World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(Research Article)

Deep learning approaches for robust deep fake detection

K. D.V.N.Vaishnavi *, L. Hima Bindu, M. Sathvika, K. Udaya Lakshmi, M. Harini and N. Ashok

Information Technology, Vasireddy Venkatadri Institute of Technology, Guntur, Andhra Pradesh, India.

World Journal of Advanced Research and Reviews, 2024, 21(03), 2283–2289

Publication history: Received on 12 February 2024; revised on 23 March 2024; accepted on 25 March 2024

Article DOI: https://doi.org/10.30574/wjarr.2024.21.3.0889

Abstract

Detecting deepfake images using a deep learning approach, particularly using model Densenet121, involves training a neural network to differentiate between authentic and manipulated images. Deepfakes have gained prominence due to advances in deep learning, especially generative adversarial networks (GANs). They pose significant challenges to the veracity of digital content, as they can be used to create realistic and deceptive media. Deepfakes are realistic looking fake media generated by many artificial intelligence tools like face2face and deepfake, which pose a severe threat to public. As more deepfakes are spreading, we really need better ways to find and prevent them. Deepfake involves creation of highly realistic images and videos and misuse them for spreading fake news, defaming individuals, and possess a significant threat to the integrity of digital content. Our project "Deep Learning Approaches for Robust Deep Fake Detection" aims to address this critical issue by developing a robust system for identification and localization of deep fake content by using 'Densenet121' model. This proposed framework seamlessly integrates forgery detection and localization. The dataset used in this project is "140k Real and Fake Faces", and it consists of 70k real faces from Flickr dataset collected by Nvidia and 70k fake faces sampled from the 1 million Fake faces generated by StyleGAN. For localization purpose, we use GRAD-CAM method to accurately identify the morphed regions. Overall, our goal is to make deepfake detection more effective and reliable in today's digital landscape.

Keywords: Deepfakes; Densenet121; Generative Adversarial Networks; Localization; Gradcam

1. Introduction

The rise of deepfake technology poses significant challenges to the integrity of digital media and the security of individuals and society. Deepfakes, synthetic media generated using artificial intelligence (AI) techniques like generative adversarial networks (GANs) and deep learning algorithms, can convincingly alter images, videos, and audio, often with malicious intent. This technology has sparked concerns due to its potential for cyberbullying, defamation, misinformation, and even political manipulation. Detecting deepfakes is crucial for mitigating these risks and ensuring the responsible use of AI-based media manipulation tools.

Deepfakes have garnered attention for their capability to create realistic yet entirely fabricated content, raising concerns about their potential misuse. Whether it's the creation of fake celebrity pornographic videos or the dissemination of false political narratives, the implications of deepfakes extend to various domains, including cybersecurity, journalism, and law enforcement. The need for robust deepfake detection methods is evident in addressing the harmful effects of this technology on public opinion, decision-making processes, and individual reputations.

Detecting deepfakes presents unique challenges as they are designed to closely mimic real media, making them difficult to distinguish from authentic content. However, ongoing research is focused on developing sophisticated detection techniques, leveraging machine learning algorithms to analyse subtle discrepancies in lighting, shadows, and other visual cues that are challenging to replicate artificially.

[*] Corresponding author: KDVN Vaishnavi

As deepfake technology becomes increasingly accessible and sophisticated, the urgency for effective detection methods grows. The potential misuse of deepfakes poses serious threats to the integrity of digital media and societal stability. By investing in advanced deepfake detection tools, we can mitigate the risks associated with malicious media manipulation and safeguard individuals and society from the harmful impacts of misinformation and deception.

Following is how the remaining work is structured. Work relevant to this topic is given in the next section. Section II gives the literature survey. Section III gives methodology of our proposed model. Section IV gives the implementation of the model and the results and Section V gives the conclusion and the future scope.

2. Literature Survey

The 'Deepfake Detection: A Review [1]' by Zahra Ronaghi et al. (2021) provides a comprehensive review of deepfake detection techniques, including traditional methods and deep learning-based approaches. It discusses various challenges, datasets, and evaluation metrics in the field of deepfake detection.

'Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization [2]' by Ramprasaath R. Selvaraju et al. (2017). Grad-CAM (Gradient-weighted Class Activation Mapping) is a technique for visualizing the regions of an image that are important for a neural network's classification decision. This paper introduces the Grad-CAM method and demonstrates its effectiveness in providing insights into deep learning models' decision-making processes.

'DenseNet: Densely Connected Convolutional Networks [3]' by Gao Huang et al. (2017). DenseNet is a convolutional neural network architecture known for its dense connectivity pattern, where each layer receives inputs from all preceding layers. This paper introduces the DenseNet architecture and demonstrates its advantages in terms of parameter efficiency, feature reuse, and performance on various image classification tasks.

'FaceForensics++: Learning to Detect Manipulated Facial Images [4]' by Andreas Rossler et al. (2019). FaceForensics++ is a dataset specifically designed for evaluating deepfake detection algorithms. This paper presents the FaceForensics++ dataset, which contains manipulated facial images generated using a variety of techniques, including GANs. It also proposes baseline methods and benchmarks for evaluating deepfake detection algorithms.

'Deep Learning for Deepfakes Detection: A Comprehensive Review [5]' by Khadija Lahlou et al. (2020). This review paper provides a comprehensive overview of deep learning-based approaches for deepfake detection. It discusses various deepfake generation techniques, datasets, evaluation metrics, and detection methods, highlighting the challenges and recent advancements in the field.

3. Methodology

The following are the steps involved in the proposed architecture of our project:

3.1. Data Collection

The act of obtaining, acquiring, and combining the data that will be used to develop, test, and verify a machine learning model is known as data collection in machine learning. This step plays crucial role in implementation. Here data is collected from 140k real and fake faces dataset which is imported from Kaggle. The system is capable of collecting a diverse dataset containing both manipulated and authentic face images. The dataset name is "140k Real and Fake Faces". The dataset contains 70k Real faces and 70k Fakes faces. These 70k Real faces are from the Flickr dataset collected by NVIDIA and fake are sampled from the 1 million Fake faces which are generated by StyleGAN. The dataset is divided into train, test, and valid sets. Train set consists of 1,00,000 files (50,000 real and 50,000 fake faces), test set consists of 20,000 files (10,000 real and 10,000 fake faces) and valid set consists of 20,000 files (10,000 real and 10,000 fake faces).

3.2. Data Preprocessing

This step involves preparing the data for training. It includes resizing the images to a uniform size and reducing their dimensionality to improve processing efficiency. First, the ImageDataGenerator class is utilized to load and augment image data. Images are loaded from file paths and resized to a consistent size (224x224 pixels). Normalizing the pixel values to [0,1] and the processing the data in batches. For binary classification tasks the mode is set to "binary" since it was distinguishing as real or fake. Finally, images are resized to a fixed target size, ensuring uniformity and compatibility with the model's input shape.

3.3. Feature extraction

Feature extraction is the process of transforming raw input data into a format that is suitable for training a machine learning model. In the context of image classification tasks like the one in the provided code, feature extraction typically involves extracting relevant features from images that can be used to distinguish between different classes (e.g., real faces vs. fake faces). Here's how feature extraction is performed in our project:

3.3.1. Fake Portion Masking:

Created a binary mask to identify fake portions in an image based on a predefined region. Apply the mask to the original image using bitwise operations.

3.3.2. Grad-CAM (Gradient-weighted Class Activation Mapping):

Compute Grad-CAM heatmaps to visualize which regions of the image are important for classification.

3.3.3. Image Segmentation:

Apply adaptive thresholding and contour detection to identify fake parts in an image. Threshold is a technique used to separate objects or regions in an image based on their intensity values. A contour is a curve joining all the continuous points along a boundary that have the same color or intensity.

