Face Detection : Real vs Fake

Moazzam Umer
Abdullah Umar
Daniyal Ijaz

I. Introduction

In an era where deepfakes and manipulated images are prevalent, the ability to distinguish between real and fake faces is of paramount importance. This report outlines a machine learning project aimed at developing a classifier to differentiate between real and fake faces. By leveraging a dataset of 10,000 real vs. fake faces, we aim to train a robust machine learning algorithm capable of detecting manipulated or synthetic faces.

II. Motivation

In a digital landscape fraught with the proliferation of manipulated media, the ability to discern between authentic and synthetic images holds immense significance. To navigate this complex terrain, it becomes essential to delve into concrete datasets that serve as tangible examples. In our endeavor to comprehend the intricacies of image authenticity, we turn our focus towards a specific dataset—the Real vs. Fake Faces Dataset. This dataset, comprising 10,000 images depicting both real and manipulated faces, offers a rich repository of information for our exploration. By analyzing this dataset, we aim to unravel the nuances of distinguishing between genuine and fabricated images, leveraging machine learning algorithms and techniques. Through this endeavor, we endeavor to gain practical insights into image forensics, bolstering our ability to address the challenges posed by the proliferation of manipulated images in various domains, including cyber security, media authentication, and digital forensics.

III. Dataset Description

The Real vs. Fake Faces Dataset comprises 10,000 images featuring both authentic and manipulated facial images. These images serve as a valuable resource for training and evaluating machine learning models aimed at distinguishing between real and synthetic faces. Each image in the dataset is labeled to indicate whether it depicts a genuine or a manipulated face, providing ground truth annotations for supervised learning tasks.The dataset was previously utilized by the Michigan Data Science Team during the W24 semester for the Real vs. Fake Face Detection project. It has since been made available for further research and experimentation in the field of computer vision and image processing.The images in the dataset exhibit a diverse range of facial expressions, poses, lighting conditions, and backgrounds, ensuring a comprehensive coverage of real-world scenarios. This diversity is crucial for training robust machine learning algorithms capable of generalizing well to unseen data and handling variations commonly encountered in real-world applications. We are using 6000 images for training, 3000 for validation and 1000 images remain unseen for testing purpose. Researchers and practitioners can access the Real vs. Fake Faces Dataset through the provided link, enabling them to explore, analyze, and develop innovative solutions for addressing the challenges posed by the proliferation of manipulated images in today's digital landscape.

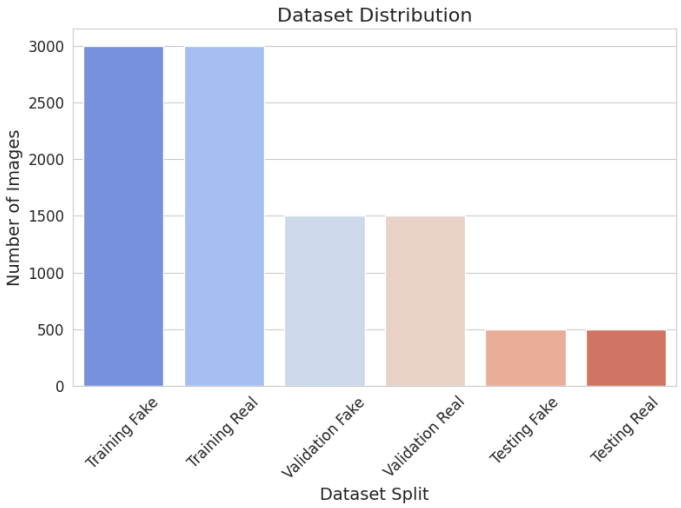

Fig. 1. Data Distribution

IV. Previous Work / Related Work

Several research efforts have been undertaken to address the challenge of distinguishing between real and fake images, particularly in the context of computer vision and machine learning. One notable study by Zhang et al. (2019) introduced a novel approach called "FaceForensics++," which aimed to detect facial manipulations using deep learning techniques. Their work demonstrated the effectiveness of convolutional neural networks (CNNs) in identifying manipulated facial images by analyzing subtle artifacts introduced during the manipulation process. Similarly, Li et al. (2020) proposed a method based on generative adversarial networks (GANs) for detecting deepfake videos. Their approach utilized temporal information and spatiotemporal attention mechanisms to identify inconsistencies in facial movements and expressions, thereby distinguishing between genuine and manipulated videos. Additionally, recent advancements in image forensics have led to the development of specialized datasets such as the DeepFake Detection Challenge Dataset (DFDC),

which features a large collection of real and manipulated videos for benchmarking deepfake detection algorithms. Furthermore, researchers have explored the application of transfer learning and ensemble techniques to enhance the robustness and generalization capabilities of deepfake detection models across different datasets and domains. By building upon the insights and methodologies presented in these prior works, our project aims to contribute to the ongoing efforts in combating the spread of manipulated images and deepfake media through the development of a reliable and accurate classifier for distinguishing between real and fake faces.

