A 3D-ASSISTED FRAMEWORK TO EVALUATE THE QUALITY OF HEAD MOTION REPLICATION BY REENACTMENT DEEPFAKE GENERATORS

Sahar Husseini, Jean-luc Dugelay

EURECOM
Digital security Department
Biot, France

Fabien Aili, Emmanuel Nars

DOCAPOSTE
Biometrics Lab
Biot, France

ABSTRACT

In recent years we have assisted the proliferation of deepfakes. The progress concerning both creation and, to a certain extent, automatic detection is spectacular. Nevertheless, there is a lack of protocols concerning the objective evaluation of deepfakes. In this article, we focus on the quality of head motion replication by deepfake generators that use a pilot video of a particular person to animate a single source image of another person. We test several publicly available generators to reproduce particular head movements (rotation around yaw, pitch, and a combination of pitch and yaw). In order to measure how well the deepfake generators replicate head motion, a 3D head model is utilized to render video sequences with the known head pose. Then the generated head movements by deepfake are compared to an exact 3D simulation that can be used as ground-truth. Several measures, such as SSIM and average facial keypoint distance, are used to quantify results.

Index Terms— deepfake, face-reenactment, deepfake evaluation

1. INTRODUCTION

Generating animated videos from a single face image has numerous applications in movie production, image editing, enhancement, and photography. Given a single source image and a driving video, face reenactment methods aim to generate a synthesized video animated by the driver's movement while keeping the identity of the source image. More precisely, when a source image is fed to a face reenactment network, the source person in the image will turn into a puppet, and the driving video will define the target's facial expression, eyes, and head movements. The recent face manipulation techniques [1, 2, 3, 4] utilize generative models such as Encoder-Decoder (ED) networks, Generative Adversarial Networks (GAN) [5], and Variational Auto-Encoders (VAEs) [6] to generate image animation. These recent works based on deep learning have significantly improved the automatic generation of the synthesized videos' quality and realism.

Despite the researcher's concerns about generating face animations, there is a lack of evaluation techniques. For instance, the lack of the ground-truth for the cross-identity reenactment does not allow researchers to evaluate the results by metrics that require pixel-by-pixel comparison such as Structural Similarity Index (SSIM), Peak Signal-to-Noise Ratio (PSNR) [7], and the facial keypoint distance. To overcome this problem, the face-reenactment methods usually evaluate the generated image quality by performing a subjective test, which often requires a visual inspection of the generated image. Or they evaluate self-reenactment results, which does not consider the identity leakage (generated face resembles the driver facial features).

In order to perform pixel-by-pixel evaluation for the cross-reenactment methods, we propose an approach to generate a face video dataset with the ground-truth. Furthermore we propose a protocol to evaluate different methods using the proposed dataset. More precisely, we generate a rich face dataset, that can be flexibly controlled; We can move the camera and render the face features in the desired view. To control the head movement, a 3D head point cloud is located in a scene and rendered in the desired head pose. Rendering faces in the 3D environment allow us to fully control the head pose, camera parameters, and scene illumination. The proposed evaluation protocol focuses exclusively on global motion, which means that the facial expression is constant (neutral) in all video frames.

This paper is organized as follows: Section 2 presents related works. In section 3 we present the proposed methodology and sections 4 and 5 present the experiment and conclusion, respectively.

2. RELATED WORK

The face-reenactment evaluation techniques can be classified into three main categories: self-reenactment evaluation, subjective test evaluation, and cross-reenactment evaluation which is based on features/embedding extracted by a pre-trained network. During self-reenactment, one video frame is considered as a source image, and the rest of the frames from the same video are used to animate the source image. Since the source and driver identity belong to the same video sequence, the driver frames can afterward be used as ground-truth to which the generated images can be compared. The self-reenactment evaluation is usually performed with the metrics that require pixel-by-pixel comparisons, such as image quality metrics, SSIM and PSNR. Further, it is used to compare the facial keypoint distance between the source image and the driver image. For example, the first-order motion model (FOMM) [4] reports the L1 error, Average Euclidean Distance (AED), and Average Keypoint Distance (AKD) between the generated and the ground-truth images for the self-reenactment results. Similarly the X2Face [1] computes the L1 error between the generated and the ground-truth images.