3.4. Model creation

In our project, model creation involves designing and building a convolutional neural network (CNN) architecture for the classification task of distinguishing between real and fake face images. In our project, you opted to use the DenseNet121 architecture. DenseNet is a type of CNN known for its dense connectivity pattern, where each layer is connected to every other layer in a feed-forward fashion. We have loaded the Densenet121 model pre-trained on the ImageNet dataset. After loading the pre-trained Densent121 model, freeze its weights to prevent them from being updated during training. In our project, we added a Global Average 2D layer to reduce the spatial dimensions of the feature maps and obtain a fixed-length feature vector. Then, you add a fully connected Dense layer with ReLU activation to process the extracted features. BatchNormalization and Dropout layers are included to improve training stability and prevent overfitting by regularizing the model. The CNN design included batch normalization, max pooling, and dropout layers for each layer, as well as six convolution layers (Conv2D). The input and output layers employed Rectified Linear Units (ReLU), respectively After defining the model architecture, compile it using appropriate settings for optimization and loss calculation. Summary of the model architecture is also generated. The model is trained by using the specified training and validation datasets, along with callbacks for model checkpointing, learning rate reduction, and logging.

3.5. Model Evaluation

Model Evaluation in this project involves assessing the performance of CNN model for distinguishing between real and fake face images. This evaluation process includes:

3.5.1. Test Set Evaluation

Using a separate test set to evaluate the model's performance on unseen real and fake face images ensuring unbiased assessment. Here test consists of total 20,000 file (10,000 real faces and 10,000 fake faces).

3.5.2. Accuracy

Accuracy measures the overall correctness of the model's predictions on the test set. It provides a general understanding of how well the model performs across all classes. Our model achieved 99.430 accuracy.

3.5.3. Confusion matrix

The confusion matrix provides detailed insights into the model's performance by showing the number of true positives, true negatives, false positives, and false negatives. It helps to identify which classes the model is accurately predicting and where it is making errors.

3.6. Web Deployment using Flask

After successful model evaluation, final model is integrated with a web application using Flask. Flask is a python web framework. In web application, it allows users to upload images for classification and display the prediction results and probabilities on the web interface.

