International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 03 | Mar 2020 www.irjet.net p-ISSN: 2395-0072

Deepfake Video Detection using Image Processing and Hashing Tools

Divya Babu[1], Uppala Santosh Kumar[2], Konduri Ajith Kumar[3], Yennana Jayanth Sai[4]

[1]Assistant Professor, Computer Science and Engineering, GITAM Deemed To Be University, Telangana, India
[2,3,4] Bachelor of Technology Students, Computer Science and Engineering, GITAM Deemed To Be University, Telangana, India

---

Abstract – *Currently, with increased awareness of Artificial Intelligence, Machine Learning and Deep Learning techniques, there has been a huge growth in ways to generate tools and software. It has become an easily achievable thing to generate credible modifications in the media content and provide new content with very little traces of transformation left behind which are coined to be "Deepfake" content.*

This paper puts forward an easy and affordable method of identifying deepfake video content which can alsobe extended over the deepfake images. But currently the paper is going to stick with a brief introduction to the method of creating deepfake videos(understanding basics of how CNNs are used in the background) and then provide our proposed simple solution which majorly uses Django tools and MD5 Hashing Algorithm. The solution is going to be in the form of a project which takes on many sample deepfake videos which were obtained from conglomerate websites. We discuss how effective our method is with the local host applications and how can the process be improvised in the very near future with greatest technological advancements happening at a very rapid rate.

Key Words: **Deep Learning, CNN, RNN, MD5, OpenCV, Neural Networks, Blockchain, Hashing.**

1. INTRODUCTION

Digital Media has been a goal for content stealers to chisel off the work from others' efforts and to get a name for themselves. The same is the motive for many unofficial and unknown cases where the creator's work has been stolen for multiple purposes and the scope for the original product has declined due to the overpowering performance of the duped product. This is not only a problem for the content creators but also for the consumers or the users who are being cheated in the name of brand and might be influenced over the recreated and fake content.

Image and text modification have already developed and are booming all over the world with the introduction of many media editors such as Microsoft Office Word which works as a text editor and Adobe Photoshop CS6 which works as Image and Template editor. These have already taken their major form in the market within the early years of 19th Century. This is being applied slowly but steadily in a more advanced and sophisticated manner with the development in AI and ML techniques over the motion pictures too. Due to rise in usage and knowledge about the various techniques possible to do the manipulation of the content, many technological endeavours have concentrated on the possibilities of digital video modification leaving few or possibly no traces behind which indicate fraudulency in the 20th Century.

The mechanical approaches of being able to toy with the established digital content let it be in the form of images or digital videos has created a fear for the chance of being deceived by the authentic media servicee companies. This has been made possible only because of the thriving opportunities in the Artificial Intelligence (AI) and Machine Learning (ML) which make the development of the art of deceiving much easier and simpler. Autoencoders, Generative Adversarial Networks (GANs) and the Convolutional Neural Networks are the major Machine Learning (ML) and Deep Learning (DL) methods which are used to create deepfake digital video content.

Fig -1: Above is an example of deepfake technology which shows the face of actress Ava Adams(left) being swapped using deepfaking method to become face of another actress Nicolas Cage

Even though the evidences for efforts of creating digitally fake content have existed right from the time of Abraham Lincoln, the present day educational projects are determined towards creating more real videos by improving video and image processing techniques. One such effort was that of the "Synthesizing Obama" program of the 2017 where the goal was to manipulate the video data by changing the lip movements present in the original video with the one which suited a given completely different audio track. This was first time around when the term of "deepfakes" came into existence by use of a Reddit user named after it.

This implied that the users or the aspirants who wanted to achieve perfect deepfake video creation need to have good knowledge mainly regarding the sub-category of Computer

Vision which falls under the main-category of Study of Computer Science. Here is where the libraries for image processing such as OpenCV (Open Source Computer Vision) and imutils start playing the role of image and digital video modification operations.

Uninterrupted developments have led to immense changes in the image and video manipulation fields. Applications such as FaceApp and FakeApp have taken over the market with surprise by making the image or digital video manipulation and processing easier over a large scale in small amounts of time. The smartphone application FaceApp lets user modify an image's attributes very easily and the desktop application FakeApp is helpful for the user to actually develop a deepfake video easily and over considerably shorter time. However the work is still in progress to make a digital content modification which is as good as no change being made.

