International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Video Deepfake Detection System Using Deep Learning

Shreyas Chavan
CSE(AI&ML)
Finolex Academy of Management and Technology Ratnagiri,India
shreyasvilas0606gmail.com

Om Jakkar
CSE(AI&ML)
Finolex Academy of Management and Technology Ratnagiri, India
omjakkar3gmail.com

Amey Karekar
CSE(AI&ML)
Finolex Academy of Management and Technology Ratnagiri,India
ameykarekar9@gmail.com

Yash Kumbhar
CSE(AI&ML)
Finolex Academy of Management and Technology Ratnagiri, India
yashkumbhar17@gmail.com

Prof. Akshay Shetye
CSE(AI&ML)
Finolex Academy of Management and Technology Ratnagiri, India
akshay.shetye@famt.ac.in

***Abstract*— This comprehensive study delves into the dynamic geography of deep literacy operations, fastening on the burgeoning realm of deep fakes. Deep literacy has seamlessly integrated into fields like natural language processing, machine literacy, and computer vision, giving rise to innovative operations. still, the swell in deep fakes, sophisticatedly manipulated videos images, has come a pressing concern. The unrighteous operations of this technology, similar as fake news, celebrity impersonations, fiscal swindles, and vengeance porn, pose significant pitfalls in the digital realm. Particularly, public numbers like celebrities and politicians are largely susceptible to the Deep fake discovery challenge. This exploration totally assesses both the product and discovery aspects of deep fakes, employing different deep literacy algorithms, including InceptionResnetV2, VGG19, CNN, and Xception. The evaluation is done by using Kaggle deep fake dataset, highlights Xception as the most accurate among the algorithms studied. As vicious uses of deep fakes escalate, the imperative for robust discovery mechanisms intensifies to guard against implicit societal consequences**

Keywords*— *DeepFake, Deep Learning, DeepFake Detection Algorithm, Kaggle Dataset*, *Accurate Prediction

I. INTRODUCTION

A. Background:

In the area of technological evolution, the human face stands as the quintessential identifier, a uniquely personal aspect that defines individuals. However, the relentless progress of face synthesis technologies has ushered in an era where the sanctity of facial identity is under siege. The escalating threat of unauthorized face alterations looms large, prominently exemplified by the advent of deep fake technology. This artificial intelligence marvel facilitates the seamless superimposition of one person's face onto another's, all executed without the subject's consent. As a consequence, the profound implications of deep fakes extend far beyond the realm of mere privacy invasion, posing a significant challenge to the integrity of personal and public security.

At the forefront of combating this emerging menace is the application of deep learning—a formidable and versatile technology that has left an indelible mark across various domains, including machine learning, computer vision, and natural language processing. With its ability to discern intricate patterns and nuances, deep learning has emerged as a potent tool in the ongoing battle against deep fake proliferation. This transformative technology holds the promise of fortifying our defenses against malicious face alterations, exemplifying the innovative synergy between artificial intelligence and the safeguarding of our fundamental right to identity and security.

B. Problem Definition

detection mechanism to discern authentic content from manipulated ones. By scrutinizing the intricacies of deep fakes, the system aspires to contribute to a deeper understanding of their production and distribution. The integration of multiple algorithms ensures a nuanced analysis, while the dataset provides a diverse range of scenarios for effective training and testing. This holistic approach seeks to enhance the overall accuracy and reliability of deep fake detection, mitigating the potential societal consequences of fake news, impersonations, and privacy violations. The proposed system stands as a pivotal step toward fortifying our digital landscape against the malevolent misuse of deep fake technology.

II. LITERATURE SURVEY

Anusha and Srinagesh et al. [1] presented an extensive survey of current state of art machine and deep learning approaches for the detection of deep fake video. Their approach organizes detection methods in large scale by spatial, temporal, and audio-visual features, which provides some insights into the evolution of so-called synthetic media. Although the survey accurately summarises the strengths and shortcomings of existing approaches, it fails to propose any; a framework for real-time detection. This makes it less practical in real-world situations where fast, automated detection is needed. The need for deployable solutions such as mobile or web-based tools has been emphasized by this gap. The current project tackles this issue with real-time and user-friendly deepfake detection mechanisms.