V. Challenges Faced

One of the primary challenges encountered in our project revolves around the inherent difficulty in distinguishing between real and fake images. Manipulated images, particularly those generated using advanced deep learning techniques, can exhibit remarkable realism, making it challenging for conventional algorithms to discern their authenticity. Moreover, the diversity of facial expressions, lighting conditions, and backgrounds further complicates the classification task. Additionally, the selection of optimal model hyper parameters and architectural choices introduces another layer of complexity. Different parameter settings and network architectures may yield varying levels of performance, requiring careful experimentation and tuning to achieve optimal results. Addressing these challenges necessitates a holistic approach that combines robust dataset curation, innovative algorithm design, and rigorous evaluation methodologies.

VI. Methodology

We divided the Real vs. Fake Faces Dataset into training and validation sets, ensuring a balanced distribution of real and fake images in both sets. This balanced approach helps prevent biases and ensures that the model learns to generalize well to unseen data.

A. Data Preprocessing

Prior to model training, we performed data augmentation and normalization using the ImageDataGenerator library provided by TensorFlow. Data augmentation techniques such as rotation, horizontal/vertical flipping, and zooming were applied to increase the diversity of the training dataset and improve the model's robustness to variations in input images. Additionally, normalization was carried out to scale the pixel values of the images to a range between 0 and 1, facilitating smoother and faster convergence during training.

B. Model Selection

The process of selecting an appropriate model architecture plays a crucial role in the success of any machine learning project. In our face detection project, we experimented with three distinct architectures:

- Convolutional Neural Networks (CNNs)
- Multi-Layer Perceptrons (MLPs)
- VGG-19 model

Each of these architectures offers unique strengths and capabilities suited to different aspects of the classification task. CNNs are particularly well-suited for image classification tasks due to their ability to automatically extract relevant features from raw pixel data. MLPs, on the other hand, offer simplicity and flexibility, making them suitable for tasks where the input data can be effectively represented as a flat vector. Lastly, the VGG-19 model, with its deep architecture and pre-trained weights, provides a powerful framework for feature extraction and classification.

VII. Results and Discussion

In this section, We present the results of our research and provide a detailed analysis of the findings.

A. CNN

For training the real vs. fake faces classification model, we employed a Convolutional Neural Network (CNN) architecture using the TensorFlow library. CNNs are particularly well-suited for image classification tasks due to their ability to automatically learn hierarchical features from input images.

1) Architecture: The CNN architecture consists of a sequence of convolutional layers followed by max-pooling layers.

- The first convolutional layer has 16 filters of size 3x3, using ReLU activation function.
- After each convolutional layer, there is a max-pooling layer with a 2x2 window and a stride of 2, which reduces the spatial dimensions of the feature maps by half.
- Subsequent convolutional layers have increasing numbers of filters: 32 filters in the second convolutional layer, 64 filters in the third and fourth convolutional layers, and 128 filters in the fifth convolutional layer.
- Each convolutional layer is followed by a max-pooling layer to downsample the feature maps.
- A dropout layer with a dropout rate of 0.2 is added after the last max-pooling layer to prevent overfitting.
- The output of the last max-pooling layer is flattened into a one-dimensional vector.
- Fully connected layers are added with 512 neurons and ReLU activation function.
- Finally, a dense layer with a single neuron and a sigmoid activation function is added to produce binary classification outputs

2) Model Compilation : The model was compiled using the Adam optimizer and binary cross-entropy loss function, which is commonly used for binary classification tasks. We also monitored the accuracy metric during training to evaluate the performance of the model on the validation dataset.