It is necessary to ensure that a more effective and efficient method is developed in order to identify or detect deepfake content for original content preparers and maintainers in order to have respect for their brand. Deepfaked content might also lead to loss of respect in the society by having made pornographic, acting or social videos with swapped faces and voices of other people who might have never even had such encounters actually in person. The development in this not only indicates advantages such as ready video of any actor for any kind of commercial usage with actor's permission, but also disadvantage of loss of authenticity and credibility if the original creator is not aware of any of the manipulation process. Only if there is an easy detection process, protection is possible as it is very well known that prevention is way better than cure.

Fig -2: Above is another example which shows how modification to the digital video content has changed the face of a singer's to a completely different person's, and that person might use this as a proof of work with original singer unaware of it.

2. UTILIZATION AND CONNECTED WORK

2.1. Pornography

This is one of the most prominently used area of manipulation of digital videos and images which are displayed openly over the active porn websites such as Pornhub where faces or identities of different people are swapped with those of famous and well known Hollywood actors or actresses which might as well tarnish their image in the public point of view even after all and lots of justification and proofs being provided which might even spoil their careers. One such example is of Daisy Ridley who is an English Actress and is popularly known for her work in huge films such as Starwars Trilogy whose deepfake porn video surfaced online and even was published about in many magazines in the recent times.

2.2. Politics

As talked about in the above Introduction, many videos containing deepfaked content of the politicians where they talk over completely irrelevant subjects which they might've never even dealt with appears suddenly over the internet. Famous Leaders such as Mauricio Macri, Adolf Hitler and Donald Trump have been subjected to this too. Even though there's been deepfaking of a video having speech of Barack Obama, the purpose was to bring knowledge to the people about this possibility of deepfaking and making them aware that they need to be aware and careful of such practices by Buzzfeed joined by Jordan Peele.

2.3. Digital Media Forensics

Another majorly concerned areas about the applications of deepfake images and digital videos is that in the field of Forensics. Here the criminals might find ways to escape punishments for the crimes which they have committed by recreating the proofs against them and placing innocent people in the position of wrong-doers.

This not only impacts over the fair decisions being made by the law and enforcement systems' working order, but also affects the trust that is held by the people over the enforcement team who take care of the case proceedings right from the point where it is filed and drafted to the point where the final judgement is given based on all the submitted proofs. And as deepfake videos are nearest possible material with very few remnants of manipulation, them submitted as proofs can affect the judgement in an unfair manner.

2.4. Object based Video Manipulation Techniques

These methods have come into picture right after the advancement in techniques related to image processing after the 1990s where the auto encoder methods which fall under the category of neural networks and even the most popularly used technique for deepfaking digital video content which is the Generative Adversial Networks (GANs) are being used.

Using these methods users are able to generate deepfaked digital video content by modifying face or any other object's attributes and redesigning them as per the requirements of the user to get desired output video in order to publish or produce it as a proof of work.

2.5. Malfunctioning of the Face recognition software

Sometimes, even trusted software for identifying the face information and then providing other data related to the user which act as a procedure for protecting authenticity and credibility, might fail to detect the modified face information as it is not trained so well that it can suffice the information provided to train a GAN or an autoencoder which generates perfect or near perfect videos containing the faces of original owners of the work or a product hence making it very much needed to have a trusted way of identifying deepfake content and differentiating it from the original face in order to make privacy meaningful.

3. GENERATING DEEPFAKE VIDEOS

The process of deepfake video generation conventionally depends upon the variants of neural networks known as autoencoders and Generative Adversial Networks (GANs). An auto encoder is composed of mainly an encoder which converts an image into a low aspect dormant space, and a decoder which rebuilds the image from the dormant form of it.

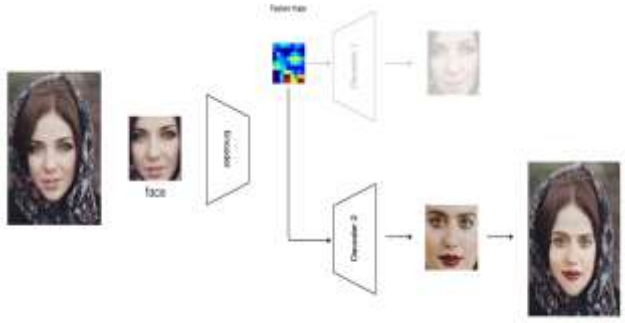

Fig -3: The above image depicts how an image is converted into it's deepfake using autoencoders from a lower dimensional space.

