Supplementary Material of Detecting and Recovering Sequential DeepFake Manipulation

Rui Shao, Tianxing Wu, and Ziwei Liu*

S-Lab, Nanyang Technological University
{rui.shao, twu012, ziwei.liu}@ntu.edu.sg
https://rshaojimmy.github.io/Projects/SeqDeepFake

1 Implementation Details

Implementation is in PyTorch. To be comparable in the number of parameters, we adopt a transformer model with 2 encoder and 2 decoder layers with 4 attention heads. For the training schedule, we employ a 20-epochs warm-up strategy. The initial learning rate is set as $1e-3$ for transformer part and $1e-4$ for CNN part, with a decay factor of 10 at 70 and 120 epochs, totally 170 epochs. We use the SGD momentum optimizer with weight decay set as $1e-4$. We use a mini-batch size of 32 per GPU and 4 GPUs in total. We set $\lambda = 4$. Model selection for evaluation is conducted by considering the trained model that has produced the best accuracy on the validation set.

2 Baseline Methods

The most straightforward solution for detecting Seq-Deepfake manipulation is to regard it as a multi-label classification problem [8]. It treats all manipulations in the sequences as independent classes and classifies the manipulated images into multiple manipulation classes. Specifically, we design a simple multi-label classification network (denoted as **Multi-Cls**) as one of the baselines. We use ResNet-34 [4] and ResNet-50 [4] pre-trained on ImageNet [2] dataset as backbones for the multi-label classification network, which is concatenated with N single linear-layer branches as N classification heads ($N = 5$ as maximum manipulation steps are 5 in Seq-Deepfake dataset). Moreover, we study a more complex transformer structure modified for our problem. **DETR** [1] is a popular transformer devised for end-to-end object detection. This model detects input images' bounding boxes and corresponding object classes conditioned on object queries. We revise this model by replacing the object queries with annotations of manipulation sequences and only preserve the output of object classes. Furthermore, to examine the performance of existing deepfake detection methods for our research problem, we adapt three state-of-the-art deepfake detection methods, a Dilated Residual Network variant (**DRN**) [9], a two-stream network modeling the correlation between high-frequency features and regular RGB features

* Corresponding author

(**Two-Stream**) [7], and a multi-attentional deepfake detection (**MA**) [10], into multi-label classification setting. To be specific, we replace their binary label classifier with multiple classification heads to classify sequential manipulations. Please note since all of the above baselines are only able to predict the facial manipulation with fixed length ($N = 5$), 'no manipulation' class will be padded into the annotation sequence if its length is shorter than N so that we can keep the same length between predictions and annotation sequences for training.

3 Evaluation Metrics

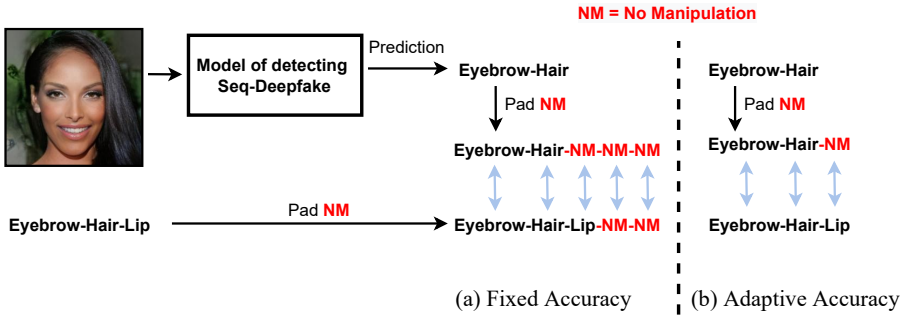


Fig. 1: Comparison between two evaluation metrics (a) Fixed Accuracy and (b) Adaptive Accuracy.

As illustrated in Fig. 1, we elaborate on two evaluation metrics proposed in the experiment for our new task.

- **Fixed Accuracy (Fixed-Acc):** As mentioned in the main paper, in the training process, since all of the baselines are only able to predict the facial manipulation with fixed length ($N = 5$), 'no manipulation' class will be padded into the annotation sequence if its length is shorter than N so that we can keep the same length between predictions and annotation sequences for training. Following this strategy, as shown in Fig. 1, under the evaluation metric of Fixed Accuracy, given the prediction, such as 'Eyebrow-Hair', from the model of detecting Seq-Deepfake, we first pad 'no manipulation' class into it to form the padded prediction sequence as 'Eyebrow-Hair-NM-NM-NM' (NM means 'no manipulation' class) so that we can obtain the prediction with fixed N-length ($N = 5$). To keep the same length between predictions and annotation sequences for evaluation, we pad 'no manipulation' class into the annotation of manipulation sequences as well, denoted as 'Eyebrow-Hair-Lip-NM-NM', and compare each manipulation class in the predicted sequences with its corresponding annotation to calculate the evaluation accuracy.

– **Adaptive Accuracy (Adaptive-Acc):** Moreover, since the proposed method exploits sequential information to detect facial manipulation sequences based on the auto-regressive mechanism, predictions will be automatically stopped once predicting the EOS token. Thus, the proposed method can detect facial manipulation sequences with adaptive lengths. To conduct the evaluation in this scenario, as illustrated in Fig. 1, the second type of evaluation is devised, which compares predicted manipulations and corresponding annotations within the maximum steps of manipulations ($N = 3$ in Fig. 1 and we just pad one 'no manipulation' class into prediction sequence) between them. This makes the evaluation focus more on accuracy of manipulations.

