DeepFake Video Production and SIFT-based Analysis

Miljan Đorđević, Milan Milivojević, and Ana Gavrovska, *Member, IEEE*

***Abstract* — Nowadays advantages in face-based modification using DeepFake algorithms made it possible to replace a face of one person with a face of another person. Thus, it is possible to make not only copy-move modifications, but to implement artificial intelligence and deep learning for replacing face movements from one person to another. Still images can be converted into video sequences. Consequently, the contemporaries, historical figures or even animated characters can be lively presented. Deepfakes are becoming more and more successful and it is difficult to detect them in some cases. In this paper we explain the video sequences we produced (e.g. using X2Face method, and First Order Motion Model for Image Animation) and perform deepfake video analysis using SIFT (Scale Invariant Feature Transform) based approach. The experiments show the simplicity in video forgery production, as well as the possible role of SIFT keypoints detection in differentiation between the deeply forged and original video content.**

***Keywords* — SIFT, video production, forgery, DeepFake, deep learning, computer vision.**

I. Introduction

FORGED video and audio content has existed for a long time, but recent advancements in the field of deep learning have opened forgeries to almost anyone. State of the art tools are available publicly and sit at anyone's disposal. Advancements in the field of Computer Vision have introduced the concept of DeepFakes [1], particularly convincing forged videos with the great advantage of being extremely easy to make. After being introduced to the world by amateurs, DeepFake generation algorithms have started to interest professionals who have brought DeepFake models to completely new grounds. DeepFake algorithms are most commonly convolutional neural networks which learn facial feature mapping from one face to another. Mappings are usually tensors containing the coefficients which describe human head (and more recently full body or even animal body) movements in the given training video. Postprocessing can make the resulting videos almost indistinguishable from real videos.

There have been multiple cases in which a DeepFake video posted on some of the leading social media platforms has caused the audience to believe what they are seeing is real. To make things worse, those examples were mostly videos of politicians, sometimes during their presidential campaigns. DeepFakes are getting harder to spot because of the availability of bigger datasets as well as the ever advancing postprocessing techniques which help with the problem algorithms have when matching two faces of different shape. Most web platforms have banned DeepFakes, but some like Facebook are allowing this type of content, in the case of Facebook until its third-party fact-checkers report the piece of media as forged. As the DeepFake generation algorithms become more advanced, the need for counter measures becomes more and more obvious. Detection algorithms mostly search for mistakes in DeepFakes like double eyebrows, lack of blinking, etc. Analysis of specific head movements like head tilts and chin motions has proven useful in forgery detection as well.

What makes DeepFakes unconvincing to some people are the factors which currently limit them. It seems that current algorithms primarily create images and videos of limited resolutions. Also, it seems that further face warping is needed to match another face. Unwanted artifacts are often produced. There are two general approaches in forgery detection: analysis with and without a reference. Since original image or video is rarely available, it is important to have a non-reference approach.

In this paper we perform deepfake production and analysis results. Also, we demonstrate possibilities for developing a technique for differentiating between original videos and DeepFakes. Experiment consists of comparing successive frames in the original and doctored videos using SIFT (Scale invariant feature transform) algorithm [2]. Namely, only original on one hand, and only doctored frames are compared in a successive manner having in mind the need for non-reference approach.

The paper is organized as follows. After introduction, in Section II we give a brief explanation of visual forgery production and the recent history of the DeepFake term. In Section III we give explanations of SIFT based keypoints usage. Details about the simulation, both production and

Paper received May 20, 2020; revised June 19, 2020; accepted June 24, 2020. Date of publication July 31, 2020. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Irini Reljin.

This paper is revised and expanded version of the paper presented at the 27th Telecommunications Forum TELFOR 2019 [16].

The research presented in this paper is partially funded by Ministry of Education, Science and Technological Development of the Republic of Serbia (No. 32048, 44009).

Miljan Đorđević is with the Department of Computer Science and Information Technology, School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mail: miljandv@gmail.com)

Milan Milivojević is with the Department of Telecommunications, School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mail: msmilance@etf.rs).