Deepfakes make use of this kind of planning by having a generic encoder which encrypts the user's image into a dormant space. This generic encoder contains important information regarding certain aspects of the image such as the size ratio, object structure and position. While decoding, the user might manipulate this predefined information of the image with his/her desired information to get a wanted image as a result. This depicts that the object's detailed information will be camouflaged over the underlying structure and object features in the each original image divisions of the given video to get hands over the deepfaked video content.

Another most used alternative for this kind of approach is the usage of GANs for the generation of deepfake videos. The GANs instruct a generator which is a kind of decoder and a distinguisher which identifies the key link. The generator builds a dormant description of the original image while the distinguisher tries to figure out if the complete target image of a video segment is generated or not. This makes it possible for the generator to build an image which is almost impossible for the distinguisher to determine if it is a depiction of the original video segment's image.

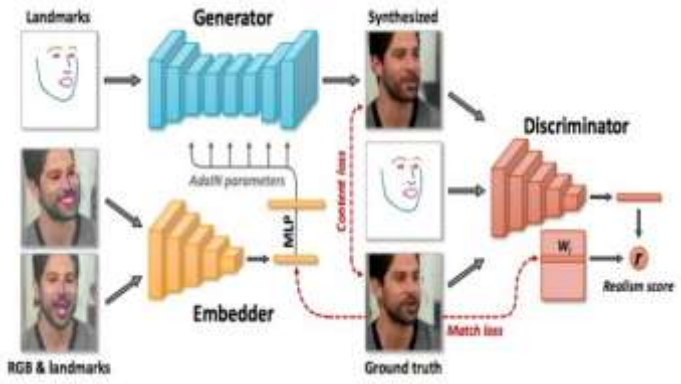

Fig -4: The above image depicts how an image is converted into it's deepfake using the generators and discriminators in the GANs.

Although this means that both the proposed algorithms work toward zero similarity and complete efficiency, the autoencoder are able to generate better compacting performance than that observed with the GANs. This encourages us to go with the detailed study of how the auto encoders work over the deepfake video generation which is not so different from the GAN method of it in many aspects.

The deepfake video generation using the auto encoders is divided into 2 main categories which are of training and video generation finally.

3.1. Training

Deepfake videos are prepared with an aim of manipulating the original content of a video into it's own digital duplicate. Hence it is only obvious that before aiming towards creation of a deepfake video, we need to first have information or the data that should be visualized to which form it has to be modified into.

Therefore during the training process, the system that acts as a deepfake video convertor powered with autoencoders uses two primary sets of images. The first set would be made up of samples from the original video and the second set of images will be that of the samples to which the video's data has to be modified into. To make the training process more fruitful these two sets are needed and both the original image and the desired image quality must be under same conditions such as brightness and contrast. As difference in the external conditions and the overall features of the image might disturb the primary goal of deepfaking which is "to manipulate without leaving any proofs behind".

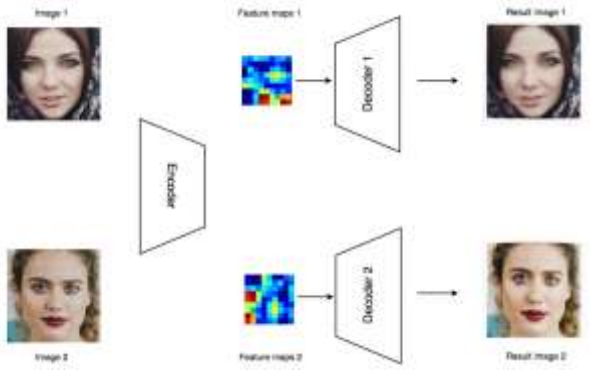

Fig -4: Network of encoders working along with multiple decoders in an autoencoder.

We are going to have a network of encoders being wrapped into a single autoencoder and also a collection of multiple decoders embedded within it. This makes the video processing, information extraction and modification by the encoder network at a different level of rigorousness. And the usage of multiple decoders provide each dormant space of an image being extracted into different possibility by different decoder giving autoencoder the choice of selecting best replacement for the original content as a deepfake from a whole set of different images.

