Zi et al. [1] also highlight that many videos in the datasets mentioned above involve a limited number of actors in restricted settings. Zi et al. [1] argue that existing deepfake detection models trained on limited datasets may not generalize well to real-world deepfakes encountered on the internet. To address this issue, they introduced the 'WildDeepFake Dataset,' a collection of 707 videos gathered from the internet. Their evaluation demonstrated that the performance of several established deepfake detection models significantly deteriorates on the WildDeepFake Dataset, highlighting the need for more robust detection methods.We intend to evaluate the performance of one of the state-of-the-art deepfake detection models not included in Zi et al.'s study [2] on the WildDeepFake dataset. Through this evaluation, we aim to identify the model's limitations and areas for improvement. Additionally, we propose to develop a confidence interval wrapper for the chosen model. This wrapper will provide a probabilistic estimate of deepfake authenticity.

Nicely done! Didn't know that there is a Khodabakhsh as well!

VIII. Approach

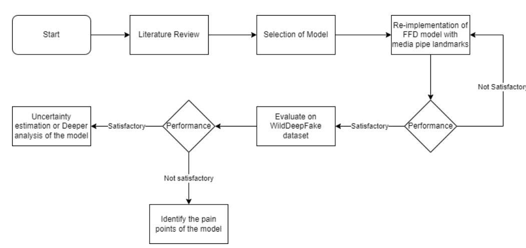

Figure 1. Approach domain diagram.

As seen in the Figure 1, the project commenced with an extensive literature review of deepfake detection models. We delved into the research papers of prevalent deepfake detection models, including Meso4[12], Xception[22], Face-Xray[23], FFD[24] etc. Additionally, we examined the papers associated with common datasets employed for evaluating these models, such as FaceForensics and FaceForensics++[22]. Following the literature review, we opted to reimplement the FFD model[24]. We selected this model due to its innovative attention map technique that effectively pinpoints the manipulated areas in a forged image. This allows the network to concentrate its efforts on the specific manipulated regions rather than processing the entire image. The effectiveness of this attention map method has been demonstrated through superior performance compared to the model without the attention map and publicly available code. Among the models presented in[2], the FFD model consistently ranks among the top three in terms of average AUC for both within-domain and cross-domain evaluation. The availability of the FFD model's code facilitated its implementation and evaluation. The original authors utilized landmarks derived from InsightFace. However, we intend to employ landmarks generated by MediaPipe[26] and reimplement the FFD model on FaceForensics++ dataset[22]. We choose MediaPipe over InsightFace because InsightFace seems to be an older method compared to MediaPipe. We hypothesize that the landmarks from MediaPipe will be more suitable for our FFD model. If the performance of the reimplemented model proves satisfactory, we aim to evaluate it on the Wild Deepfake dataset[2].

This dataset poses a significant challenge, as it has not been utilized for evaluating the FFD model in previous studies. Moreover, several traditional deepfake detection models struggle with the WildDeepFake dataset, highlighting its difficulty. In the event that the performance on the original datasets falls short of expectations, we will explore strategies to enhance the FFD model's performance on those datasets. The Wild Deepfake dataset closely resembles real-world deepfake images encountered on the internet. Therefore, the model's performance on this dataset will guide our decision to either identify its shortcomings or develop an uncertainty estimation[25] of the model. This approach has not been explored in previous works. The literature review revealed that deepfake detection models often exhibit limitations in detecting deepfakes that have undergone advanced manipulation techniques.

We believe that open-source collaboration is essential for advancing the field of deepfake detection. Therefore, we plan to make our re-implemented FFD model and associated code publicly available. This will enable researchers and developers to build upon our work and further improve the capabilities of deepfake detection models.

Our team embarked on this project by conducting a comprehensive literature review on Deepfake Detection methods. We then selected the FFD model, which met our project's criteria and demonstrated superior performance among the models evaluated in[2]. We re-implemented the FFD model and evaluated its performance on the WildDeepfake dataset. This evaluation allowed us to identify the model's strengths and weaknesses. Additionally, we may develop an uncertainty estimation approach to quantify the model's uncertainty in its predictions or a deeper analysis of the model. Currently, both team members have worked on the same tasks.