Ahmed et al. [2] a generalized review of visual deepfake detection methods, rate visual deepfake detection methods where they included a diverse range of methods, touch points, algorithms and datasets between measure against synthetic media manipulations. Although this study provides valuable insights into the current detection systems and their limitations, it is mostly an analysis based on existing literature and fails to propose or test novel detection systems. The absence of practical deployment underscores the need for real-time, actionable solutions (e.g., mobile or cloud-based applications) for effective deepfake detection and mitigation at the user level, which this project seeks to address.

Dongre et al. [3] comparatively analyzed the performance on various deepfake detection models, across different datasets and manipulations. This study compares classical machine learning algorithms with deep learning models that can be used based on metrics such as accuracy, robustness, and computational efficiency. It finds that, in general, deep learning approaches, in particular CNN-based models, perform better than traditional ones. However, the study is limited to static performances and does not bear relevance in real-time or resource-limited spaces. They do not suggest any implementation or deployment strategies. This highlights the demand for deployable detection solutions, like the real-time mobile applications addressed in this project.

Singh et al. [4] investigated a CNN-RNN hybrid model for deepfake video detection that extracts spatial and temporal features combining CNN and RNN architecture. They utilize Convolutional Neural Networks (CNNs) for image-level artifacts and Recurrent Neural Networks (RNNs) to examine frame-wise temporal inconsistencies. This approach demonstrates improved detection accuracy over individual models, But the study is only theoretical modeling and controlled experiments. It does not investigate real-time or user-level applications of the model. This is an indication that what the (present project is targeting) must be a practical toolset (like a mobile app) for real life deepfakes detection.

Tianchen Zhao et al. [5] This paper works on the hypothesis that, every image has distinct source features contained which may be preserved even post deepfake generation process and these features can be extracted to authenticate this deep fake data. This detection framework have two key features: pair-wise self-consistency learning (PCL), the learning approach designed to train convolutional neural network (ConvNets) to extract and compare source features. And inconsistency Image Generator (I2G), an approach which generates counterfeit images by mixing facial regions of different identities. The performance of the proposed method was evaluated on seven publicly available deepfake datasets. Our training and experimentation, AUC got to 98.05% through In-dataset Evaluation from a raw 96.45% generalization capability on various datasets improved: AP was raised from 86.03% to 92.18%.

Fernandes et al.[6] The recent development of generative adversarial networks makes it very difficult to detect fake videos which is putting public interests into danger. This study introduce a specialized metric called Attribution-Based Confidence (ABC) to detect deepfake videos.So the "attributed based confidence metric" didn't need the access to the training data, or the training of calibration model on validation data. So, you can predict the ABC metric without training on these features in the model because you always have a trained model. utilization of the confidence metric based on attribution is done to distinguish whether the video is original or fake. Note that the deep learning model is only trained on original videos. Original videos yield confidence values above 0.94 using the ABC metric, which is produced with the trained model.

Apurva Gandhi and Shomik Jain et al. [7] This study employs adversarial distortions to improve deepfake images and deceive deepfake detectors. We created adversarial distortions using the Fast Gradient Sign Method, Carlini and Wagner L2 norm attack in both blackbox and whitebox configurations. Detectors achieved over 95% accuracy on stable deepfakes, but gives less than 27% accuracy on distorted deepfakes. We also examine two enhancements for deepfake detectors: First is Lipschitz regularization, and second is Deep Image Prior (DIP). By applying Lipschitz regularization, we constrain the gradient of the detector with respect to the input, thereby improving its robustness against input distortion. The DIP defense removes distortions using generative convolutional neural networks(GCNN) in an unsupervised learning context. On average, regularization enhanced the detection of distorted deepfakes, yielding a 10% accuracy boost in the black-box case. The DIP defense achieved 95% accuracy on distorted deepfakes while maintaining 98% accuracy in other scenarios on approximately 100 image subsample.

III. EXISTING SYSTEM

The existing system focuses on the analysis of previously deployed deep fake detection algorithms. It extensively explores classic detection methods and contemporary deep learning-based approaches, including Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM). Classic methods often rely on rule-based heuristics, while deep learning methods harness the hierarchical learning capabilities of neural networks to discern intricate patterns

indicative of deep fakes. The study aims to provide a comprehensive understanding of the strengths and limitations of these methods in identifying manipulated content. In contrast, the proposed system takes a proactive stance by integrating cutting-edge deep learning algorithms. The proposed system aims to enhance detection accuracy, contributing to a deeper comprehension of deep fake production and distribution for more effective countermeasures.