3.2. Video Generation

For final deepfake video generation, autoencoders go with the deep learning technique of CNN. As we know that deep learning is a software mechanism that allows the computer system to imitate the nervous system of humans by recreating an artificial network of neuron functions just like the network of neurons in our brain. This member of Machine Learning (ML) is coined the term Deep Learning (DL) as it makes appropriate use deep neural network mechanisms.

All the neural network related DL algorithms are created via interlinked layers which are input layer (first layer), output layer (last layer) and multiple hidden layers. These layers facilitate the automated learning concept of a Deep Learning software powered system without having previous information externally entered by the programmers.

The Convnets or Convolutional Neural Networks (CNNs) are the type of multilayered networks we are going to be using here. The CNNs' structure is designed in such a way that it precisely recognizes an object's dimensions and features from a picture or a video. Hence CNNs are mostly used over the unstructured data such as images or digital videos.

A CNN mainly completes generation process in 2 phases where the first one implies the creation of a non continuous analytical model as output and the second phase is driven towards modifying the developed model using CNN manipulation techniques.

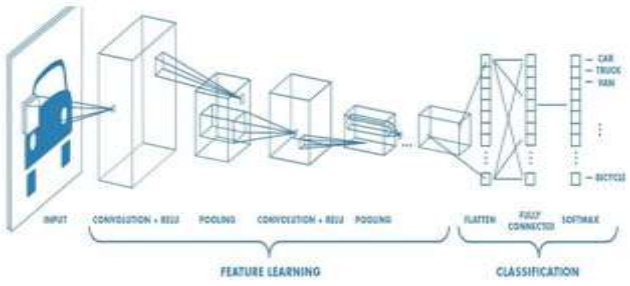

Fig -5: A CNN's structure hence can be depicted as in the above figure.

The four major components of CNNs play an important role in the video generation process from each segment's image modification.

The first component of **convolution** derives information of the object in an image sequence and gets to know about various patterns specifically. The second component of **non-linearity** checks the deep concepts derived from convolution about the features of the images such as sharpness, edge modification and border identity. Then the third component of **pooling** (also known as **subsampling**) provides a stage for the user to manipulate with the data presented originally in an image at a very deeper level with it's features. The final component of **classification** then checks for the relation between information and if even after this level of processing, the image's classification is very different from the original image data features, the image is yet again going to be processed using another set of layers of CNN.

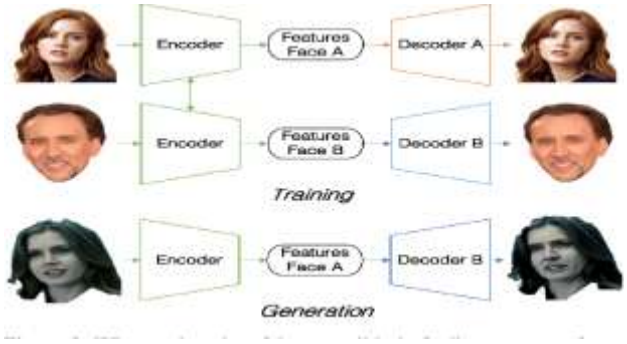

Fig -6: Here is how swapping of faces is done just by solving through all the possible face swaps being encoded as having same image information or features.

4. DEEPFAKE VIDEO IDENTIFICATION

After studying about what are deepfake videos and going through the process of creating them, here comes the challenge of detecting them. Many technologies and techniques have come into picture after the deepfakes came into existence.

Just as the Deep Learning method of CNN is being used in the generation of deepfake videos, another Deep Learning

based technique of Recurrent Neural Networks (RNNs) can be used for identification or detection of deepfake videos. The Deep Learning software and it's methods mainly use TensorFlow in order to put those concepts in practical usage through coding in python with it. But we are not going to use it here right now in our work.

Many programs and applications have already come into motion using the RNN methods even though they are still undergoing rapid growth and development as lots of research is still under progressive conditions. The RNNs have facilitated the booming technology in present market which is, "AI in HR", i.e., Artificial Intelligence in Human Resource management.

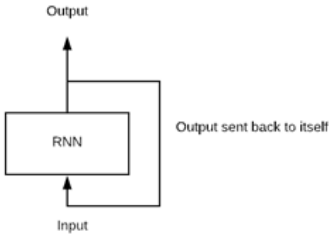

Fig -7: This shows how RNNs work with the concept of automation and improvisation.