IX. Architecture

We reviewed existing Deepfake Detection methods and selected the FFD[24] deepfake detection model for our research. We evaluated the model's performance on the FaceForesics++ dataset[22]. The FFD model employs a novel attention map technique that effectively detects manipulated facial images.

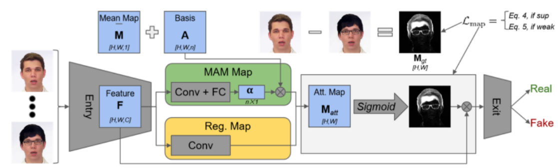

Figure 2. Architecture of FFD model. Source: [24]

Illustrated in Figure 2 is the overall structure of the FFD model. Digital facial manipulation techniques encompass four main categories: (1) Expression swap, (2) Identity Swap, (3) Attribute Manipulation, and (4) Entire Face Synthesis. The authors emphasize the importance of automatically identifying manipulated regions in facial images, which can be altered either completely or partially. Accurately estimating spatial information in manipulated images enhances the network's decision-making process. To achieve this, the model automatically generates an image-specific attention map. We choose MediaPipe over InsightFace because InsightFace seems to be an older method compared to MediaPipe. We hypothesize that the landmarks from MediaPipe will be more suitable for our FFD model.

The attention-based layer receives a convolutional feature map F, $F \in R^{HXWXC}$ as input, where R, H, W, and C represent the batch size, height, width, and number of channels, respectively. An attention map: $M_{att} = \phi(F) \in R^{HXW}$ is generated from this input. The primary objective is to develop a method for estimating the function $\phi(F)$.

Now, after estimating the attention Map M_{att}, a sigmoid function is applied and a refined feature map F' is produced: F'= F $\odot$ Sigmoid(M_{att}), where

$\odot$ means the element wise multiplication. This is the part where the attention map is fed back to the network to aid in binary classification of whether the image is forged or not. Two different methods were proposed to learn the attention map: (1) A Manipulation Appearance Model (MAM) and (2) Direct Regression. In the Manipulation Appearance Model (MAM) method, the task of learning the manipulated map is brought down to the task of learning α in $M_{att} = \bar{M} + A\alpha$, where $\bar{M}$ is the predefined average manipulation map, and A is the predefined basis function of maps. A is calculated by applying Principal Component Analysis (PCA) to 100 ground truth manipulation maps computed from FaceApp[27]. The feature weights F are passed through one additional convolutional layer and one additional fully connected layer before the Mean Map and the basis function of maps are fed in. In the regression based approach, the attention map is learned by a single convolutional layer or a series of convolutional layers: $F \xrightarrow{f} M_{att}$ where f is the convulational layer.

In order to train the network, either a pre-trained backbone or a backbone with uninitialized weights is used. Two different types of losses are considered $L_{classifier}$ and L_{map}. The overall loss is given by: $L_{total} = L_{classifier} + \lambda * L_{map}$, where λ denotes the proportion of weight given to the loss for the map. L_{map} is calculated as the L1 norm of the difference between M_{att} and $M_{groundtruth}$. For images that are not manipulated at all, $M_{groundtruth}$ would be a map of zeros. For completely manipulated images, $M_{groundtruth}$ would be a map of ones. For partially manipulated images, $M_{groundtruth}$ would have ones in those pixels where the image has been manipulated, and zero anywhere else.