Ana Gavrovska is with the Department of Telecommunications, School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mail: anaga777@gmail.com; anaga777@etf.rs).

analysis, are given in Section IV. DeepFake algorithms for generating data (X2Face and First order motion model) [3] - [4] are briefly explained in this Section. This is followed by the production results and SIFT based experimental results with appropriate discussion in Section V. Finally, main conclusions can be found in Section VI.

II. Visual Forgery and DeepFake Production

A. Visual forgery production

Creating visual forgeries has become possible for almost anyone, tools which allow rapid generation of forged videos, images and audio content are on the rise. Forgery detection algorithms analyze the content we suspect was forged and give us their prediction on whether the content was manipulated in any way or not. Unprofessionally implemented copy-move or tamper manipulation techniques are usually easy to detect. The real challenge stems from the recent advancements in the area of Computer Vision and Machine Learning, which have brought forth a class of forgery production methods. They can be easy to set up and use, and most importantly they are faster than common forgery creation mechanisms of the past. One class of recent algorithms which has proven to be particularly concerning are DeepFake generation algorithms [3]-[6].

B. DeepFake production technology development

DeepFake is a video or a photo altering technique based on deep learning [1]. Convolutional neural networks are typically applied, especially GANs (Generative Adversarial Networks) which allow us to train networks in order to recognize the facial features contained within one video or image sequence and then use the model trained in such a way to reapply the learned gestures onto another video or image [3] - [6]. It seems that the term DeepFake comes from the reddit username of this platform's member who is thought to have come up with the idea of training a CNN to replicate human facial expressions [7]. This was posted in December of 2017, which led to an explosive rise in interest for this type of forgery inside Computer Vision communities. Tech leaders have realized the potential of this type of content in 2018, with most of them banning DeepFakes entirely. Although the platforms have forbidden DeepFakes and removed users who post them from their sites. Nowadays, from 2019 there are competitions related to deep forgeries [8].

DARPA (Defense Advanced Research Projects Agency) started to tackle a DeepFake detection issue [9]. Their algorithm used commonly observed errors in the DeepFakes generated at that time to detect manipulated content by detecting double eyebrows and similar artifacts. This algorithm failed to maintain its relevance, since DeepFake professionals managed to circumvent this. Then, several videos of famous politicians surfaced on Facebook, and based on user comments it can be stated that a greater audience was fooled by them [10]. In the May of 2019 there was another great breakthrough in the field with paper entitled "Few-Shot Adversarial Learning of Realistic Neural Talking Head Models"[5] which introduced a model which could generate DeepFake videos based on a single image, and thus made creating videos of famous paintings and people of whom we lack visual recordings: Mona Lisa (Fig. 1), Fyodor Dostoyevsky, Marilyn Monroe, etc. In Fig. 2 showcases of X2Face network pose predicting example are presented [3]. The network tries to transfer head positions from the images on the bottom to another person and the results are the five images on the top. In the X2Face network example image artifacts can still be easily spotted.

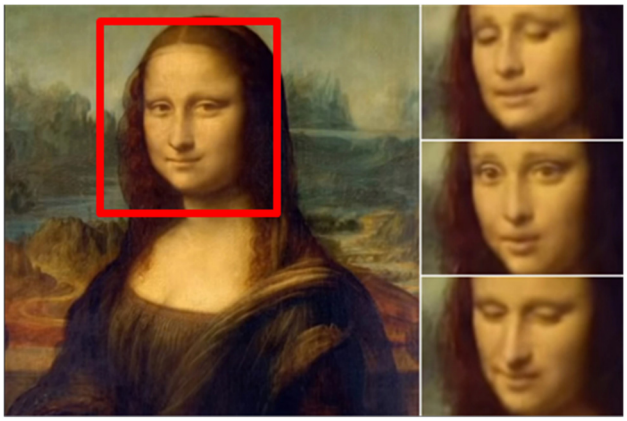

Fig. 1. Deepfakes generated based on portrait paintings - example of Mona Lisa with detected face (left) and Deepfake production (right) [5].

Fig. 2. Original frames are presented in a red box with X2Face network results in the upper row [3].