Disadvantages:

1. The existing system depends on the traditional methods, potentially limiting adaptability to emerging deepfake techniques.
2. Classic methods depends on rule-based heuristics may struggle with the evolving sophistication of deep fake creation, leading to less accuracy.
3. The existing system may lack a comprehensive understanding of the very complicated patterns and feature characterizing evolving deepfake technologies.

IV. PROPOSED SYSTEM

A. Problem Identification and Gap Analysis:

The proposed system aims to combat the escalating threat of deep fakes through an integrated approach leveraging cutting-edge deep learning algorithms. Utilizing a Kaggle deep fake dataset, our system employs InceptionResnetV2, VGG19, CNN, and Xception algorithms for comprehensive evaluation. The emphasis is on developing a robust

Advantages of proposed system:

1. The proposed system integrates advanced algorithms like InceptionResnetV2 and Xception, enhancing its capability to detect sophisticated deep fake manipulations effectively.
2. Utilizing multiple algorithms facilitates a nuanced analysis, ensuring a robust detection mechanism capable of discerning authentic content from manipulated instances.
3. The system benefits from a Kaggle deep fake dataset, providing a diverse range of scenarios for training and testing, enhancing its adaptability to real-world situations.

The proposed system takes a holistic approach, aiming not only to enhance accuracy but also to deepen comprehension of deep fake production, distribution, and societal consequences, offering a more proactive defense strategy.

B. System Design and Architecture

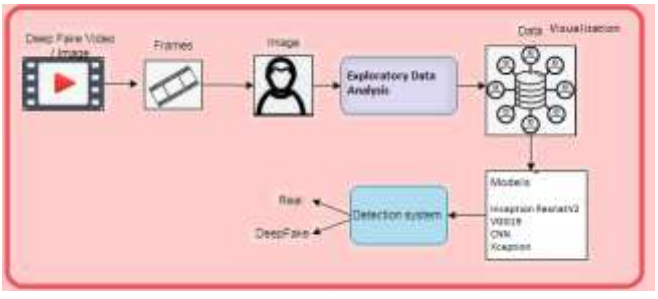

Fig.1.1: System Design

In Above figure, The deepfake video is taken and it is breakdown into frames. That single frames(photos) undergoes EDA process. After that frames are trained under the proposed models. Resulting in detecting whether the video is deepfake or real.

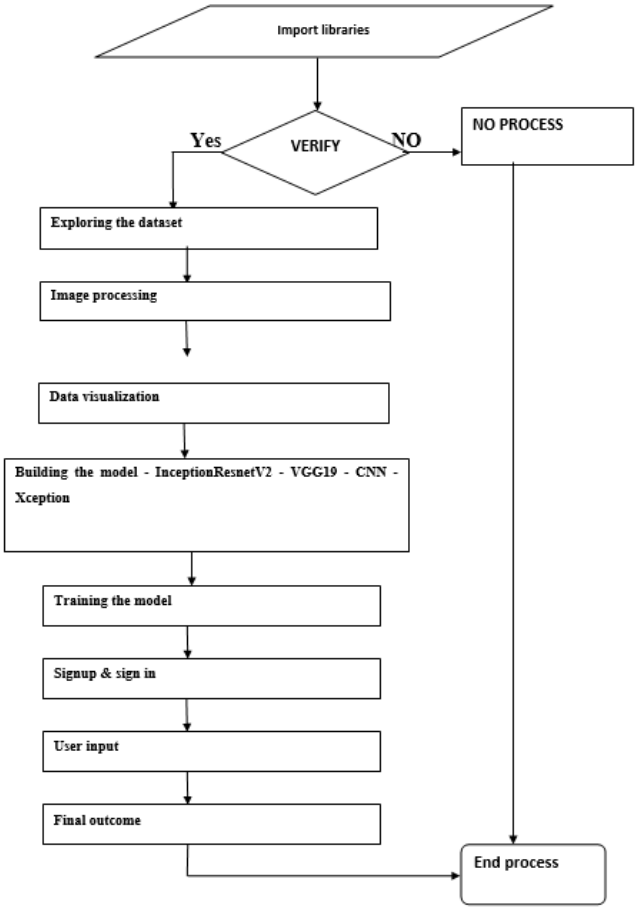

Fig.1.2: Data Flow

1. Import libraries

You will first need to import all the libraries required for image processing, data manipulation, visualization, and modeling. These can be popular libraries you might leverage, such as Matplotlib (graphs) and charts, Pandas (data), more typically OpenCV (image recognition) itself as well as TensorFlow and Keras (deep learning).