Finally, everything is converted to grayscale and normalized to have values in [0, 1]. $L_{classifier}$ is the standard binary softmax classification loss function. In order to evaluate the performance of the model, two different types of metrics are needed: One for detection, and one for assessing the correctness of the attention map. For the attention map, the standard Area Under the Curve (AUC) of ROC will be used. For evaluating the correctness of the attention map, 4 different metrics are proposed: Pixel Wise Binary Classification Accuracy (PBCA), Intersection over Union (IoU), Cosine Similarity between 2 maps, and a new novel metric proposed by dang liu et al[24] Inverse Intersection Non Containment (IINC). This is another novel evaluation metric introduced by dang liu et al[24].

$$IINC = \frac{1}{3-|U|} * \{0 \text{ if } \bar{M_{gt}} = 0 \text{ and } \bar{M_{att}} = 0, 1 \text{ if } \bar{M_{gt}} = 0 \ xor \ \bar{M_{att}} = 0, 2 - \frac{|I|}{|M_{att}|} - \frac{|I|}{|M_{gt}|} \ otherwise\}$$

Here, I and U denote the intersection and union between the ground truth map and the predicted map respectively. Further, $\bar{M}$ and $|M|$ denote the mean and the L1 norm of the Map M respectively. Dang liu et al[24] have shown that IINC is a more robust metric for comparing two attention maps, compared to the other metrics like pixel wise binary classification accuracy and Intersection over Union.

Now, in the experimentation phase, we plan to use XceptionNet as the backbone in the above architecture. The State of the art XceptionNet has depthwise separable convolution layers. Basically, the model given by dang liu et al[24] is made by inserting the novel attention based layer between block 4 and block 5 of the network. Two different approaches for estimating the attention map, viz regression based approach and the Manipulation Appearance Model(MAM) will be considered. We will try to check the performance of each method and choose the one that best suits our data. In training the model, dang liu et al[24] created a new dataset called Diverse Fake Face Dataset (DFFD). They created this dataset from various diverse sources. In our approach, we plan to utilize the Face Forensics ++ dataset[22].

The FaceForensics++ dataset is a comprehensive collection of facial videos that have been manipulated using various deepfake techniques. It is considered one of the most challenging and realistic benchmarks for evaluating deepfake detection algorithms. The dataset contains over 1000 original video sequences manipulated with four different deepfake methods: DeepFakes, Face2Face, FaceSwap, and NeuralTextures. This diversity of manipulation techniques makes it challenging for deepfake detection models to generalize to different types of deep fakes. The videos in the dataset have high resolution and frame rates, ensuring that the facial features are clearly visible for analysis. This quality is crucial for accurate deepfake detection. The deepfakes in the dataset are highly realistic and often indistinguishable from real videos. This realism makes the dataset a more challenging and relevant benchmark for evaluating real-world deepfake detection scenarios. The data will be stored and computation will be done using RIT research computing facilities. RIT's research computing facilities likely provide access to high-performance computing clusters. Deepfake detection, especially when working with large datasets or complex models, can benefit significantly from the parallel processing capabilities offered by HPC resources. An appropriate train-validation-test split will be done on the FaceForensics++ dataset after consultation with the advisors.

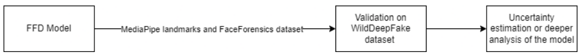

Figure 3. Uncertainty estimation or Deeper analysis of the model.

After re-evaluating the FFD model with MediaPipe landmarks and on the FaceForensics++ dataset and after achieving satisfactory performance on the metrics suggested above, we proceed to the WildDeep Fake dataset[2]. Both the deep fake and real videos in the Wild Deepfake have been collected from the internet, and compared to current existing virtual Deepfake Detection datasets, WildDeepfake contains more diverse scenes, more people in each scene. W. Zi, et al[2] show that many common deepfake detectors such as XceptionNet, MesoNet etc perform well on the common datasets but perform poorly on the WildDeepfake dataset. We plan to identify the pain points of the model and test it on the Wild Deepfake dataset. Once that is done, we aim to quantify the predictive uncertainty as in[25]. Figure 3 shows the same.