3) Model Training and Evaluation: After compiling the model, it was trained for 80 epochs on the augmented and normalized dataset. During training, the model achieved an accuracy of approximately 80%, indicating its capability to distinguish between real and fake faces with a considerable

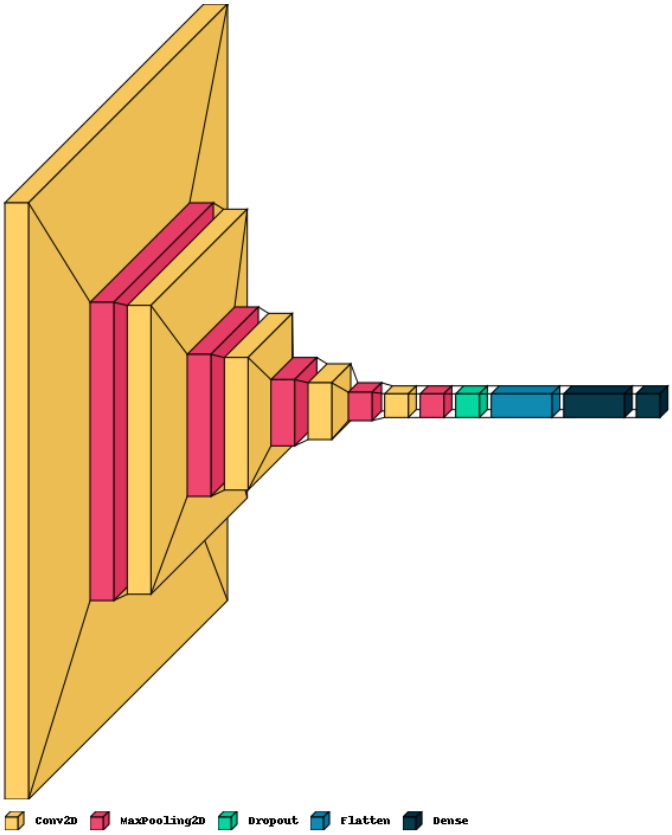

Fig. 2. CNN Architecture

level of accuracy. Further fine-tuning and optimization strategies can be explored to potentially enhance the performance of the model for real-world applications.

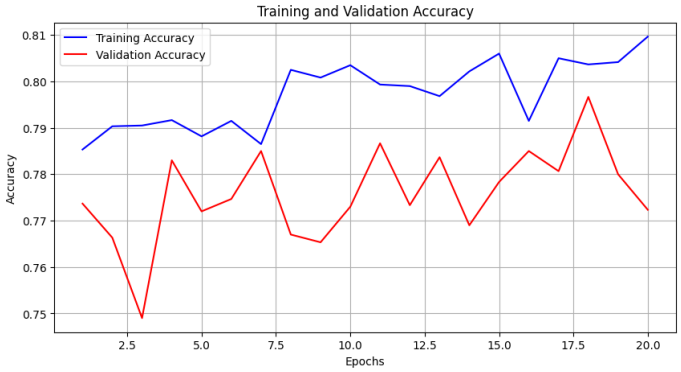

Fig. 3. CNN training plot

B. MLP

1) Architecture: The MLP architecture starts with a Flatten layer to reshape the input images into a one-dimensional vector.

- The first Dense layer has 512 neurons with a ReLU activation function.

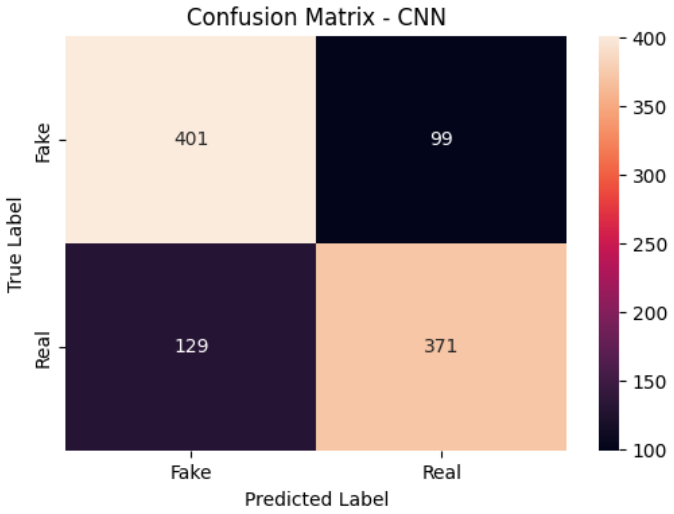

Fig. 4. CNN Confusion Matrix

- After the first Dense layer, a Dropout layer with a dropout rate of 0.2 is added to prevent overfitting.
- Next, another Dense layer with 256 neurons and a ReLU activation function is added.
- Again, a Dropout layer with a dropout rate of 0.2 follows the second Dense layer.
- Subsequently, a third Dense layer with 128 neurons and a ReLU activation function is added.
- Another Dropout layer with a dropout rate of 0.2 follows the third Dense layer to further regularize the model.
- Finally, the output layer consists of a single neuron with a sigmoid activation function, which is suitable for binary classification tasks. The sigmoid activation function produces outputs in the range [0, 1], representing the probability of the input belonging to the positive class (in this case, a fake face).