2. Verification

It first verifies that all the resources such as the dataset, and other required files are in place and correctly defined before execution. If something is missing or not set up right, it simply halts the process right there to avoid errors and nothing else runs

3. Exploring the Dataset

In this stage, we explore the dataset to have an idea about the structure and the contents. Some major characteristics of the data are the number of real and fake samples, the data format (image, video frames), the presence of any class imbalance or noise etc. At this stage, we will also be doing statistical summaries and distribution plots for each class. Developers can make thoughtful decisions about preprocessing methods, data augmentation, and model selection strategies more easily by exploring the

dataset. This also assists in spotting potential biases that are driving down performance in the model.

4. Image Processing
It performs an important task in processing images to get them ready for model input. This involves their extraction at regular intervals from videos, detecting and aligning the facial regions, resizing the resulting images to a standard size (e.g,, 224x224 pixels), and normalizing the pixel values. Depending on the case, additional processes like the histogram equalization or changing a color space may be used for better visibility of the features. This step of preprocessing helps in keeping the image in uniformity and reduce noise which ultimately allows the deep learning models to extract the features easily during the model training phase.

5. Data Visualization
Journey-After preprocessing data visualization is performed to understand the data-set better. Sample images from real/fake classes, face landmarks, data distributions, and results from image augmentations. This step confirms that preprocessing has worked and the data is balanced and clean. You can also use visualization techniques like t-SNE or PCA to see how features cluster in high-dimensional space, which provides further insights into performance and areas for improvement.

6. Building the Model
In the second step, several deep learning models are defined and prepared for the task of deepfake detection. The scope of architectures to choose from are InceptionResNetV2, which efficiently combines inception modules with residual connections to speed up the learning; VGG19, the deep architecture known for its simple structure; CNN, which signifies custom or shallow network; and Xception that utilizes depthwise separable convolutions to enhance performance. This can be to pretrain these models on something like ImageNet, and then fine-tune using the deepfake dataset. The choice of architecture is determined by accuracy, computational resources, and properties of the dataset.

7. Training the Model
After choosing a model architecture, it will use the processed dataset to be trained. This step can involve tuning hyperparameters like batch size, epochs, learning rate, and optimizer (Adam, SGD, etc.). The training loop makes small updates to the model weights to minimize loss and improve classification accuracy. The process is tracked using metrics (like accuracy, precision, recall, and F1 score) and validation loss to check for overfitting. Checkpointing the seem and saving the model is performed at regular intervals for future prediction or fine-tuning.

8. Signup and Sign-in
It contains a part for the user interface with authentication functionality for secure access. The deepfake detection tool requires all users to register (sign-up) and then log in (sign-in) before using it. This step is particularly crucial when the system is implemented as a web or mobile app. The system ensures a personalized user experience, keeps track of activity with logging plants, and prevents unauthorized access or malicious utilization.

9. User Input
Once logged in, users can upload a video or image for analysis. The uploaded content is passed through the same preprocessing pipeline as the training data and then fed into the trained model. The model analyzes the facial features in the image or video frames and returns a prediction indicating whether the input is real or a deepfake. This stage emphasizes real-time or near real-time processing, especially in interactive applications.

10. Final Outcome
The final output to the user based on the model analysis. The result is usually a label — Real or DeepFake — and a confidence score that indicates how sure the model is. In more sophisticated scenarios, the system can also mark modified areas or give visual explanations (e.g., Grad-CAM) to make the system interpretable. Resultant may be stored to logs or databases for an extra auditing, or analysis.

11. End Process
Finally, terminate the session/process.

C. Tech Stack

1. Programming Language
Python: The system is built predominantly using Python because it provides huge libraries for machine learning, image processing and scientific computing. Python also has a rich ecosystem, making it easy to integrate deep learning models and video processing pipelines.

2. Deep Learning Frameworks
TensorFlow and Keras — Used to create and train neural network models including InceptionResNetV2, VGG19, Xception, and CNN architectures. The Keras high-level API allows for rapid model development and experimentation.

3. Image and Video Processing Tools
PyTorch: This is for researchers who prefer dynamic computation graphs and would like more control while training and debugging a model.
OpenCV: Used for video input/output, extracting frames from videos, face region detection, and image data preprocessing.
These libraries provide accurate face detection and alignment, which is necessary for deep learning models to work with uniform input.

