A Machine Learning Approach for DeepFake Detection

Gustavo Cunha Lacerda and Raimundo Claudio da Silva Vasconcelos
Instituto Federal de Educação, Ciência e Tecnologia de Brasília - IFB
Taguatinga-DF, Brasil
Email: gustavocunhalacerda@gmail.com, raimundo.vasconcelos@ifb.edu.br

***Abstract*—With the spread of DeepFake techniques, this technology has become quite accessible and good enough that there is concern about its malicious use. Faced with this problem, detecting forged faces is of utmost importance to ensure security and avoid socio-political problems, both on a global and private scale. This paper presents a solution for the detection of DeepFakes using convolution neural networks and a dataset developed for this purpose - Celeb-DF. The results show that, with an overall accuracy of 95% in the classification of these images, the proposed model is close to what exists in the state of the art with the possibility of adjustment for better results in the manipulation techniques that arise in the future.**

I. Introduction

Disinformation and its sharing are considered a worldwide concern entitled as fake news, that can be understood as the dissemination of content that is not true and that was purposely generated in order to convince readers about the veracity of information [1]. Digitally manipulated images or videos to spread fake news have emerged and this technique is known as DeepFake, which is based on machine learning that offers a wide variety of methods for face exchange and manipulation, including the use of computer vision, deep learning and word ”fake”.

This manipulation technology has been widely used in the cinema, as in the case of the film ”Fast and Furious 7”, in which the face of an actor, who died, was replaced in the body of his brother for the recording of the last scene of the film [2]. In addition, this type of generation has already been used in the creation of a controversy speeches in a documentary about a cook who also passed away [3]. DeepFakes have also become very popular in the internet, going viral with memes, videos of famous politicians and artists. Despite its harmless use in entertainment product cases, with the evolution of imaging technologies and processing power, DeepFake has been used by people for malicious purposes. As home computers have increased their processing power to the point where most of the images generated could be made in a amateur way, this technology has become mainstream.

To avoid the harm caused by the misuse of this technology to privacy and veracity of information, some of the major technology companies have started initiatives to combat DeepFakes. An example of this is the DeepFake Detection Challenge (DFDC) [4] initiative, which is a program developed by Facebook to promote solutions in detecting and classifying possibly manipulated images. In addition to these bigtech solutions, there has been an increase in research related to the classification of face manipulation in images. According to a projection made with data from Dimensions [5] by the end of 2020, about 737 DeepFake related papers were expected. However, according to the same site, using the same research method, there were 1,333 DeepFake related papers by the end of the year, an increase of about 80% compared to the projection.

Current DeepFake detectors show good classification results, even though, at the same time, face manipulation techniques have also received constant updates and improvements. Looking at this, the present work suggests an updated approach, starting with Celeb-DF [6] dataset, recently created for the purpose of clustering DeepFake images. In addition to use convolutional neural network models which, if well configured and fed with enough data, can detect manipulated images and, with finetuning, can classify new generations of face manipulation software.

The goal of this paper is to develop a computer vision algorithm for deepfake detection with the help of deep learning. This work is organized as follows: section II presents related articles, section III presents materials and the proposed method is shown in section IV. Results and discussions are in section V and conclusions are in section VI.

II. Related Works

With increasing discussions about the misuse of DeepFakes, academic researchers have improved research on detecting face manipulation. In [7] Güerra and Delp suggests using ImageNetV3 [8] pre-trained with ImageNet [9] dataset for feature extraction and using a Long Short-Term Memory (LSTM) for analysis and classification of results. The authors analyse the formation of deepfakes, in which encoders are applied to resize, extracting features of a face, to exchange these features with another face. In the process of exchanging these features, the target face and the original are not always in the same light conditions and even file format, which makes it difficult for the DeepFake creation algorithms to generate a realistic image. These errors in the creation can be targeted by algorithms that aim to decrypt them. This technique yielded an overall accuracy of 97.1% on a 80-frame video fragment, demonstrating the high accuracy and effectiveness of this technique.

Another important paper related to DeepFake detection is the researh of Lima *et al* [10], that suggest an approach to detect manipulations on artifacts present in AI-generated DeepFakes videos in the frames transitions. The researchers used Celeb-DF and some pre-trained convolutional networks with Kinetics dataset to make the model for predictions. After train, the model achieved an average of 98.26% in videos.

In [11] it is also suggested the detection of DeepFakes through analysis of convolutional traces generated in the creation of this type of image. This new technique consists of a method of analyzing the relationship of each pixel and its neighbors, finding the relationship of these neighborhoods using expectation maximization. Then, after analyzing the relationships, it is possible to do a classification using KNN, SVM and LDA to define whether the image is a DeepFake. In this research there was a comprehensive analysis of different kernels and datasets combined which resulted in a maximum accuracy of 99.31% using a linear SVM.