The cross-reenactment evaluation is performed when the source face is animated by a different identity. Currently, the generated images by cross-reenactment can not be evaluated by the metrics requiring the exact ground-truth. To overcome the lack of the ground-truth problem, usually, a subjective test is performed for the quality assessment. In [4], the cross-reenactment quality assessment is performed by a user study where a source image, the driving video, and the corresponding results for different methods are shown to the users, and the user selects the most realistic image animation. On the other hand, the few-shot vid2vid method [3] performs AB tests where they provide the user with videos from two different methods and ask user to choose the one with better quality.

To evaluate the cross-reenactment results quantitatively and overcome the lack of the ground-truth, the researchers use a set of metrics that do not require the ground-truth. First, a pretrained network is used to extract some embedding/features from the generated and the driver image, and then the distance between these two embeddings is computed. For instance, recent face-reenactment methods [2, 8, 9] evaluate the identity preservation by computing Cosine Similarity (CSIM) of embedding vectors generated by pre-trained face recognition model [10]. Furthermore, Ha et al. [2] use pretrained networks to estimate the head pose angles and facial action units of generated image and compare the result with the driver's head pose and action unit. Using pre-trained networks to extract and compute the error between the feature embeddings has partially solved the cross-reenactment evaluation; however, there is still a lack of metrics for computing the error pixel-by-pixel. In this paper we propose an approach to generate a face video dataset with ground-truth and a protocol to evaluate the cross-reenactment results.

3. PROPOSED METHODOLOGY

We use Pyrender 3D environment to create the face video dataset. A 3D head model is inverse projected to create the 2D video sequences where the head pose is controlled and known for each frame. Thanks to our protocol, it is possible to create an exact 2D video ground-truth with exact geometrical information that can be later compared to the deepfake results, which utilize only the 2D information to create the synthesized images. The dataset generation and the evaluation protocol are explained in section 3.1.

3.1. Dataset generation and preprocessing

We use an existing 3D face dataset, namely, FaceScape [11]. This dataset contains high-quality 3D face models captured using a multi-view system (MVS) consisting of 68 Digital Single-Lens Reflex (DSLR) cameras. The face models are captured from different subjects under controlled position, and expression. The dataset provides a detailed mesh model without RGB values for each subject, and the corresponding multi-view RGB images with their intrinsic and extrinsic parameters. We use the detailed mesh with the neutral expression and the provided camera parameters to render the depth for each RGB image. Then we use the inverse projection to map 2D data to the 3D point cloud. During head point cloud reconstruction, we apply auto-white balance [12] to all the images to normalize the effect of the scene's illumination [13].

Having a 3D point cloud head model, we can locate it in the desired scene, define a camera with specific parameters and render it in the desired head pose [14]. The size of rendered images is 256x256, and to improve the quality of the rendered images, we further apply the super-resolution method proposed in [15]. Figure 1, from left to right, illustrates the detailed mesh model, Multi-view images with the corresponding depth, 3D point cloud head model with RGB values, and images rendered from the head point cloud with the desired head and camera pose.

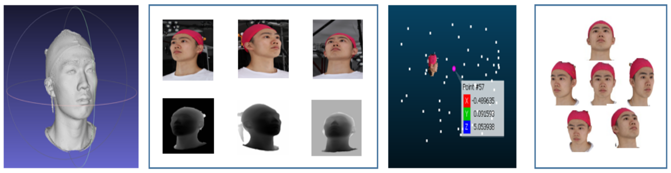

Fig. 1: Dataset generation and preprocessing

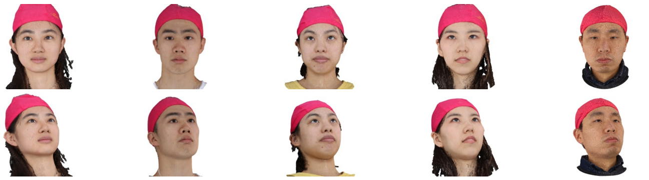

Fig. 2: The sample images illustrate head rotations around combinations of pitch and yaw-axes, with each row showing a different identity at the same angle. The characters are depicted with neutral expressions.

3.2. Evaluation protocol

To comprehend which head movements are more challenging for the face-reenactment method, we generated video sequences, each containing movement around a specific axis in the Euler angles system. More specifically, three types of head movement are investigated: rotation around the pitch, yaw, and a combination of pitch and yaw axis. Each video sequence consists of 100 frames; in each frame, we rotate the head by 0.30 degrees toward the desired axis. It means that in the first frame, we have the frontal head; in the last frame, the head is rotated 30 degrees around the desired axis. Furthermore, for each head rotation, 5 video sequences are created, each having different identities with a neutral expression. Figure 2 presents sample images containing a combination of pitch and yaw head rotation. The first row shows images corresponding to the first video frame, and the second row corresponds to the same video where the head is rotated by 23 degrees. From left to right, the identities' names are A, B, C, D, and E, which refer to the identities 122, 212, 340, 344, and 359 in the original paper [11]. Identity A serves as a source, while the other identities are utilized to drive identity A.