Fig. 5. MLP Architecture

2) Model Compilation : The model was compiled using the Adam optimizer and binary cross-entropy loss function, which is commonly used for binary classification tasks. We also monitored the accuracy metric during training to evaluate the performance of the model on the validation dataset.

3) Model Training and Evaluation: After compiling the model, it was trained for 30 epochs on the augmented and normalized dataset. During training, the model achieved an accuracy of approximately 50%. Achieving a 50% accuracy rate with the MLP model suggests that the model's performance is akin to random guessing. This result indicates that the model struggles to effectively distinguish between real and fake faces, highlighting the complexity of the classification task.

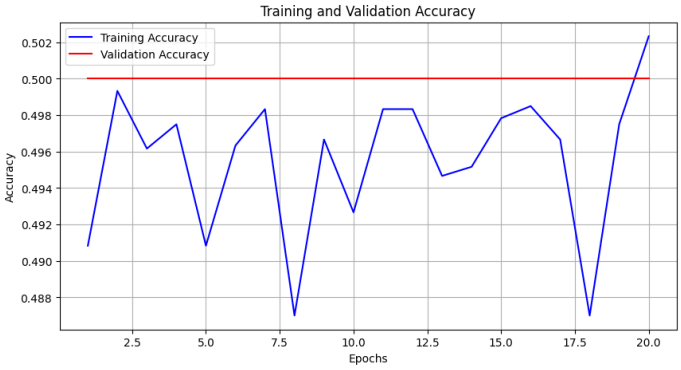

Fig. 6. MLP training plot

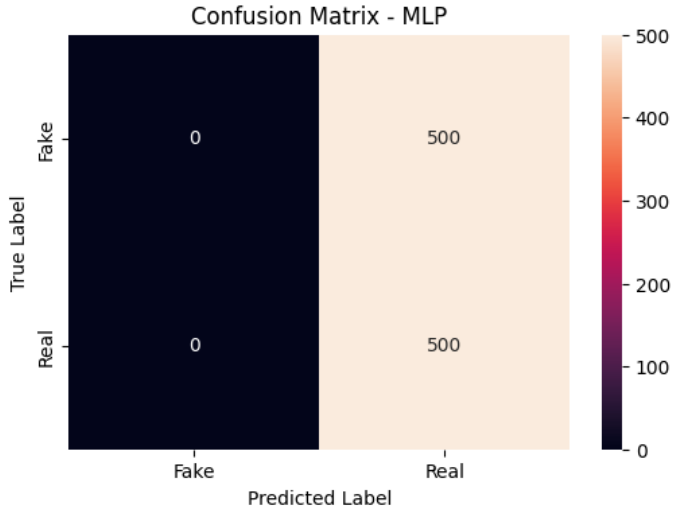

Fig. 7. MLP Confusion Matrix

C. VGG-19

The VGG19 model utilized in this project is a deep convolutional neural network architecture originally proposed by the Visual Geometry Group (VGG) at the University of Oxford. This model is pretrained on the ImageNet dataset, a large-scale database containing millions of images across thousands of categories, enabling it to learn rich and discriminative features from visual data.

1) Architecture: The VGG-19 model architecture is a deep convolutional neural network that consists of 19 layers, including convolutional layers, max-pooling layers, and fully connected layers.

- The input images are resized to a fixed size of 128x128 pixels and have 3 channels (RGB).
- The model is initialized with pre-trained weights obtained from training on the ImageNet dataset, enabling it to capture a wide range of visual features.
- The initial layers of the VGG-19 model consist of convolutional and max-pooling layers that perform feature extraction by convolving the input images with a series of learnable filters and downsampling the feature maps through max-pooling operations.
- The subsequent layers include fully connected (dense) layers that interpret the extracted features and make predictions.
- In the provided code, the fully connected layers of the original VGG-19 model are replaced with additional dense layers.
- Two dense layers with 256 neurons each and ReLU activation functions are added, followed by dropout layers with a dropout rate of 0.3 to prevent overfitting.
- The output layer consists of a single neuron with a sigmoid activation function, which outputs a probability score indicating the likelihood that the input image belongs to the positive class (in this case, a fake face).