4. Pretrained Model Libraries
Keras Applications: You can access models such as VGG19, Xception, InceptionResNetV2 through Keras Applications, which you can further fine-tune for deepfake detection.

5. Web Interface
The video deepfake detection system consists of a web interface, which is developed using the Flask framework for the backend and uses HTML, CSS and JavaScript for the frontend. Flask enables you to define routing, handle file uploads and communicate with the model, while HTML provides templates for your content layout. CSS is used for styling for a clean and responsive approch while Javascript is not only have form validation its also provide real time feedback. The system, features Signup and Login for basic user authentication to ensure a secure and personalized experience for each user.

D. Proposed Algorithms

1. **Inception ResNetV2:** Inception-ResNet-v2 is a convolutional neural network architecture that builds on the Inception architectures but incorporates residual connections (replacing the filter concatenation stage of the Inception architecture).

2. **VGG19:** VGG-19 stands for Visual Geometry Group this is a convolutional neural network that is 19 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database [1]. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals.

3. **CNN:** A Convolutional Neural Network (CNN) is a type of deep learning algorithm that is particularly well-suited for image recognition and processing tasks. It is made up of multiple layers, including convolutional layers, pooling layers, and fully connected layers.

4. **Xception:** Xception is a convolutional neural network that consist of 71 layers. You can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals.

E. Features

1. Frame-by-Frame Video Analysis
It processes videos frame by frame, enabling it to detect small movements in the face and patterns of light or artifacts appearing in multiple frames that are typical of deepfakes.

2. Scalable and Modular Architecture
It has a modular architecture whereby the preprocessing, model selection, and prediction are separate components, allowing you to scale, upgrade, or integrate into other systems with ease.

3. Facial Feature Extraction
Given that faces are most commonly targeted in deepfake manipulation, the system extracts and analyzes faces from every frame for analysis. It lets the model focus on the important data to achieve accurate detection.

4. Real-Time Prediction
Depending on the implementation, the system can provide real-time detection for short clips or predictions in batch mode for larger datasets, providing flexibility for different use cases.

F. EDA

	Accuracy	Recall	Precision	F1 Score	Sensitivity	\
InceptionResnet V2	0.918432	0.918415	0.918415	0.918415	0.918415	
VGG19	0.796416	0.796435	0.796435	0.796435	0.796435	
CNN	0.987048	0.987036	0.987036	0.987036	0.987036	
Xception	0.993000	0.993002	0.993002	0.993002	0.993002	

	Specificity	MAE	\
InceptionResnet V2	0.918415	<function mae at 0x000002040627E2288>	
VGG19	0.796435	0.32427	
CNN	0.987036	0.019514	
Xception	0.993002	0.009542	

	MSE
InceptionResnet V2	<function mse at 0x000002040627E234B>
VGG19	0.162385
CNN	0.009877
Xception	0.004718

Fig.1.3 Comparison

V. EXPECTED OUTCOMES

The success of deploying a video deepfake detection solution are expected to arrive at accurately classifying whether the video content has been manipulated or synthesized, such as face-swaps and facial reenactments. This approach employs deep learning models such as InceptionResNetV2, VGG19, and Xception to achieve high prediction accuracy. It can segment images/videos and provides predictions in real-time or batch along with confidence scores. The tool is designed to improve media integrity, also helping prevent the spread of misinformation and digital forensics by providing a reliable way to detect tampering of content. It is also easy to use via web interface, which makes it well suited to journalists, social media sites, researchers, and legal investigators.

1. Accurate Detection of Manipulated Videos
Using its deepfake detection AI, the system can analyze various subtle tell-tale features that identify frames as fake; these include inconsistencies in the facial features themselves, how far facial features move, and how frames transition. It succeeds in recognizing common manipulations such as face swaps and expression changes. This is useful for detecting fake videos that may otherwise appear legitimate."

2. High Prediction Accuracy
Using deep learning models including InceptionResNetV2, VGG19 and Xception, the system achieves good classification performance. These models are trained on massive amounts of real and fake media. Due to their architecture, they can even detect very realistic synthetic content.