III. SIFT Features

SIFT (Scale invariant feature transform) is an algorithm used to detect points of interest in an image, so called keypoints. Namely, it is a feature detection algorithm introduced in 2004 by David G. Lowe [2]. Keypoints represent distinguished parts of an image, points of interests which are mostly found in sections with abrupt contrast changes, such are abrupt changes in the image texture and color, as well as object corners. At the point of its first appearance it differed from previous keypoint detection algorithms in its invariance to scale, orientation and to a point illumination and affine transformations. SIFT has proven useful in taking on varying challenges in the fields of computer vision, image retrieval and amongst other applications, can be used for creating panoramas and augmented reality production [2]. In Fig. 3 two examples are presented with calculated SIFT keypoints [2].

Generally speaking, SIFT consists of four main steps: scale-space extrema detection (this step makes SIFT invariant to scale), keypoint localization, orientation assignment (rotation invariance is provided by this step) and generation of keypoint descriptors (this step makes SIFT invariant to illumination to a point).

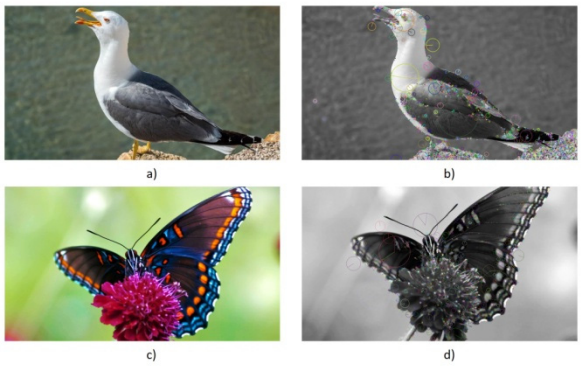

Fig. 3. Two pairs of images showing originals (on the left) and calculated SIFT keypoints (on the right).

In the first step, the preliminary list of keypoints is calculated from an image based on extrema detection. Here, a Gaussian pyramid is formed, four levels in depth with five scaled images on each level. This is done by applying a Gaussian filter to the grayscale image, as well as to each following element in the Gaussian pyramid. At the end of a level, image is downscaled and then the algorithm continues applying Gaussian filters. By calculating the difference between successive images in same scales, and then applying extrema detector on the results, keypoints are calculated. A pixel is a potential keypoint if it represents a local extreme in the surrounding 26 pixels, 8 from the same scale, 9 from the scale upwards and 9 from the downscaled image subtraction result. Reason for using several levels in the pyramid and downscaling lies in the fact that some keypoints are lost with dimension reduction. This allows us to extract relevant keypoints which uniquely identify the image at all scales up to the smallest used in the pyramid [2]. Previously found keypoints are then tested for stability in the next step, by discarding all keypoints with intensity lower than a predefined threshold. Keypoints are additionally rejected through sub-pixel localization and edge removal. Sub-pixel localization consists of approximating the quadric Taylor expansion of the scale space function and then computing its extrema. SIFT also applies the Harris corner detector to remove edges whist keeping corner keypoints. This removes previously poorly chosen keypoints such as are those created by artifacts. In the third and fourth step the final list of keypoints is extracted from the image and filtered throughout the second step. Finally, 128bin feature vector is produced for each keypoint. It is created from 8 bin histograms for each 4x4 sub-block of 16x16 block surrounding the keypoint, having in mind gradient magnitude and orientation [11]. Nearest neighbor approach is used over the vectors generated by the SIFT method over two images and thus the images can be compared. Accidental matches with background keypoints can even be further mitigated [8].

There are several different less and more advanced variants of the SIFT algorithm in the literature, all of which solve some of the issues found in the original or some extended SIFT versions. For example, SURF (Speeded up robust features) is one of the most used amongst them [12]. There are other methods as well: PCA-SIFT (Principal Component Analysis performed on SIFT features), GSIFT (Global information into SIFT), CSIFT (Colored SIFT), ASIFT (Affine-SIFT) [12]. We use the basic SIFT in our experiment because of the nature of our dataset, which makes some of those advanced capabilities of upgraded versions of SIFT still not so relevant

IV. Simulation

In this paper, we perform experiments with DeepFake production and SIFT analysis.