III. Materials

A. *Hardware and Software*

For this work we used a computer with Intel(R) Xeon(R) CPU E3-1270 v6 @ 3.80GHz coupled with NVidia Titan V video card 16 Gb of VRAM and 66 Gb of RAM DDR4@2400Mhz was used. The computer uses Ubuntu 20.04.3 LTS operating system and the main tool for building the convolutional network model was Pytorch library.

B. *Celeb-DF v2 and MediaPipe*

The dataset chosen was Celeb-DF [6] in its second version. This dataset contains 590 videos without DeepFake, and 5,639 videos with DeepFake. The videos are about 13 seconds long at 30 frames per second, totaling over two million frames of data for use in DeepFake classification problems. In Figure 1, it is possible to see a batch with examples of the two classes that is is the dataset.

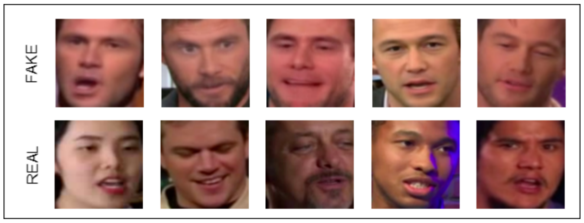

Fig. 1. CelebDF v2 dataset sample divided by class.

In this image set the people's faces underwent several deepfaking techniques that resulted in manipulated videos to compose the class of fakes present in the set. It is noticeable that during the process certain regions end up being affected in the fusion between target face and original, forming image artifacts. The most critic regions of these errors are mouth, nose and eyes, which is exemplified in the Figure 2.

Another tool that was be used in conjunction with the dataset is the computer vision toolkit developed by Google programmers and researchers - MediaPipe [12]. It is a set of computer vision tools created with a focus on efficiency and portability in which even embedded systems can run applications that consume the library. In this work, the set of tools related to point detection and face meshing in images was used.

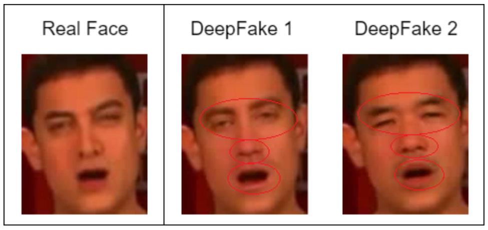

Fig. 2. Example of face critic regions and artifacts.

IV. Proposed Method

The method proposed here is composed of three main steps: pre-processing, training and validation, as we can see in Figure 3. After choosing the dataset, videos are processed to extract frames and faces that will be used as input to the network for training step. These data are then normalized and their sizes adjusted for homogenization purposes. Training is the kernel of the research, in which a pre-trained convolutional neural network model was chosen to go through a fine-tuning phase to adjust network parameters and characteristics for the DeepFakes binary classification problem.

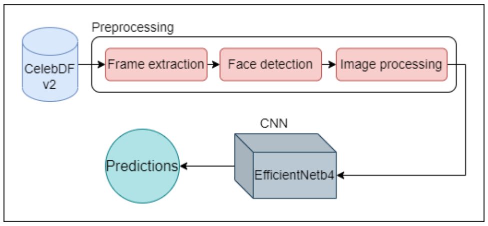

Fig. 3. Proposed method

A. *Data processing*

To carry out the training with the Celeb-DF dataset, it was first necessary to balance the amount of videos of the two classes. The Celeb-DF in its second version had 590 real videos and 5639 fake videos, so the limit of videos for each class was set to 590, resulting in 1780 videos regarding the binary classification of the problem. Then the videos were separated into a test subset to ensure that in training subset there is no frame of the same video that is also in the test subset. This step ensures that the tests are blind and that the network has never trained with the data used for this purpose. Table I shows this separation.

After this separation, up to 500 frames of each video were extracted, counting only frames in which the MediaPipe library was able to find a human face with an 80% certainty rate. In addition, the facial points found by google's library were used to make a cut on the boundaries of the face,

TABLE I
SUBSET VIDEOS DIVISION

Fake Train Subset	472
Real Train Subset	472
Fake Test Subset	118
Real Test Subset	118

decreasing the information and focusing on what the network must learn to perform the classification. These faces went through normalization and resizing by 224x224 pixels before being saved in a dataset that will be used by the EfficientNet network. The details of the processing flow is showed in Figure 4.