3. Real-Time or Batch Processing

The system is capable of real-time detection for short-duration analysis or batch processing to cater to boost large volumes of data up to a point. This adaptability enables it to be leveraged for both live applications as well as offline investigations

VI. CONCLUSION

In conclusion, this study underscores the imperative to evolve deep fake detection systems amid the rising sophistication of manipulative technologies. The existing system, while exploring classic and contemporary methods, reveals limitations in adapting to emerging

threats and understanding intricate patterns. Recognizing these shortcomings, the proposed system takes a proactive stance, integrating cutting-edge algorithms such as InceptionResnetV2 and Xception. The use of a Kaggle deep fake dataset ensures a diverse training ground, enhancing the system's adaptability to real-world scenarios. By prioritizing a holistic approach, the proposed system not only aims to improve detection accuracy but also strives to deepen insights into the production and distribution dynamics of deep fakes. This comprehensive understanding is crucial for effective countermeasures against the potential societal ramifications of manipulated content, including fake news, privacy violations, and impersonations. The proposed system stands as a pivotal step toward fortifying our digital landscape against the malevolent misuse of deep fake technology, contributing to a more secure and resilient digital environment for users worldwide.

VII. ACKNOWLEDGMENT

We would like to thank Finolex Academy of Management and Technology for providing the resources and support needed to conduct this research.

VIII. REFERENCE

[1] Tenali Anusha, A. Srinagesh, “Deepfake Video Detection: A Comprehensive Survey of Emerging Techniques,”2023.
URL: https://ieeexplore.ieee.org/document/10894187

[2] Naveed Ur Rehman Ahmed, Afzal Badshah, Hanan Adeel, Ayesha Tajammul, Ali Daud, Tariq Alsahfi, "Visual Deepfake Detection: Review of Techniques, Tools, Limitations, and Future Prospects."2025.
URL: https://ieeexplore.ieee.org/document/10816641

[3] Shital Dongre, Priyasha Agarwal, Nupur Agrawal, Shantanu Adak, Aaryan Mattoo, "Comparative Analysis of DeepFake Detection Models", 2024
URL: https://ieeexplore.ieee.org/document/10881519

[4] Harshpal Singh, Rakesh Kumar, Meenu Gupta, Venkata Suresh Babu Chilluri, "Detecting Digital Deception: A CNN-RNN hybrid Approach of Deepfake Detection." 2025
URL: https://ieeexplore.ieee.org/document/10940830

[5] T. Zhao, J. Yang, Y. Chen, and H. Li, “Self-Consistency Driven Deepfake Detection via Source Feature Analysis,” 2023
URL:https://openaccess.thecvf.com/content/ICCV2021/papers/Zhao_Learning_Self-Consistency_for_Deepfake_Detection_ICCV_2021_paper.pdf

[6] K. Fernandes, L. Silva, and R. Branco, “ Attribution-Based Confidence for Deepfake Video Detection,”2022.

[7] A. Gandhi and S. Jain, “Robust Deepfake Detection Against Adversarial Attacks using Regularization and Prior-based Defenses,”2023.
URL: https://arxiv.org/pdf/2302.11704

[8] Shahad Altamimi, Walid Salameh, "Towards Analysis Detection of Deepfake Video via Deep Learning Models: A Review", 2024
URL: https://ieeexplore.ieee.org/document/10847278

[9] Made Adi Paramartha Putra, Nengah Widya Utami, I Gede Juliana Eka Putra, Nyoman Karna, Ahmad Zainudin, Gabriel Avelino R Sampedro, "Collaborative Decentralized Learning for Detecting Deepfake Videos in Entertainment", 2024
URL: https://ieeexplore.ieee.org/document/10585565

[10] V Gowri Priyaa, M Jaya Harrish, M Udhayakumar, N Jothieswaran, K Dinesh, "Efficientnet-Based Deep Learning Approach for Video Forgery Detection and Authentication", 2024
URL: https://ieeexplore.ieee.org/document/10544113

[11] T M Aruna, G N Divyaraj, C Chethan, T L Nethravathi, "Recognition of Human emotions using Semi-Supervised Technique with SHO based DTM", 2023
URL: https://ieeexplore.ieee.org/document/10397001

[12] B N Jyothi, M. A. Jabbar, "Deep fake Video Detection Using Unsupervised Learning Models: Review", 2023
URL: https://ieeexplore.ieee.org/document/10391497

[13] Kartik Bansal, Shubhi Agarwal, Narayan Vyas, "Deepfake Detection Using CNN and DCGANS to Drop-Out Fake Multimedia Content: A Hybrid Approach", 2023
URL: https://ieeexplore.ieee.org/document/10263628