Production was performed using two methods. Namely, we produced deepfakes using X2Face [13] and First order motion models [11]. These two models were chosen because of the difference in their approaches, as well as the difference in the times they were published at. X2Face uses two encoder-decoder networks, an embedding network and a driving network. Both are based on the pix2pix network [14] with changes introduced to its input and output layers. Driving and embedding network differ by the presence of skip connection in embedding network, and by the positioning and size of certain smaller inner layers. Some of the examples are shown in [15].

Deepfakes generated using most of the models are not perfect, and most of them can be detected by the human eye. There are exceptions to this rule however, and humans have been successfully fooled in the past.

An experiment is performed with matching percentages between successive frames based on extracted SIFT features, as in the case of non-reference forgery detection methods. Successive frame matching percentages generated in this way are compared between original and deepfake videos. The analysis is performed using Python programming language, as well as OpenCV cross-platform library for SIFT implementation [11]. In the first part of the Python script we run SIFT algorithm over successive frames of a deepfake video sequence. Feature calculation is based on the ratio test:

$$\text{m.distance} < k * \text{n.distance}, \quad (1)$$

where m and n correspond to the closest-distance and second-closest distance taken into account [8]. This is also repeated for the original video for the purpose of comparison. Threshold was empirically selected (default threshold is k=0.8). In the second part of the algorithm, we calculate the frame matching percentage for both deepfake and original video sequence. The results are postprocessed by filtering with a moving-average filter of window length N=5. Moreover, standard deviation and median value for frame matching percentage are calculated for filtered video results. Statistical analysis was performed, so we calculate mean and standard deviation of the array of mean squared errors (MSE(i)) of absolute difference between original and deepfake frames (i=1,..., total number of frames):

$$\text{MSE}[i]=\text{mse}(\text{abs}(\text{deepfake}[i]-\text{original}[i])). \quad (2)$$

Expression (2) uses a reference video.

We perform experimental analysis on different groups of DeepFakes. In the first group we test it on DeepFakes of unknown origin found on the internet [16]. A comparison is made between results obtained using produced deepfakes based on X2Face [13] and First order motion models [11]. A standard reference dataset VoxCeleb [17] is used for

production purposes since it is often used for training. In this paper mostly animated (like Shrek or Aladin) and historical examples (like Tesla and Pupin) are tested using the two deepfake methods. Video sequences are of different frame rates (10-30fps), where original and corresponding obtained deepfake have the same frame rate. Finally, we adopt the first order motion model to webcamera facial presentation for the purpose of further testing.

V. Experimental Results

A. Production Deepfake results

There are many ways to produce a forgery [18-19]. The first part of production is dedicated to X2Face method. Produced deepfake videos based on VoxCeleb set examples using X2Face method are shown in Fig. 4. Even though it is expected to obtain visually satisfying results, artifacts are obvious. The same method was applied on some historical and animated (or cartoon) images, and artifacts got worse, which is expected since the model was not trained or tested on such examples. The examples of obtained deepfakes are shown in Fig. 5. Deepfake production depends also on the tested image and artifacts are particularly visible when performing a specific action., like mouth opening or looking down, as presented in Fig. 5.

Fig. 4. Produced deepfake videos based on VoxCeleb set using X2Face method.

Fig. 5. Produced deepfake videos based on historical and animated/cartoon images using X2Face method: a) Mihajlo Idvorski Pupin example and b) faces performing specific actions.

The generated deepfake data using First order motion model enabled a better performance from a visual standpoint. This is obvious when generating deepfakes while performing specific actions. Such produced deepfake videos based on VoxCeleb inputs using First order motion model are shown in Fig. 6. These results but for our dataset can be seen in Fig. 7, where deepfake videos are calculated using historical photos and animated characters.

Fig. 6. Produced deepfake videos based on VoxCeleb set using First order motion model.

Fig. 7. Produced deepfake videos based on historical and animated/cartoon images using First order motion model.

B. SIFT-based frame matching results

SIFT method was tested under different circumstances. One of the experiments was with logo from the monitor used for analysis and the same logo image taken from internet. Some of our SIFT comparison results are illustrated in Fig. 8, where a real world logo image is not rotated by 90, 180 and 270 degrees. Resulting matching percentages were mostly similar (roughly in the range 20-30%) showing SIFT rotation invariance.