Coming to different softwares and libraries used, we will primarily use the Python programming language. More specifically, we will be using PyTorch for most of the Deep Learning work in addition to libraries like Numpy, scikit-learn etc. There are multiple reasons why we chose PyTorch as the main framework for our project. Firstly, PyTorch provides a rich ecosystem of libraries and tools like torchvision. Deep fake detection models can vary in complexity, and the dynamic computation graph in PyTorch accommodates this flexibility. It is well-suited for models that require dynamic structures or involve recurrent neural networks. PyTorch is known for its simplicity and ease of use. Its Pythonic syntax and dynamic computation graph make it more accessible for both beginners and experienced developers.

X. Ethical Statement

All images utilized in this work are sourced from publicly available datasets and are appropriately attributed through proper citations. However, whether or not explicit consent was taken from the participants whose images are used is unclear. Now, we are going to reimplement the Facial Forgery Detection (FFD) model and validate it on the Wild Deep fake[2] data set. Although our ultimate aim behind this is to prevent the spread of misinformation through deep fakes, malicious users could use the same technology to create more sophisticated and realistic deep fakes. Since our code is publicly available, malicious users could find vulnerabilities in our code and launch adversarial attacks on our model. Also, if our model goes on to be used in more complex systems, a false positive could subject a person in a video to unwarranted scrutiny or suspicion. Similarly, false negatives could make people to lose faith in deepfake detection technology.

What about the potential misuse of your detection methods? - We have updated the Ethical statement

XI. Preliminary Results

The FaceForensics++ data which we are using to train the re-implemented FFD model includes manipulated videos generated using a variety of techniques to create realistic and challenging deepfake scenarios. It includes videos with facial manipulations generated using a range of techniques, such as DeepFake, Face2Face, FaceSwap, and NeuralTextures.

We have examined a subset of the dataset and observed that the average image resolution is 200 x 200 pixels. There are different categories of videos in the FaceForensics++ dataset such as FaceForensics++ aligned, FaceForensics++ Enhanced, FaceForensics++ HifiFace, FaceForensics++ segmented. The FaceForensics++ directory is the one we examined.

There two categories of videos in this datset: Real and Fake videos. These videos are split into images frame by frame. For both real and fake, there are two subcategories: Those derived from Youtube videos and those derived from actors. For the real category, there are 684 videos for training from Youtube, which amount to 353,837 images and 195 videos of actors for training, which amount to 203,107 images. For the fake category, there are 1907 videos manipulated using Deepfake techniques for training. These amount to 1,404,136 images showcasing diverse scenarios ranging from individuals walking down indoor hallways to joyfully embracing one another. Further, for every image, there are 468 landmarks in the form of x,y,z coordinates.

References

[1] F. Yan, Y. Yu, J. Huang, and X. Zhang, "DeepfakeBench: A Comprehensive Benchmark of Deepfake Detection," arXiv:2307.01426, 2023.

[2] W. Zi, J. Huang, F. Liu, S. R. Li, and Y. Yu, "WildDeepfake: A Challenging Real-World Dataset for Deepfake Detection," arXiv:2101.01456, 2021.

[3] Y. Li, M. Lyu, W. Wang, B. Li, and H. Farid, "In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking," in *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security (WIFS)*, 2018, pp. 1-10.

[4] A. Haliassos, B. Verouden, N. Beebe, and C. Busch, ”Lips Don’t Lie: A Generalisable and Robust Approach to Face Forgery Detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 1587-1596.

[5] Jia Wen Seow, Mei Kuan Lim, Raphaël C.W. Phan, Joseph K. Liu, A comprehensive overview of Deepfake: Generation, detection, datasets, and opportunities, Neurocomputing, Volume 513, 2022, Pages 351-371, ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2022.09.135.

[6] N. Liv, D. Greenbaum, Deep fakes and memory malleability: False memories in the service of fake news, AJOB Neuroscience 11 (2) (2020) 96–104

[7] A. Sengur, Z. Akhtar, Y. Akbulut, S. Ekici, U. Budak, Deep Feature Extraction for Face Liveness Detection, in: 2018 International Conference 1535 on Artificial Intelligence and Data Processing (IDAP), IEEE, 2018, pp. 1–4.