4.1. Existing Systems

There has been an observable development in the field of deepfake video creation and detection in the commercial arena with the rise of software applications such as the FakeApp and the FaceSwap.

FaceApp is one such powerful face transformation application developed for usage in Android or IOS smart phones which is powered by Artificial Intelligence (AI) techniques of Neural Networks and Genetic Algorithms. It helps users to click pictures of them having certain advanced filters that act as hidden layers in the neural network with input layer having the original image and the output layer generating an edited image or photo.

FakeApp is a desktop application program that allows us hiding certain features and modifying them in an image by means of some AI training method which later, can be overlapped with photos of the faces in videos thus producing a deepfake digital video content after classification of video segments into individual images which have a watermark for it's detection and recognition.

Certain web applications such as FaceSwapOnline also allow us to give our videos or images as inputs with AI and DL powered systems running in the background that allow us to modify the image content according to our wish through changing or applying filters and even changing resolution or dimensions of the images. Such web based applications use very powerful servers at datacenters all around the world, in order to manage the hit traffic from users and also perform the image or video manipulation in the backend.

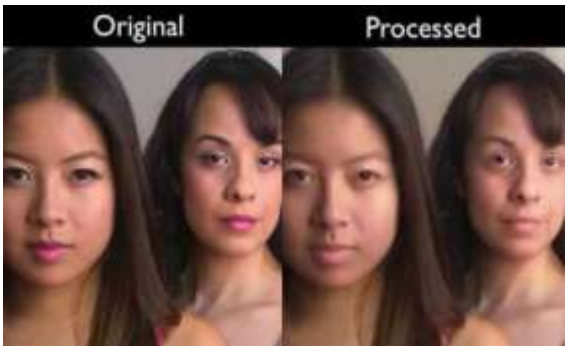

Fig -8: Depicts how different a processed image or video might get after using filters in the FaceApp desktop application.

4.1.1. Disadvantages

All of the above methodologies might require high-end specifications of systems that some systems might only be able to work with.

They also might require fast internet connection to work.

4.2. Proposed Solution

The solution or method we have put forward, focuses on the usefulness of hashing and image processing facilities for detection of deepfake video content. The work is majorly split into five modules which shows how the videos are accepted at a web page, traverse through the localhost, processing of the videos will happen at the backend and result is displayed over the webpage.

4.2.1. Advantages

The static web application we have developed works well even with decent specifications of a system (such as in a system with 4GB RAM and 32-bit Operating System).

It even runs without internet as it is designed to work on the localhost.

4.2.2. Module-wise explanation

Module I: Here, two vulnerable videos are selected by the user and are uploaded over the static web application which is created using the Django python web framework which makes development of rapid, powerful and smoothly running webpage design and development possible.

Usage of Django framework which acts as an integration of python implementation modules and also front end technologies such as HTML (Hypertect Markup Language, which is used for webpage creation), CSS (Cascaded Stylesheets, which is used for webpage styling), etc. that facilitate the creation of static webpages. Here, videos of MP4, Webm or OGV formats are acceptable.