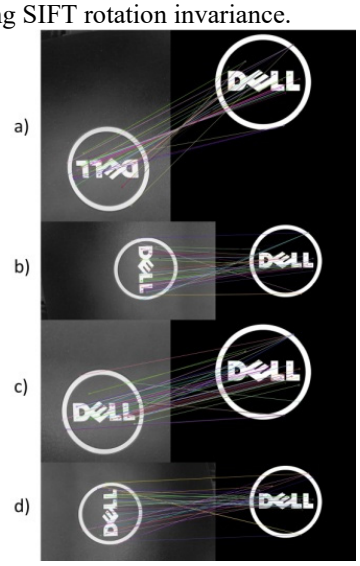

Fig. 8. SIFT based comparison between real world and internet logo image. Matching percentages are: a) 25.8% b) 27.5% c) 25.3% d) 28.9%, respectively.

SIFT-based matching was performed on successive deepfake frames, as well as the original. Using a global

network some typical deepfakes are found, such as "Man of Steel deepfake", "R. Reagen deepfake" and "T. May deepfake" [16], Fig. 9. The obtained experimental results using SIFT method are presented in Fig. 10. In the "Man of Steel deepfake" video a face of an actress is modified. The last two video sequences correspond to original and deepfakes of famous politicians speaking. Sequences are of mp4 format.

Fig. 9. Original and deepfake pairs downloaded from internet: "Man of Steel deepfake" (left), "R. Reagen deepfake" (upper right) and "T. May deepfake" (lower right)

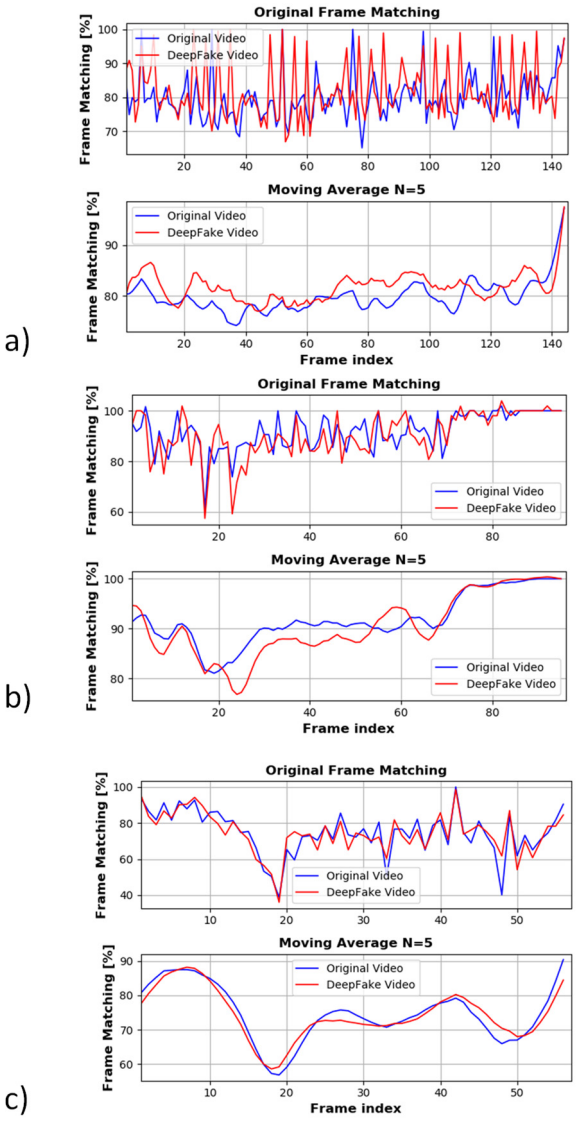

Fig. 10. Original and deepfake matching results for k=0.8 tested on: a) "Man of Steel deepfake", b) "R. Reagen deepfake" and c) "T. May deepfake".