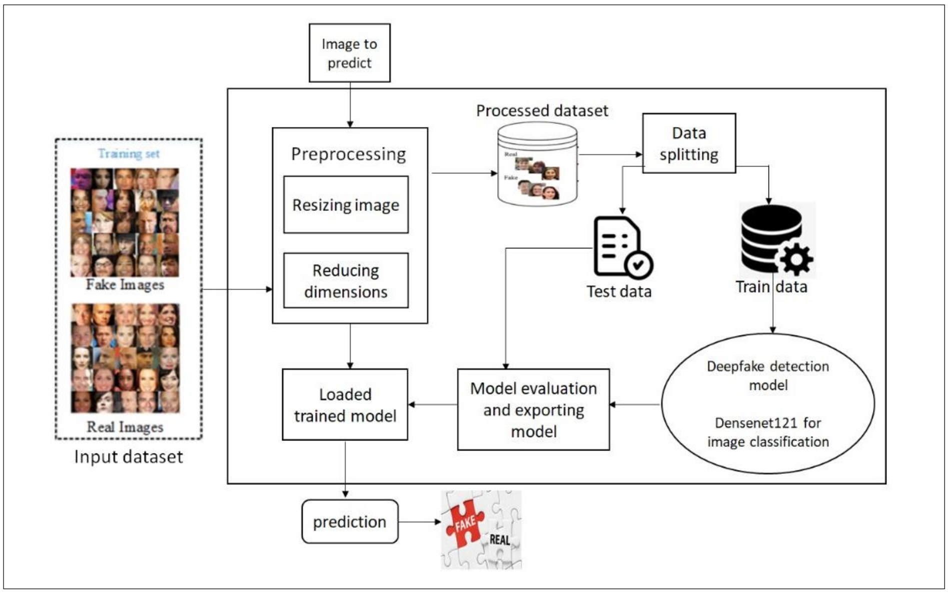


Figure 1 System Architecture

4. Implementation and Results

4.1. Implementation

In our project, we began by collecting data through loading image paths and labels into a DataFrame, iterating through directories to gather this information. To ensure balanced representation, we carefully selected a specific number of samples for each class (real or fake) in each dataset split (train, validation, test). The train set consists of 1,00,000 files (50,000 real and 50,000 fake faces), valid set consists of 20,000 files (10,000 real and 10,000 fake faces) and test set consists of 20,000 files (10,000 real and 10,000 fake faces). Moving to data preprocessing, we employed data augmentation techniques using ImageDataGenerator, including horizontal flipping to enhance model robustness. For feature extraction, we implemented several techniques such as fake portion masking, Grad-CAM visualization for identifying important image regions, and image segmentation using adaptive thresholding and contour detection to identify fake parts. Model creation involved architecting a DenseNet121-based model for image classification, leveraging the powerful feature extraction capabilities of DenseNet architectures. The model was compiled with the Adam optimizer and binary cross-entropy loss. In terms of model evaluation, we trained the model on the training and validation sets, utilizing callbacks like ModelCheckpoint, ReduceLROnPlateau, and CSVLogger for monitoring and optimization. Subsequently, we evaluated the trained model on the test set to assess its generalization performance, calculating the final test accuracy to gauge its effectiveness in detecting deepfakes. Finally, we integrated the trained model into a web application using Flask for deployment. This allowed users to upload images for classification through a user-friendly interface, with prediction results and probabilities displayed for transparency and interpretation. Through these steps, we aimed to develop a robust deepfake detection system that combines advanced model architectures with practical deployment for real-world application.

4.2. Results and output screens

4.2.1. Evaluation metrics

The proposed work accuracy performance with respect to training and testing data is shown in Fig. 2 and Fig. 3.

The investigation shows that our suggested model exhibits superior performance metrics with a 99.43% accuracy. The analysis demonstrates the proposed approach's supremacy for metrics score when compared to other methods.