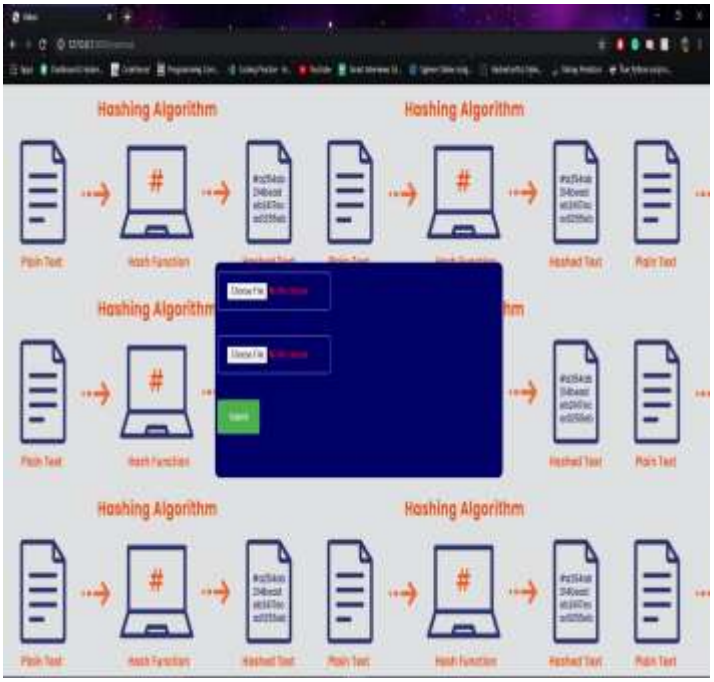

Fig -9: Static webpage created using Django

Module II: The videos taken as input traverse through the localhost server and reach the internal storage of the system, here is the connection between Django files (for obtained videos) and python script files is made possible through the argparse module.

The argparse module makes it easy for the user to work with command-line interfaces and also help in parsing and mapping of files which are to be interfaced with each other.

Module III: The videos accepted are divided into frame-wise images , where 19 frames' images form a single second of the video. Here is where all the images go through rapid amount of processing with certain filters and operations being done using the OpenCV (which stands for open source computer vision) and imutils packages. The OpenCV (which also has its implementation extended over C++ and Java platforms and not only over the Python platform) provides many important image processing utility functions with a cross-platform working capability. And the imutils package provides a combination of comfortable functions that enable basic image processing actions such as rendition, transcription, rescaling and delineation in much easier manner.

```
# Function to extract frames
def FrameCapture(path):
    # Path to video file
    vidObj = cv2.VideoCapture(path)

    # Used as counter variable
    count = 0

    # checks whether frames were extracted
    success = 1
    x=[]
    while success:
        # vidObj object calls read
        # function extract frames
        success, image = vidObj.read()

        # Saves the frames with frame-count
        x.append(image)

        count += 1
    return Dhash(x)
```

Fig -10: The above python code snippet shows how the videos are obtained and divided into frames.

Module IV: The images which correspond to each frame of a second in the videos are assigned with hash values using the MD5 Hashing Algorithm.

The cryptographic hash function of MD5 (Message-Digest Algorithm 5) is one that produces a hash value as an output which is as small as 128 bits. Even though we have algorithms like SHA1 (Secured Hash Algorithm 1) which is another such easier cryptographic hash function to generate a hash value for a corresponding data key, we use MD5 as we get much efficient and practical performance due to it's faster processing than the rest and produce compact hash values as a result of size restricted to 128 bits.

```
#Hashing and creation of hash tables
imghash = []
# print("Test")
def Dhash(imgs,hashSize=8):
    for p in imgs:
        if p is None:
            continue
        p=cv2.cvtColor(p, cv2.COLOR_BGR2GRAY)
        resized = cv2.resize(p, (hashSize + 1, hashSize))
        diff = resized[:, 1:] > resized[:, :-1]
        imghash.append(sum([2 ** i for (i, v) in enumerate(diff.flatten()) if v]))
    for i in imghash:
        print(i)
    return imghash
```

Fig -11: The above python code snippet shows how the videos' frame images act as keys and are assigned hash values using Dhash() function.

These hash values generated respectively for each of the videos are tabulated which gives us 2 videos' hash tables of each frame.

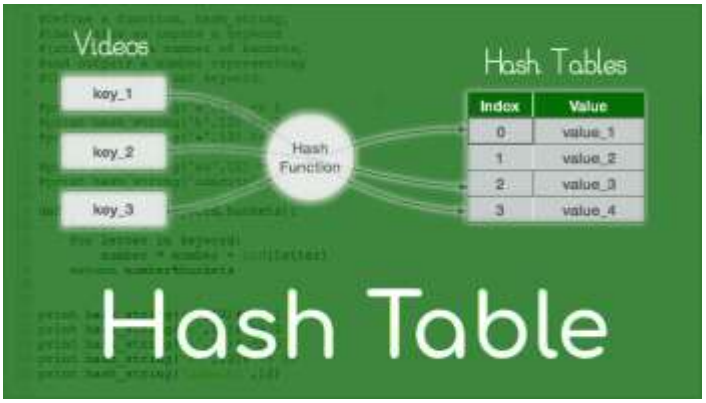

Fig -12: A depiction of how hashing takes places with videos as keys and 128 bit hash values as output with the hash function being MD5.