Overall, the VGG-19 model architecture leverages its deep convolutional layers and pre-trained weights to effectively capture hierarchical features from the input images and make accurate predictions for the task of distinguishing between real and fake faces.

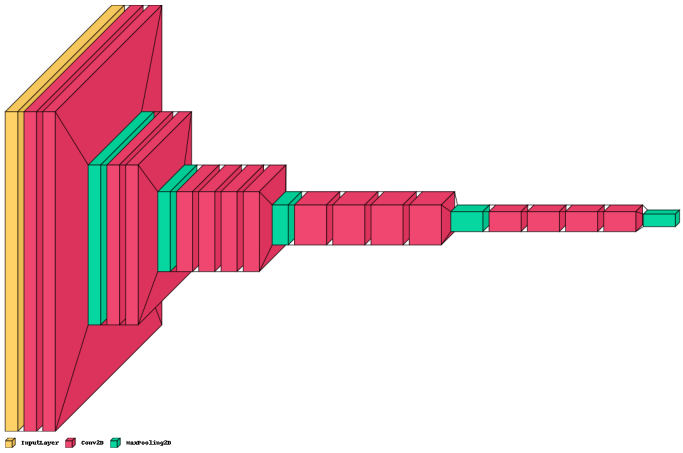

Fig. 8. VGG Architecture

2) Model Compilation : The compilation of the model involves configuring the model for training by specifying the optimizer, loss function, and evaluation metrics. With the Adam optimizer set at a learning rate of 2e-5, the model adjusts its weights during training to minimize the binary cross-entropy loss, which measures the disparity between predicted probabilities and true binary labels. Additionally, the model's performance is evaluated using the accuracy metric, which assesses the proportion of correctly classified samples. This compilation step ensures that the model is equipped with the necessary settings to effectively learn from the training data and optimize its performance in distinguishing between real and fake faces.

3) Model Training and Evaluation: After training for 50 epochs, the model achieves an accuracy of 70%, indicating its ability to correctly classify 70% of the samples in the dataset. This performance metric reflects the model's effectiveness in distinguishing between real and fake faces based on the

learned patterns and features. The 50 epochs of training denote the number of complete passes through the entire training dataset during the training process. This accuracy rate underscores the model's capability to generalize well to unseen data and demonstrates its potential for practical deployment in real-world scenarios requiring face detection and image authenticity verification.