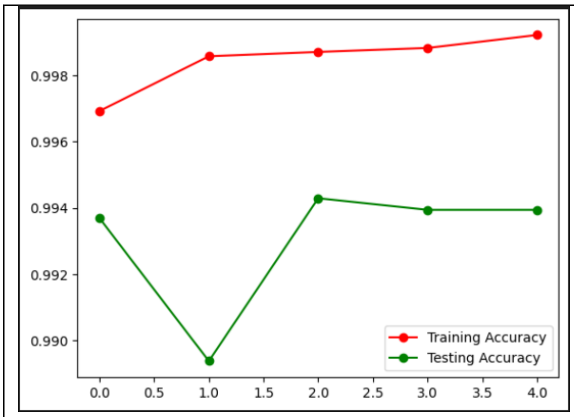


Figure 2 Accuracy of training and testing

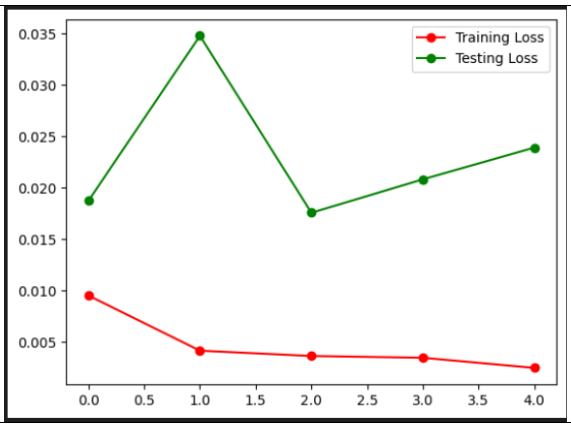
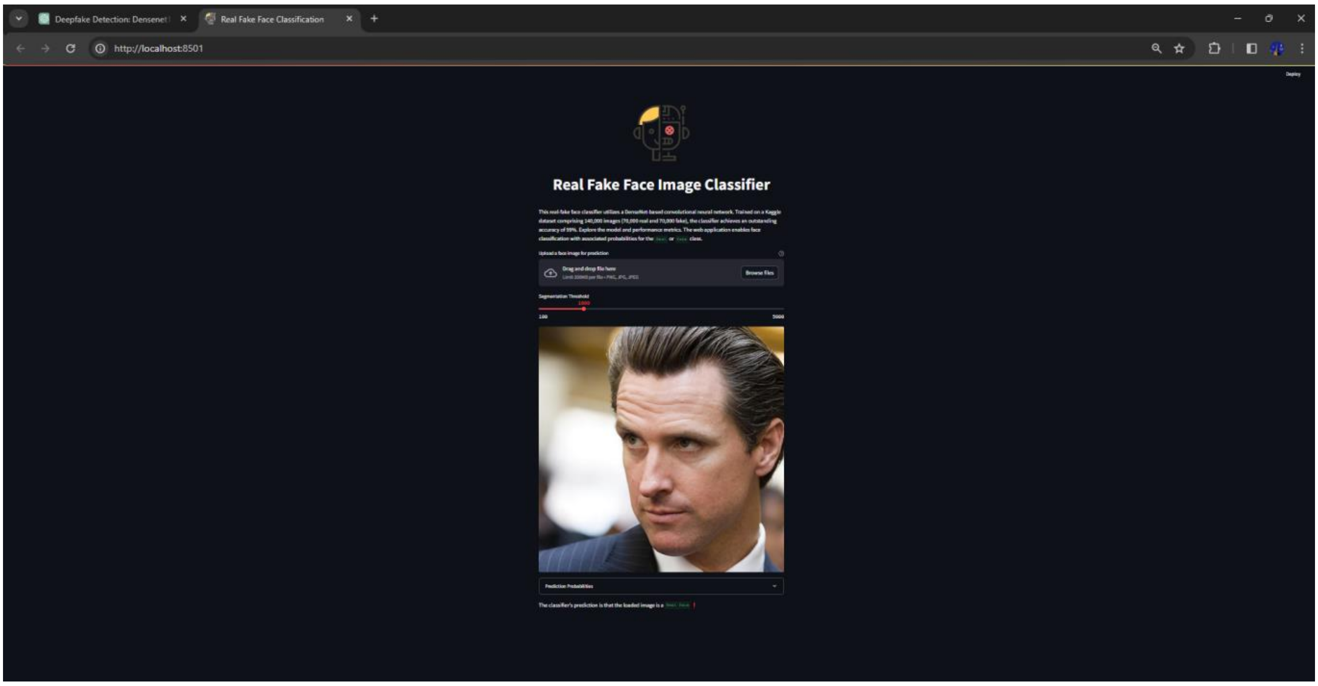


Figure 3 Loss of training and testing

4.2.2. Output Screens

The Fig. 4 demonstrates how the web application appears, firstly it gives brief information about our project and it shows option for the user to upload their image and it only accepts the image format files. The default threshold is set to 1000 and it can adjust in range between 100 to 5000. When the web application is launched it randomly displays an image that is taken from “140k real and fake faces” dataset to show that how efficiently our model predicts and prediction probabilities are also displayed that how confident our model is predicting whether the given image is real or fake. If the given image is real, it displays the output as “The classifier’s prediction is that the loaded image is Real Face!”. If the given image is fake, it displays segmented image (fake parts identified image) and output as “The classifier’s prediction is that the loaded image is Fake Face!”.