[8] A. Khodabakhsh, R. Ramachandra, K. Raja, P. Wasnik, C. Busch, Fake face detection methods: Can they be generalized?, in: 2018 international conference of the biometrics special interest group (BIOSIG), IEEE, 2018, pp. 1–6.

[9] S. A. Gowda, N. Thillaiarasu, Investigation of comparison on modified cnn techniques to classify fake face in deepfake videos, in: 8th International Conference on Advanced Computing and Communication Systems 1560 (ICACCS), Vol. 1, 2022, pp. 702–707.

[10] N. S. Ivanov, A. V. Arzhskov, V. G. Ivanenko, Combining deep learning 1545 and super-resolution algorithms for deep fake detection, in: IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), 2020, pp. 326–328.

[11] Q. ul ain, N. Nida, A. Irtaza, N. Ilyas, Forged face detection using ela and deep learning techniques, in: 2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST), 2021, pp. 271–275. doi: 10.1109/IBCAST51254.2021.9393234.

[12] D. Afchar, V. Nozick, J. Yamagishi, I. Echizen, Mesonet: a compact facial video forgery detection network, in: 2018 IEEE International Workshop on Information Forensics and Security (WIFS), IEEE, 2018, pp. 1–7.

[13] Y. Wang, A. Dantcheva, A video is worth more than 1000 lies. Comparing 3DCNN approaches for detecting deepfakes, in: 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG), IEEE, 2020, pp. 515–519.

[14] A. Mitra, S. P. Mohanty, P. Corcoran, E. Kougianos, A novel machine learning based method for deepfake video detection in social media, in: 2020 IEEE International Symposium on Smart Electronic Systems (iSES)(Formerly iNiS), IEEE, 2020, pp. 91–96.

[15] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, Matthias Niessner; Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 1-11

[16] H. H. Nguyen, J. Yamagishi and I. Echizen, ”Capsule-forensics: Using Capsule Networks to Detect Forged Images and Videos,” ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 2019, pp. 2307-2311, doi: 10.1109/ICASSP.2019.8682602.

[17] J. Cao, C. Ma, T. Yao, S. Chen, S. Ding and X. Yang, ”End-to-End Reconstruction-Classification Learning for Face Forgery Detection,” 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 4103-4112, doi: 10.1109/CVPR52688.2022.00408.

[18] UCF: “Uncovering Common Features for Generalizable Deepfake Detection”, Zhiyuan Yan, Yong Zhang, Yanbo Fan, Baoyuan Wu, The School of Data Science, The Chinese University of Hong Kong, Shenzhen (CUHK-Shenzhen), China 2 Tencent AI Lab

[19] Consistent Representation Learning for Face Forgery Detection” Yunsheng Ni, Depu Meng, Changqian Yu, Chengbin Quan, Dongchun Ren, Youjian Zhao, Tsinghua University, University of Science and Technology of China, Meituan

[20] Generalizing Face Forgery Detection with High-frequency Features Yuchen Luo, Yong Zhang, Junchi Yan, Wei Liu

[21] 2019. Deep Fake Detection Dataset. https://ai.googleblog.com/2019/09/ Contributing-data-to-deepfake-detection.html

[22] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner. 2019. FaceForensics++: Learning to Detect Manipulated Facial Images.

[23] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, B. Guo 2020. Face X-ray for More General Face Forgery Detection.

[24] Stehouwer, J., Dang, H., Liu, F., Liu, X., Jain, A.K. (2019). On the Detection of Digital Face Manipulation. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 5780-5789.

[25] Bao, W., Yu, Q., Kong, Y. (2021). Evidential Deep Learning for Open Set Action Recognition. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 13329-13338.

[26] MediaPipe, https://developers.google.com/mediapipe/solutions/vision/face_landmarker, Accessed: November 22, 2023

[27] FaceApp. https://faceapp.com/app, Accessed: November 22, 2023