Module V: Here, the hash tables generated in the previous modules are compared with each other. If hash values differ right at the starting value, the videos are completely different and hence the localhost returns a message as "Not Same Videos".

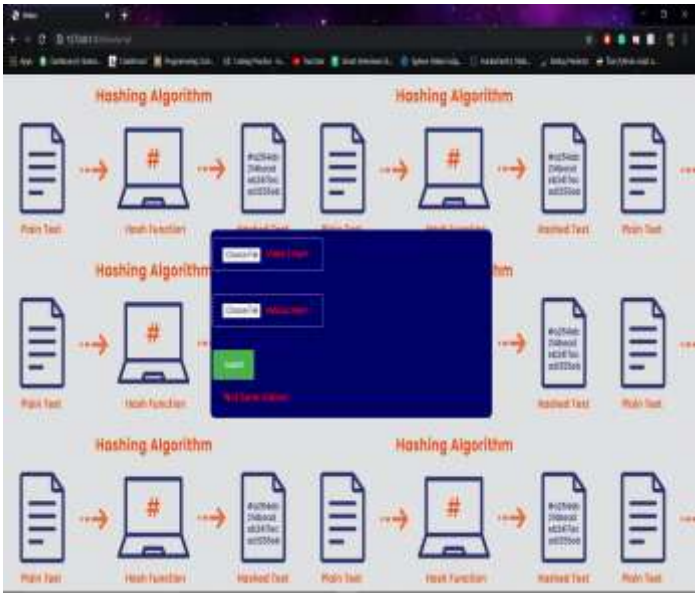

Fig -12: This shows the message of "Not same videos", when the videos are entirely different.

But, if the hash values in the two tables remain the same till a certain point, system stops comparison immediately and returns "Deepfake Videos" to the user interacting with the system through the webpage, as the frames are same till a particular point in the video.

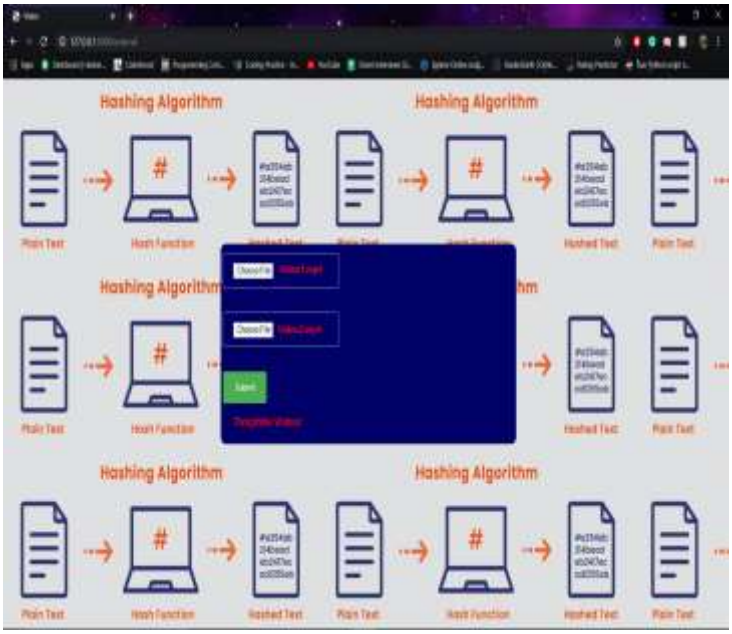

Fig -13: This shows the message of "Deepfake Videos", when the videos have similar hash values till a certain point of comparison.

Similarly, if the videos are entirely having same hash values for the frames all over, then the system declares the user-submitted videos to be original or "Same Videos".