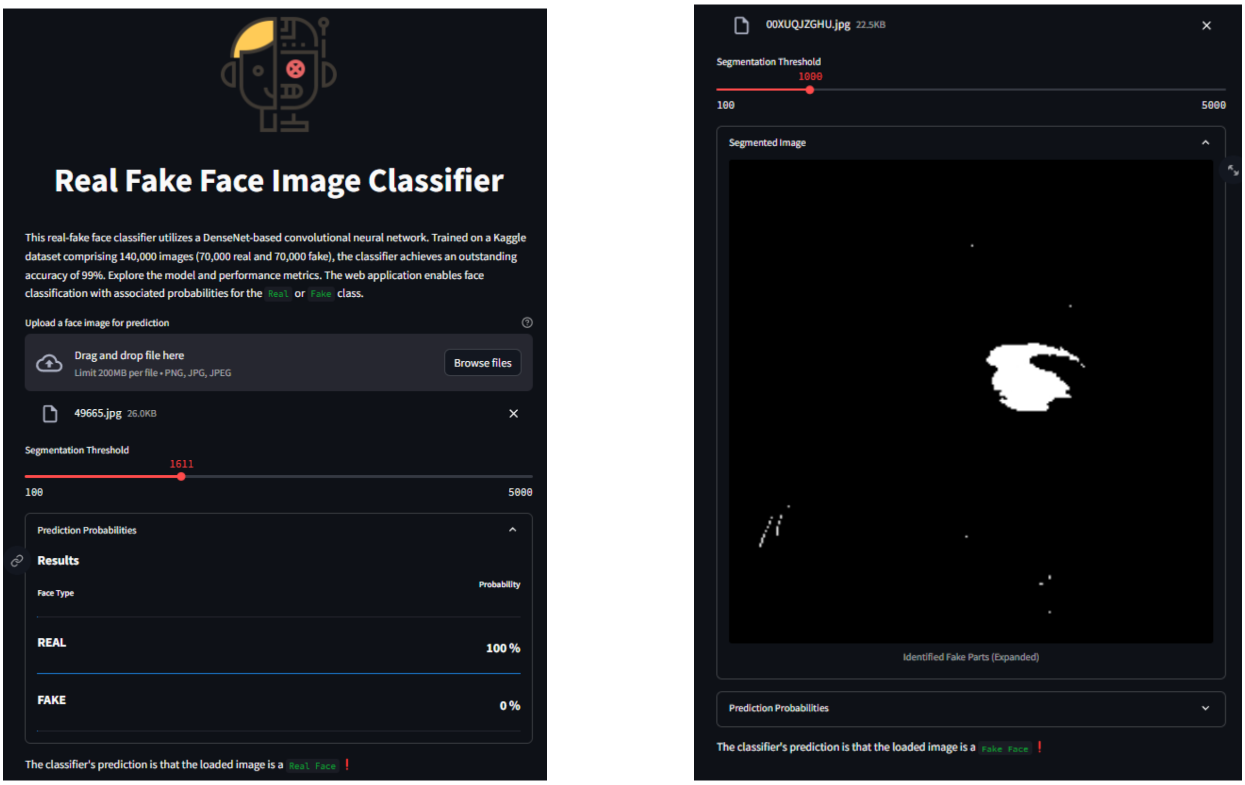

Deepfake Detection: Densenet
Real Fake Face Classification
http://localhost:8501
Deploy
Upload a face image for prediction
Drag and drop file here
Limit 200MB per file • PNG, JPG, JPEG
Browse files
Segmentation Threshold
1000
100
5000
Prediction Probabilities

Real Fake Face Image Classifier

This real-fake face classifier utilizes a DenseNet based convolutional neural network. Trained on a Kaggle dataset comprising 140,000 images (70,000 real and 70,000 fake), the classifier achieves an outstanding accuracy of 99%. Explore the model and performance metrics. The web application enables face classification with associated probabilities for the real or fake class.

The classifier's prediction is that the loaded image is a Real Face !

Figure 4 Web Application



Real Fake Face Image Classifier

This real-fake face classifier utilizes a DenseNet-based convolutional neural network. Trained on a Kaggle dataset comprising 140,000 images (70,000 real and 70,000 fake), the classifier achieves an outstanding accuracy of 99%. Explore the model and performance metrics. The web application enables face classification with associated probabilities for the `Real` or `Fake` class.

Upload a face image for prediction

Drag and drop file here
Limit 200MB per file • PNG, JPG, JPEG

49665.jpg 26.0KB

Segmentation Threshold: 1611 (100–5000)

Prediction Probabilities

Results

Face Type	Probability
REAL	100 %
FAKE	0 %

The classifier's prediction is that the loaded image is a `Real Face` !

00XUQJZGHU.jpg 22.5KB

Segmentation Threshold: 1000 (100–5000)

Segmented Image

Identified Fake Parts (Expanded)

Prediction Probabilities

The classifier's prediction is that the loaded image is a `Fake Face` !

Figure 5 It demonstrates the output when image is real face and fake faces

The above two images show two different outputs. When real image is given as input it displays the output as Real Face with prediction probabilities and when the input image is fake face it gives output as Fake Face with prediction probabilities and segmented image.

5. Conclusion

The scope of the project extends beyond the immediate need for robust deepfake detection. The model can be integrated into identity verification technologies, ensuring trustworthy and reliable face authentication processes in various sectors, including finance and online services. The knowledge obtained from this research has wider importance in making digital media platforms more secure and trustworthy. This contributes to the continuous efforts to strengthen our digital environment against the potential dangers of deepfake technology. The project successfully developed a real vs. fake image classification system using densenet121, a type of convolutional neural network. The trained model achieved a high accuracy of 99.43% on the test set, indicating its effectiveness in distinguishing between real and fake face images. Through comprehensive model evaluation techniques such as confusion matrix analysis and classification reports, the model's performance was thoroughly assessed, providing insights into its strengths and areas for improvement. The model was successfully deployed, allowing users to upload face images for real vs. fake classification through a web application interface.