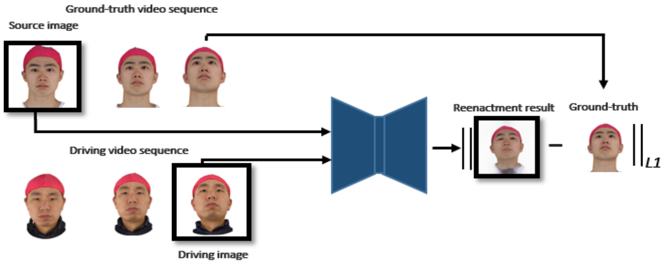

Fig. 3: Proposed protocol: Computing error between generated image and the ground-truth

To evaluate self-reenactment results, one video sequence corresponding to identity A is evaluated. The first frame of video A is considered as a source, and the rest of the frames in the same video are used to drive the source face. To evaluate cross-reenactment, first, the video sequence corresponding to identity A is considered as a ground-truth, and its first frame is considered as a source image. Since the head pose and expression in the remaining video sequences, corresponding to identities B, C, D, and E, match with identity A, they are used to drive the source image. Finally, four cross-reenactment results are generated in which the identity matches the source image and the expression and head movement matches the driving images.

The cross-reenactment evaluation protocol is as follows: 1) select the first frame of the ground-truth video as a source image 2) use one of the driving videos to animate the source image 3) feed the source image and the driving video frame to one of the face reenactment methods 4) having the result from the face reenactment method, one can compare the result with the ground-truth. Figure 3 illustrates the evaluation protocol, where we have two video sequences, one corresponding to ground-truth and the other to the driving sequence. As can be seen, the first frame of the ground-truth sequence is selected as a source image. Then, another video frame is used to drive the source image. Finally, both the source frame and driving frame are fed to the face reenactment method. In this step, the resulting image must match the ground-truth. Then, having the face-reenactment result and the ground-truth, one can compute the error between them.

In our protocol, we evaluate four face reenactment methods, namely, X2Face [1], Fs-vid2vid [3], FOMM [4], and ICface [16]. The performance of these methods is evaluated in terms of identity preservation, pose replication, and image quality. First, to investigate the capability of the model to reenact the driver's pose properly, we compute the Average Keypoint Distance (AKD) between the ground-truth image and face-reenactment result using MediaPipe library [17], which extracts 468 keypoints over the face. To inspect the image quality, we compute the Masked-SSIM (M-SSIM), where the measurements are restricted to the face region. Furthermore, we compute the Cosine Similarity (CSIM) of embedding vectors generated by the pre-trained face recognition model [10] to evaluate the quality of identity preservation.

4. EXPERIMENTAL RESULTS

This section presents the evaluation results for the proposed dataset and protocol. Figure 4 visualizes a selection of the results during head rotation around the yaw-axis. The images on the right side present the self-reenactment results, and the images on the left side are animated by identity B. The source frame to all the reenactment models is the first frame from ground-truth (identity A); therefore, the identity in all the reenactment results should match this identity.

4.1. Results and discussion

We report the self-reenactment and cross-reenactment results over our dataset in Table 1. The self-reenactment values are computed using simply one video sequence and the cross-reenactment results are the average of four generated images (e.i. the four videos generated by B, C, D, and E identities). Moreover, to compare with the existing self-reenactment protocol, we have included in table 2 the results from existing works [18], [9]. Although the evaluation type and results are not the same as the deepfake generator reported in the literature, they are highly consistent. The results gained by our protocol depict similar results compared to the existing self-reenactment protocol; The FOMM generator outperforms other methods, and ICFace gets the highest error. However, in our proposed protocol, the generated images are compared with ground-truth faces with the exact source identity and head pose derived from the driver. By this means, it depicts which of these generators keeps identity for cross-reenactment as the subject rotates her/his head from the frontal pose (see Figure 4, 5).