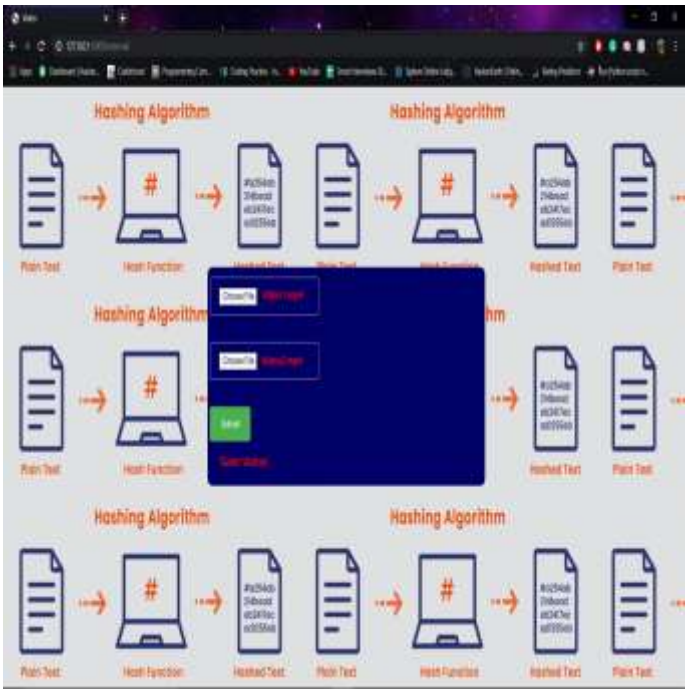

Fig -14: This shows the message of "Same Videos", when the videos have similar hash values in both hash tables.

5. CONCLUSIONS

In this paper, we have proposed a system which works over making deepfake video detection easier and much more simple task by using the image processing and hashing techniques unlike using the RNNs for this purpose.

Utility of Deep Learning Techniques such as RNNs for the detection of deepfake videos might be more in use than the proposed solution but the solution we proposed has higher efficiency over narrow areas of applications such as in a small business enterprise or just for an individual's usage and concentrates on working with lesser time having access to limited resources (such as not having internet access or RAM being 8GB or lesser).

In RNNs the entire image frames generated for each second of a video are compared to one another and tested according to the training data provided. But in the case of our proposed solution we need not compare the entire images' data of each frame in the video, instead discover if a single hash is displaced which generates an error in the visual chain of the video and the video is concluded to be Deepfake.

6. FUTURE SCOPE FOR ENHANCEMENT

We will be making use of the advanced technological concept of the "Blockchain" in the detection of deepfake videos.

Blockchain even though an emerging trend currently in the world of technology, is not new in application as it is just an integration of three main existing technological applications namely Peer-To-Peer Networks, Private Key Encryption and Software Programming.

In a blockchain, the nodes in the network are interlinked having control over a decentralized database known as a "Ledger" which means that all the details of every transaction made by each of the node owner or user is made available to every other user who is a part of the chain.

Each individual block over a blockchain network is considered as a user and the block consists of user data or the transaction details, a unique hash for the current block, hash of the previous block and some program which enables the transaction of that user.

Even a slightest modification in the user data of a block changes the hash entirely and the whole blockchain gets disrupted.

This could be seen as an applicative enhancement for our project where we developed a hashing function for each video frame image (as a key). This hash, video frame data, the previous block hash and an action whenever the hash gets changed, forms a Blockchain for deepfake video detection purpose.

Etherium is one such most widely known and used Blockchain network whose range is all over the world where the software programs embedded into each block are known as Smart Contracts. This could be used in much more practical and simple manner in nearest future.

REFERENCES

[1] Haya R. Hasan and Khaled Salah, "Combating Deepfake Videos Using Blockchain and Smart Contracts", IEEE Journal published in March 2019.

[2] David Güera and Edward J. Delp, "Deepfake Video Detection Using Recurrent Neural Networks," IEEE International Conference Paper published in 2018.

[3] https://en.wikipedia.org/wiki/Deepfake.

[4] https://www.guru99.com/deep-learning-tutorial.html.

[5] https://www.guru99.com/convnet-tensorflow-image-classification.html.

[6] https://www.malavida.com/en/soft/fakeapp/#gref.

[7] https://faceswaponline.com/.

[8] https://www.faceapp.com/.

[9] https://www.djangoproject.com/.

[10] https://docs.python.org/3/library/argparse.html.

[11] https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_setup/py_intro/py_intro.html.

[12] https://github.com/jrosebr1/imutils.

AUTHORS

Mrs. Divya Babu
Assistant Professor
Department of CSE,
GITAM Hyderabad

Uppala Santosh Kumar
2210316556
Department of CSE,
GITAM Hyderabad.

Konduri Ajith Kumar
2210316532
Department of CSE,
GITAM Hyderabad

Yennana Jayanth Sai
2210316564
Department of CSE,
GITAM Hyderabad