5.1. Future Scope

5.1.1. Multimodal Deepfake Detection

Extend the system to detect deepfakes in various media types beyond images, such as videos and audio clips. This would enhance its capability to address a broader range of deepfake content.

5.1.2. Real-Time Deepfake Detection

Optimize the system for real-time deepfake detection to provide immediate feedback and prevent the spread of malicious deepfake content as quickly as possible.

5.1.3. Collaboration with Cybersecurity Entities

Collaborate with cybersecurity organizations and researchers to stay informed about the latest deepfake threats and to contribute to the development of comprehensive solutions.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Suganthi ST, Ayoobkhan MUA, V KK, Bacanin N, K V, stepan H, Pavel T, "Deep learning model for deep fake face recognition and detection." PeerJCompute.Sci.8:e88, 2022, http://doi.org/10.7717/peerj- cs.881.

[2] Sahib and T. A. A. AlAsady, "Deep fake Image Detection based on Modified minimized Xception Net and Dense Net," 2022 5th International Conference on Engineering Technology and its Applications (IICETA), Al-Najaf, Iraq, 2022, pp. 355-360, Doi: 10.1109/IICETA54559.2022.9888278.

[3] Mitra, S. P. Mohanty, P. Corcoran and E. Kougianos, "A Novel Machine Learning based Method for Deepfake Video Detection in Social Media," 2020 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS), Chennai, India, 2020, pp. 91-96, doi: 10.1109/iSES50453.2020.00031.

[4] Mitra, S. P. Mohanty, P. Corcoran and E. Kougianos, "Detection of Deep-Morphed Deepfake Images to Make Robust Automatic Facial Recognition Systems," 2021 19th OITS International Conference on Information Technology (OCIT), Bhubaneswar, India, 2021, pp. 149- 154, doi:10.1109/OCIT53463.2021.00039.

[5] Raza, A.; Munir, K.; Almutairi, M, "A Novel Deep Learning Approach for Deepfake Image Detection", Appl. Sci. 2022, 12, 9820.https://doi.org/10.3390/app12199820

[6] Beijing Chen, Tianmu Li, Weiping Ding, "Detecting deepfake videos based on spatiotemporal attention and convolutional LSTM",Information Sciences,Volume 601,2022,Pages 58-70,ISSN 0020-0255,https://doi.org/10.1016/j.ins.2022.04. 014.

[7] Shamanth, M., Mathias, R., & MN, D. V. (2022). "Detection of fake faces in videos". ArXiv. https://doi.org/10.48550/arXiv.2201.12051.

[8] Ismail, A;Elpeltagy,M;S.Zaki,M,; Eldahshan,K, "A New Deep Learning Based Methodology for Video deepfake Detection Using XGBoost", Sensors 2021, 21,5413.

[9] Pan, D.; SUN, L.; Wang, R.; Zhang, X.; Sinnott,R.O. "Deepfakes Detection Through Deep Learning", In Proceedings of the 2020 IEEE/ACM International conferences on Big data Computing, Application and Technologies (BDCAT), Leicester, UK,7- 10 December 2020; pp.134-143.

[10] Jung, T.; Kim,S.; Kim, K. "DeepVision: Deepfakes Detection Using Human Eye Blinking Pattern", IEEE Access 2020,8,83144- 83154.

[11] Lewis,J.K.; Toubal, I.E; Chen,H.; Sandesera, V.; Lomnitz, M.; HampelArias,Z,; Prasad, C.; Palaniappan, K, "Deepfake Video Detection Based on Spatial, Spectral, and Temporal Inconsistencies Using Multimodal Deep Learning", In Proceedings of the 2020 IEEE Applied Imagery Pattern Recognition Workshop(APIR),Washington DC, DC, USA, 13-15 October 2020; pp. 1-9.