	Method	AKD↓ self	AKD↓ cross	CSIM↑ self	CSIM↑ cross	M-SSIM↑ self	M-SSIM↑ cross
pitch	X2Face [1]	1.33	2.76	0.52	0.45	0.81	0.71
	Fs-vid2vid[3]	2.36	4.07	0.35	0.32	0.62	0.53
	FOMM [4]	1.18	1.90	0.49	0.48	0.81	0.74
	ICFace [16]	8.61	8.45	0.31	0.32	0.49	0.48
yaw	X2Face [1]	1.13	3.03	0.78	0.68	0.79	0.69
	Fs-vid2vid [3]	2.36	4.07	0.18	0.21	0.69	0.61
	FOMM [4]	0.75	1.82	0.84	0.67	0.81	0.74
	ICFace [16]	10.75	10.89	0.30	0.37	0.47	0.47
pitch-yaw	X2Face [1]	3.14	4.29	0.75	0.68	0.70	0.64
	Fs-vid2vid [3]	7.90	9.18	0.19	0.21	0.51	0.48
	FOMM [4]	1.90	2.27	0.75	0.66	0.77	0.70
	ICFace [16]	14.90	15.16	0.34	0.37	0.42	0.41

Table 1: Evaluation result for the self and cross-reenactment over the pitch, yaw, and combination of pitch and yaw axis. Upward/downward pointing arrows correspond to metrics that are better when the values are higher/lower.

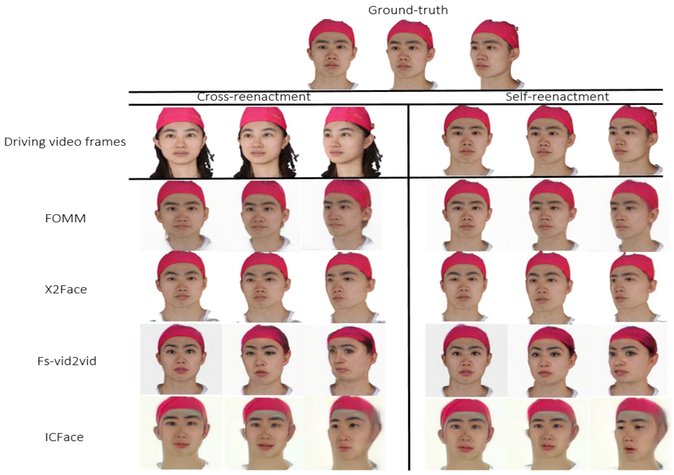

Fig. 4: A few example images from identity A being self-reenacted and cross-reenacted by identity B during head rotation around the yaw axis.

Method	AKD↓	CSIM ↑	SSIM ↑
X2Face	0.75	0.48	0.65
Fs-vid2vid	N/A	0.41	0.67
FOMM	0.44	0.61	0.75
ICFace	1.71	0.30	0.54

Table 2: The evaluation results of self-reenactment setting on VoxCeleb2 dataset [18, 9].

In figure 4, we present a detailed analysis of the errors associated with four different face reenactment methods, thereby highlighting their respective sensitivities to changes in head poses relative to the initial frontal pose. This examination allows for a more comprehensive understanding of the comparative performance of these methods. As can be seen, the four deepfake methods produce animated faces close to each other, near the frontal head position. Thereafter, it is difficult to tell from the frontal head position how much the results of deepfake generators differ. The metrics error for X2Face, Fs-vid2vid and ICFace are low until a certain critical viewing radius (5 degrees) and, after that, increase sharply. However, the curve corresponding to the FOMM method shows that rotating the head by 5 degrees around the combination of pitch and yaw axis produces roughly the same error as rotating the head by 30 degrees.