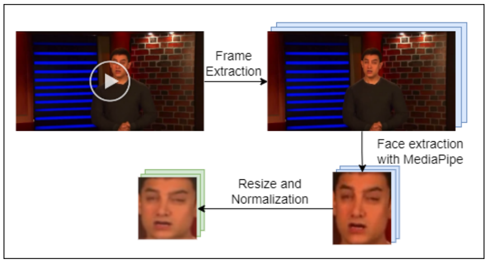

Fig. 4. Video processing steps

B. EfficientNet

A Convolutional Neural Network (CNN) is a Machine Deep Learning algorithm that can capture an input image, assign weight and bias to various characteristics of an image, differentiate objects, and perform less expensive analysis on image sets [13]. The prepossessing required on a CNN is much lower compared to other ranking algorithms. While in primitive methods filters are handmade, with sufficient training, CNNs have the ability to learn these filters.

With the advancement and dissemination of the power of CNNs, architectures emerged that sought to extract the most from this concept, such as ResNet or Xception. In this research, the EfficientNet [14] architecture was chosen, a convolutional neural network model that is very efficient in relation to the amount of resources and interactions for convergence.

While EfficientNet work well in ImageNet [9], it should also be transferable to other datasets to be as useful as possible. EfficientNet was tested on eight widely used transfer learning datasets. EfficientNet models achieve better accuracy with 4.7x average (up to 21x) parameter reduction in 5 of the 8 datasets with transfer learning compared to the state-of-art solutions. Such as CIFAR-100 [15] (91.7%) and Flowers [16] (98.8%) results of accuracies suggest that the architecture is highly recommended for problems that can be solved by transfer learning or fine tuning, which is the case for DeepFake classification.

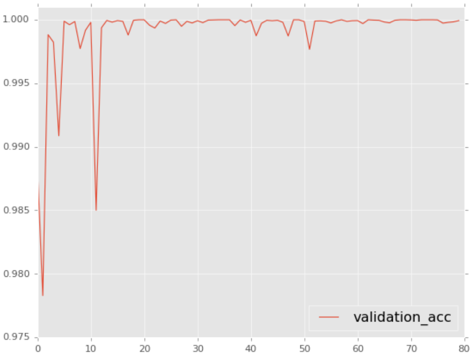

Fig. 5. Validation accuracy over the epochs.

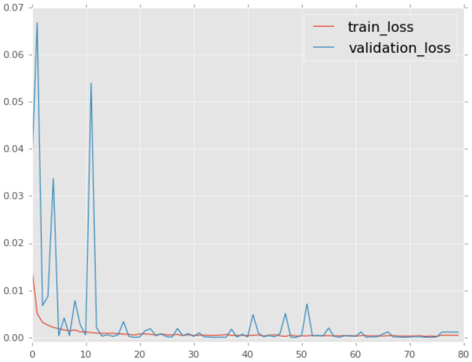

Fig. 6. Validation and train losses over the epochs.

C. Train and validation

This work used a pre-trained version of EfficientNet-B4, an architecture of the EfficientNet family, to train the model. This version was chosen because it presents the best cost-benefit ratio with the hardware available for the present research. A learning rate of 0.0001 and Adam optimizer with 0.005 weight decay used to avoid overfitting. The dataset for training was divided into the 80/20 training and validation ratio, respectively. The model was trained with dataset in batches of 32 faces in 80 epochs, it also performs a validation step in each epoch to verify the quality of predictions and the convergence of the classification.

V. RESULTS AND DISCUSSION

An EfficientNet-B4 architecture and Celeb-DF V2 dataset presented very satisfactory results when used together. During training and validation, accuracy progression as shown in Figure 6, remained above 97% at all times with increasing trends. The validation and training losses remained in a decreasing trend over the iterations, indicating the convergence of the model throughout the training stage.

Some metrics were used for the analysis of the proposed model, among them F1 Score, Accuracy, Precision and Recall with results presented in Table II.

Another important metric that was used to qualify the method generated by this paper is the confusion matrix.

TABLE II
FINAL RESULTS OF TESTS WITH THE MODEL

Final Accuracy	0.9552
Recall	0.9161
Precision	0.9999
F1 Score	0.9562

With this metric is possible to verify the performance of the algorithm by comparing the predictions with the real values of the labels. In our binary classification model there are 4 results that can be seen in the Figure 7: true positive (TP) true negative (TN), false positive (FP) and false negative (FN) where true or false is understood as whether an image is a DeepFake or not. In it, it can be seen that the model tends to be more rigorous in correctly classifying real images, which is good for this type of classification considering its applicability. There is more sensitivity in classifying an image with any suspicion of manipulation, in order to ensure safety rather than certainty that the image is true fake. This behavior also explains why the precision metric resulted in a value close to 1, since there is a only one real prediction in fake image.