In Table 1 experimental statistical results are shown for this set. The matching percentage is slightly higher in the case of the first and the third video. In other words, slightly higher resemblance or correlation between successive frames exists. In the case of doctored videos of politicians there is a high correlation. Most of the changes are found in the area of mouth and chin. The median values of percentage may be higher, where MSE is the smallest in the case of second video, where the matching statistics are similar for the original and deepfake. The third case shows a high MSE value but similar percentage values. This is why the k parameter should be set accordingly for the third case (k=0.95) in order to obtain improved difference between the original and the deepfake. The frame matching result for k=0.95 is presented in Fig. 11. A larger difference is found due to the rough threshold set for the keypoints.

TABLE 1: PRODUCTION RESULTS FOR ORIGINAL AND DEEPFAKE VIDEO SEQUENCES DOWNLOADED FROM INTERNET

Video sequence examples from internet		Frame rate [fps]	Origin. (median) [%]	Deep-Fake (med.) [%]	MSE
1	"Man of Steel deepfake"	33.33	90.25 ± 1.91 (90.03)	91.52 ± 1.67 (91.45)	0.1164 ± 0.0068
2	"R. Reagen deepfake"	15	97.89 ± 1.93 (97.91)	97.27 ± 2.90 (97.30)	0.0379 ± 0.0101
3	"T. May deepfake"	12.5	89.66 ± 3.31 (89.48)	90.00 ± 2.79 (89.51)	0.3219 ± 0.0058

C. *Experimental analysis results with web camera*

The results of adapting a web camera to deepfake production are presented on "Nikola Tesla" example in Fig. 12. The adaptation is only made for the First order motion model. Two sequences are used for the production and comparison in this case. In Table 2 it can be observed that higher matching percentages are obtained for the deepfakes. Also, higher median values can be found in the "Nikola Tesla" video 1 similarly as in the case of "R. Reagen" example.

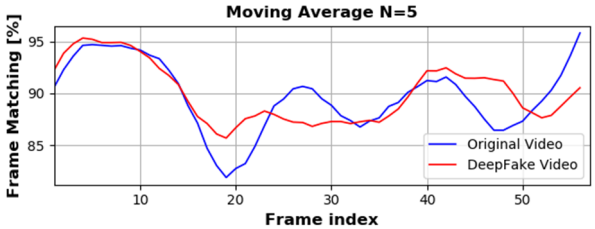

Fig. 11. Frame matching result after filtering for "T. May deepfake" for k=0.95.

Features which give away DeepFakes are most commonly artificial artifacts which such methods sometimes produce, especially surrounding the edges of a human face. The first order motion model network's main advantage lies in relative keypoint matching, from source to target face, which relatively solves the challenge when DeepFake models have when transitioning between faces of different shapes.

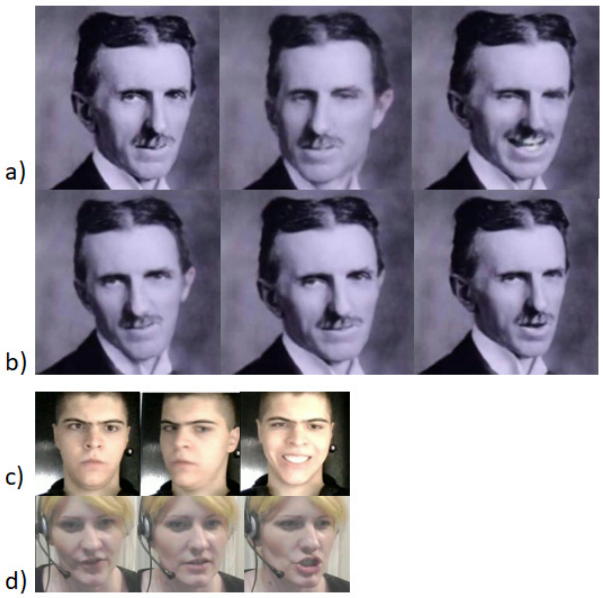

Fig. 12. Deepfake production results adopted to web camera presentation. Nikola Tesla deepfake frames from a) correspond to c) web camera frames. Nikola Tesla deepfake frames from b) corresponds to d) web camera frames.