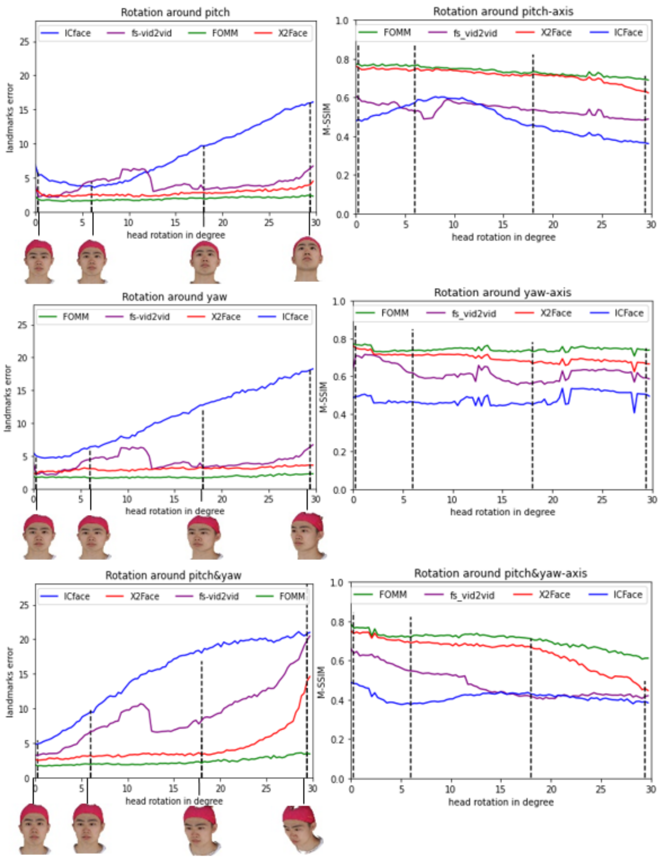

Fig. 5: The AKD and M-SSIM scores per frame are shown for four different reenactment methods: FOMM, X2Face, Fs-vid2vid, and ICFace. The scores are plotted for head rotations around the pitch (first row), yaw (second row), and a combination of pitch and yaw-axis (third row).

5. CONCLUSION

This paper introduces a protocol for the objective evaluation of face reenactment methods when the source and driving identities differ (cross-reenactment). To accomplish this, we utilize a 3D head model to generate video sequences in which the head is rotated towards a desired axis. The head pose is precisely controlled and known for each frame of the video. Four publicly available models were examined to determine their ability to replicate head movements. The results show that replicating the rotation around a combination of pitch and yaw axis is a more difficult task for the reenactment models, as evidenced by the higher error values observed in this scenario.

6. REFERENCES

[1] Olivia Wiles, A Koepke, and Andrew Zisserman, "X2face: A network for controlling face generation using images, audio, and pose codes," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 670–686.

[2] Sungjoo Ha, Martin Kersner, Beomsu Kim, Seokjun Seo, and Dongyoung Kim, "Marionette: Few-shot face reenactment preserving identity of unseen targets," in *Proceedings of the AAAI conference on artificial intelligence*, 2020, vol. 34, pp. 10893–10900.

[3] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Jan Kautz, and Bryan Catanzaro, "Few-shot video-to-video synthesis," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[4] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe, "First order motion model for image animation," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[6] Diederik P Kingma and Max Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[7] Alain Hore and Djemel Ziou, "Image quality metrics: Psnr vs. ssim," in *2010 20th international conference on pattern recognition*. IEEE, 2010, pp. 2366–2369.

[8] Egor Zakharov, Aleksei Ivakhnenko, Aliaksandra Shysheya, and Victor Lempitsky, "Fast bi-layer neural synthesis of one-shot realistic head avatars," in *European Conference on Computer Vision*. Springer, 2020, pp. 524–540.

[9] Lele Chen, Guofeng Cui, Ziyi Kou, Haitian Zheng, and Chenliang Xu, "What comprises a good talking-head video generation?," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020.

[10] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.

[11] Haotian Yang, Hao Zhu, Yanru Wang, Mingkai Huang, Qiu Shen, Ruigang Yang, and Xun Cao, "Facescape: A large-scale high quality 3d face dataset and detailed riggable 3d face prediction," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[12] Mahmoud Afifi and Michael S Brown, "Deep white-balance editing," in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2020, pp. 1397–1406.

[13] Sahar Husseini, Pouria Babahajiani, and Moncef Gabbouj, "Color constancy model optimization with small dataset via pruning of cnn filters," in *2021 9th European Workshop on Visual Information Processing (EUVIP)*. IEEE, 2021, pp. 1–6.

[14] Pouria Babahajiani, "Geometric computer vision: Omnidirectional visual and remotely sensed data analysis," 2021.

[15] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 136–144.

[16] Soumya Tripathy, Juho Kannala, and Esa Rahtu, "Icface: Interpretable and controllable face reenactment using gans," *arXiv preprint arXiv:1904.01909*, 2019.

[17] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, et al., "Mediapipe: A framework for perceiving and processing reality," in *Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR)*, 2019, vol. 2019.

[18] Han Xue, Jun Ling, Anni Tang, Li Song, Rong Xie, and Wenjun Zhang, "High-fidelity face reenactment via identity-matched correspondence learning," *ACM Transactions on Multimedia Computing, Communications and Applications*, 2022.