Although the final accuracy was below the state-of-the-art of 99.73% seen in [7] or 97.1% seen in [11], the results are as good as there is the possibility of improvement and the network can still be fed with new data in the current state for finetuning and can adapt to new models of DeepFake creation. Besides, the Celeb-DF v2 dataset proved to be quite sophisticated compared to others used in other research on digitally manipulated face classification such as the DFDC [4] datasets.

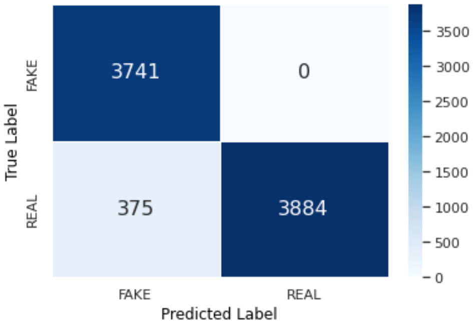

Fig. 7. Confusion matrix in the model applied on the images test subset.

VI. CONCLUSIONS

DeepFake detection is something quite complex, since techniques for creating new manipulated images arrives faster than good solutions for detection and protection against this type of spoofing. Thus, the technique presented by this article presented very satisfactory results for the current state of the problem. The EfficientNet network in conjunction with the Celeb-DF dataset were assertive and combined for the generation of a robust model that achieved satisfactory predictions, maintaining an accuracy of more than 93% in images of 224x224 pixels of height and width, achieving Recall of 0.9161 and F1 of 0.9562.

With the results found it is possible to find a way to improve the model. For future work it is necessary that the model presents tests on images from other sources and more varied manipulation techniques, increasing the generalization of the model for predictions. Moreover, increasing the number of iterations seems to be beneficial for the model, since the results showed a tendency to increase for accuracy and decrease for losses without reaching an overfitting that would lead to more divergent results in the tests. Another factor that can be decisive for improving classification performance is, with greater features, choosing a more complex CNN model or one of the other models in the EfficientNet family that has more parameters and a larger kernel.

REFERENCES

[1] Q. Xul, X. Zou, and J. Zhao, "On-line detection of defects on fruit by machinevision systems based on three-color-cameras systems," in *International Conference on Computer and Computing Technologies in Agriculture*. Springer, 2008, pp. 2231–2238.

[2] H. R. Carolyn Giarda. (2022) How 'furious 7' brought the late paul walker back to life. [Online]. Available: https://www.hollywoodreporter.com/movies/movie-news/how-furious-7-brought-late-845763/.

[3] T. N. T. Helen Rosner. (2022) The ethics of a deepfake anthony bourdain voice. [Online]. Available: https://www.hollywoodreporter.com/movies/movie-news/how-furious-7-brought-late-845763/.

[4] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The deepfake detection challenge (dfdc) dataset," *arXiv preprint arXiv:2006.07397*, 2020.

[5] D. . T. most advanced scientific research database. (2022) MS Windows NT kernel description. [Online]. Available: https://www.dimensions.ai/

[6] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-df (v2): a new dataset for deepfake forensics," *arXiv preprint arXiv:1909.12962*, 2019.

[7] D. Güera and E. J. Delp, "Deepfake video detection using recurrent neural networks," in *2018 15th IEEE international conference on advanced video and signal based surveillance (AVSS)*. IEEE, 2018, pp. 1–6.

[8] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[10] O. de Lima, S. Franklin, S. Basu, B. Karwoski, and A. George, "Deepfake detection using spatiotemporal convolutional networks," *arXiv preprint arXiv:2006.14749*, 2020.

[11] L. Guarnera, O. Giudice, and S. Battiato, "Deepfake detection by analyzing convolutional traces," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 666–667.

[12] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee *et al.*, "Mediapipe: A framework for building perception pipelines," *arXiv preprint arXiv:1906.08172*, 2019.

[13] D. S. A. Team. (2022) Capítulo 40 – introdução as redes neurais convolucionais. [Online]. Available: https://www.deeplearningbook.com.br/introducao-as-redes-neurais-convolucionais/#:∼:text=Uma\%20Rede\%20Neural\%20Convolucional\%20(ConvNet,de\%20diferenciar\%20um\%20do\%20outro.

[14] B. Koonce, "Efficientnet," in *Convolutional neural networks with swift for tensorflow*. Springer, 2021, pp. 109–123.

[15] A. Balaji, Y. Wu, and J. Yoon, "Cifar100 convolutional model based classification benchmark."

[16] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*. IEEE, 2008, pp. 722–729.