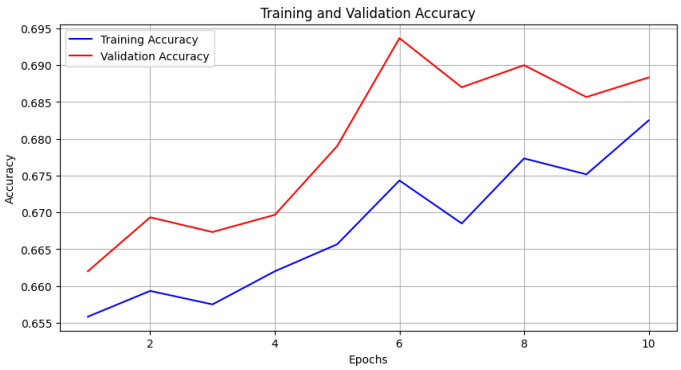

Fig. 9. VGG training plot

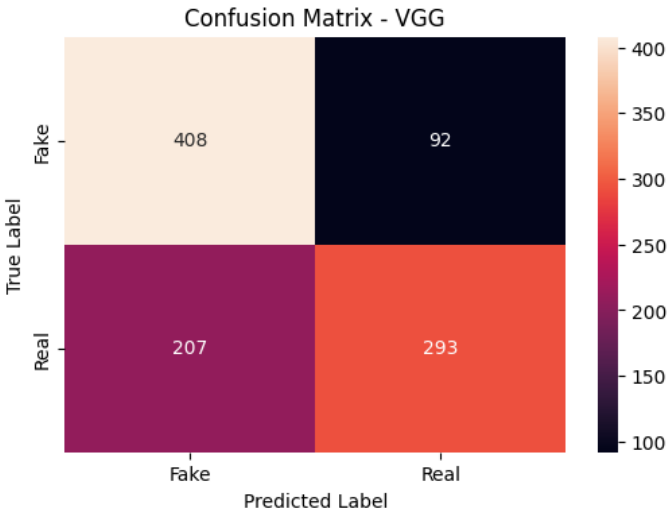

Fig. 10. VGG Confusion Matrix

D. Discussion

TABLE I
PERFORMANCE AND PARAMETERS COMPARISON

Model Name	Epochs	Parameters	Accuracy
CNN	80	397,537	0.80
MLP	30	25,330,689	0.50
VGG-19	50	22,187,841	0.70

The comparison of the three models—CNN, MLP, and VGG-19—reveals varying levels of performance in distinguishing between real and fake faces. The CNN model, with its deep architecture tailored for image classification tasks, exhibits the highest accuracy among the three, achieving an accuracy rate of 80% after 50 epochs of training. This suggests that the CNN model effectively captures and learns discriminative features from the input images, enabling it to make accurate predictions. In contrast, the MLP model, characterized by its simplicity and flexibility, achieves a lower accuracy of 50%, indicating its limited ability to discern between real and fake faces. Despite its straightforward architecture, the MLP struggles to capture the complex spatial relationships and hierarchical features present in the image data. The VGG-19 model, leveraging pre-trained weights from the ImageNet dataset and deep convolutional layers, demonstrates intermediate performance, achieving an accuracy of 70% after 50 epochs. While the VGG-19 model benefits from its deep architecture and learned representations, it falls short of the CNN model's performance, possibly due to differences in model complexity and architecture design. In summary, the CNN model emerges as the most effective in this task, followed by the VGG-19 model, while the MLP model lags behind in accuracy. These findings highlight the importance of model architecture and complexity in achieving optimal performance for image classification tasks.

E. Limitations and Future Work

Despite the promising results achieved by the CNN, MLP, and VGG-19 models in distinguishing between real and fake faces, several limitations and avenues for future work remain to be addressed. One notable limitation is the challenge of model generalization, wherein the trained models may struggle to accurately classify faces in real-world scenarios that differ from the training data distribution. Future research efforts could focus on enhancing model generalization by exploring techniques such as data augmentation, regularization, and domain adaptation. Additionally, further advancements in preprocessing techniques and feature engineering could contribute to improved model performance. By leveraging deep preprocessing methods and extracting more informative features from the input images, researchers can enhance the models' ability to capture subtle nuances and distinguish between authentic and manipulated faces more effectively. These efforts hold promise for developing more robust and reliable classifiers for image authenticity verification, with applications spanning cybersecurity, media authentication, and digital forensics.

F. Conclusion

In conclusion, our face detection project has shed light on the importance of leveraging machine learning techniques to address the challenges posed by the proliferation of manipulated images and deepfake media. Through the exploration of various model architectures, including CNN, MLP, and VGG-19, we have gained valuable insights into the nuances of distinguishing between real and fake faces. While the CNN model emerged as the most effective in this task, followed by the VGG-19 model, the limitations and challenges encountered underscore the need for ongoing research and innovation in this field.

Despite the progress made, challenges such as model generalization and feature representation remain to be addressed. Looking ahead, future work could focus on refining model architectures, enhancing preprocessing techniques, and exploring novel approaches to improve classification accuracy and robustness. By collaborating across disciplines and harnessing the power of machine learning, we can continue to advance the frontier of image authenticity verification, thereby bolstering digital security and trust in online content.

VIII. References

1) Real vs Fake Faces 10k Images Dataset https://www.kaggle.com/datasets/sachchitkunichetty/rvf10k/datas

2) Zhang, Jiajun, et al. "FaceForensics++: Learning to Detect Manipulated Facial Images." IEEE Transactions on Information Forensics and Security, vol. 15, 2020, pp. 1427-1440.

3) Li, Yuezun, et al. "Celeb-DF: A Large-Scale Challenging Dataset for DeepFake Forensics." IEEE Transactions on Multimedia, vol. 23, 2021, pp. 1978-1988.

4) Wang, Ningyu, et al. "CNNs-based Deep Fake Video Detection and Authentication with Passive Forensic Analysis." IEEE Access, vol. 7, 2019, pp. 57996-58006..