TABLE 2: PRODUCTION RESULTS FOR ORIGINAL AND DEEPFAKE VIDEO SEQUENCES USING WEB CAMERAS

Video sequence examples from internet		Frame rate [fps]	Origin. (median) [%]	Deep-Fake (med.) [%]
1	"Nikola Tesla" video 1	10	72.85 ± 6.66 (61.6)	78.28 ± 7.12 (79.35)
2	"Nikola Tesla" video 2	30	70.15 ± 6.45 (70.0)	81.86 ± 5.21 (81.19)

VI. CONCLUSION

In this paper we produce and analyze deepfake video sequences using SIFT feature vector. Two methods are applied for the production, like X2Face and First order motion model. Successive deepfake and video frames are matched using the keypoints. The SIFT features showed their advantages such as rotation invariant matching, but it can be considered promising for differentiating original and deepfake videos. Also, a web camera was used for deriving deepfakes from a still image.

The future work will be oriented towards collecting a larger dataset for the purpose of testing from both historical and animated characters, as well as analysis of other possible features and methods for deepfake video detection.

REFERENCES

[1] New York Post, AI brings Mona Lisa to Life, https://nypost.com/2019/05/28/ai-brings-mona-lisa-to-life-loses-signature-smile-in-process/, 28.05.2019.

[2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, 60, no. 2, pp. 91-110, 2004.

[3] O. Wiles, A. Sophia Koepke, and A. Zisserman, "X2face: A network for controlling face generation using images, audio, and pose codes," In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 670-686, 2018.

[4] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, "First Order Motion Model for Image Animation," *Advances in Neural Information Processing Systems*, pp. 7135-7145, 2019.

[5] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky, "Few-shot adversarial learning of realistic neural talking head models," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9459-9468, 2019.

[6] P. Isola, J. Y. Zhu, T. Zhou, and A. A, Efros, "Image-to-image translation with conditional adversarial networks," In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125-1134, 2017.

[7] What are Deepfakes and How Are They Created?, IEEE Spectrum, https://spectrum.ieee.org/tech-talk/computing/software/what-are-deepfakes-how-are-they-created (last accessed in 20.05.2020.)

[8] Deepfake Detection Challenge. Keggle, https://www.kaggle.com/c/deepfake-detection-challenge (last accessed in 20.05. 2020.)

[9] 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, pp. 1-6, November, 2018.

[10] Why 'deepfake' videos mean you can no logner believe what you see, Financial review, https://www.afr.com/technology/can-you-believe-your-eyes-20191028-p534wl (last accessed in 20.05. 2020.)

[11] OpenCVSIFT, https://docs.opencv.org/master/da/df5/tutorial_py_sift_intro.html (last accessed in 01.09.2019.)

[12] J. Wu, C. Zhiming, V. S. Sheng, P. Zhao, D. Su, and S. Gong, "A Comparative Study of SIFT and its Variants," *Measurement science review*, Vol. 13, No. 3, pp. 122-131, 2013.

[13] P. Korshunov, and S. Marcel, "Vulnerability assessment and detection of deepfake videos," In *The 12th IAPR International Conference on Biometrics (ICB)*, pp. 1-6, 2019.

[14] D. Güera, and E. J. Delp. "Deepfake video detection using recurrent neural networks," In *15th IEEE Int. Conf. on Advanced Video and Signal Based Surveillance* (*AVSS*), pp. 1-6, IEEE, 2018.

[15] Github, https://github.com/miljandv/DeepFake-Video-Production-and-SIFT-based-Analysis

[16] M. Đorđević, M. Milivojević, and A. Gavrovska, "DeepFake Video Analysis using SIFT Features," In *2019 27th Telecommunications Forum* (TELFOR), IEEE, pp. 1-4, November 2019.

[17] VoxCeleb, http://www.robots.ox.ac.uk/~vgg/data/voxceleb/ (last accessed in 20.05. 2020.)

[18] P. Aleksandra, N. Glišović, A. Gavrovska, and I. Reljin, "Copy-move forgery detection based on multifractals," *Multimedia Tools and Applications,* pp. 1-24, 2019.

[19] B. Yang, S. Xingming G. Honglei Guo, X. Zhihua, and C. Xianyi, "A copy-move forgery detection method based on CMFD-SIFT," *Multimedia Tools and Applications* 77, no. 1, pp. 837-855, 2018.