4 Multi-head version of SECA

Similar to [3], we extend the basic version of Spatially Enhanced Cross-Attention (SECA) introduced in the main paper into multi-heads version, which enhances cross-attention features differently for different cross-attention heads. As mentioned in Eq. 2 in the main paper, the basic version of SECA estimates the 2-dimensional coordinates corresponding to spatial centers $[t_h, t_w]$. Similarly, the multi-head version of SECA estimates a head-shard spatial center $[t_h, t_w]$ and then predicts a head-specific center offset $[\triangle t_{h,i}, \triangle t_{w,i}]$ and corresponding head-specific scales $[r_{h,i}, r_{w,i}]$ for i-th cross-attention head. In this way, we generate i-th head-specific Gaussian-shape spatial weight map M_i based on the i-th head-specific center $[t_h + \triangle t_{h,i}, t_w + \triangle t_{w,i}]$ and scales $[r_{h,i}, r_{w,i}]$ as:

$$M_i(h, w) = \exp\left(-\frac{(h - (t_h + \triangle t_{h,i}))^2}{\lambda r_{h,i}^2} - \frac{(w - (t_w + \triangle t_{w,i}))^2}{\lambda r_{w,i}^2}\right) \tag{1}$$

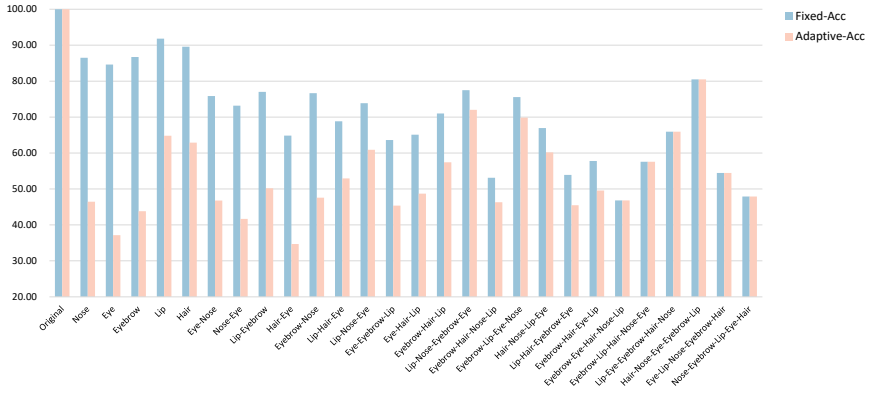
Based on this, we can calculate the features of sequential relation f_i^{seq} from i-th cross-attention head enhanced by the i-th SECA as follows:

$$f_i^{seq} = \text{Softmax}(K_i^T Q_i \sqrt{d} + log M_i) V_i, \tag{2}$$

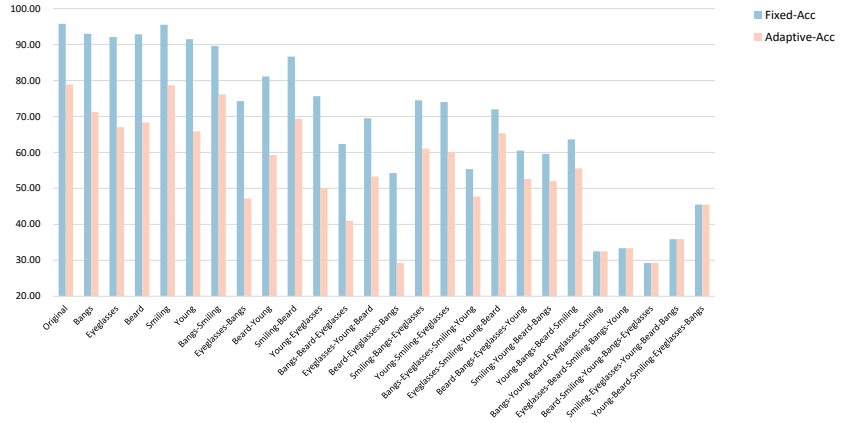
Different from basic version of SECA, above Eq. 2 shows that in the multi-head version of SECA, the cross-attention of the i-th head is element-wise added with logarithm of i-th head-specific spatial weight map M_i, which contributes to a more adaptive and specific enhanced cross-attention. Experiments regarding the proposed method in the main paper are all carried out based on the multi-head version of SECA.

5 Accuracy For Each Manipulation Sequence

As mentioned in the main paper, we generate 28 types of manipulation sequences based on facial components manipulation while 26 types of manipulation sequences based on facial attributes manipulation. To provide a more detailed analysis, in this section, we plot accuracy for each manipulation sequence in both



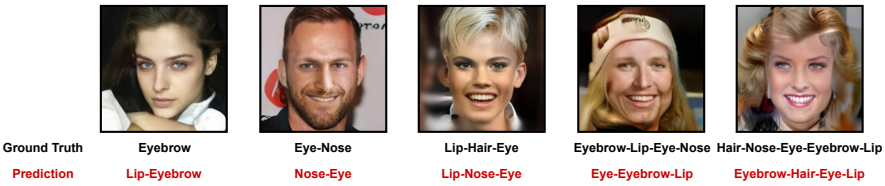
(a) Accuracy on sequential facial components manipulation dataset



(b) Accuracy on sequential facial attributes manipulation dataset

Fig. 2: Accuracy for each manipulation sequence.

facial manipulation methods as shown in Fig. 2. It can be observed that diverse accuracy performance are achieved for different manipulation sequences, ranging from 46.81% to 100% under Fixed-Acc and 34.69% to 100% under Adaptive-Acc in sequential facial components manipulation, while ranging from 29.25% to 95.75% under Fixed-Acc and 29.21% to 78.88% under Adaptive-Acc in sequential facial attributes manipulation. This demonstrates various manipulation sequences are challenging for detection and there exist some extremely hard cases. Therefore, we should further improve our method to cope with all types of manipulation sequences in the future. Furthermore, it can be seen from Fig. 2 that the accuracy gap between two evaluation metrics, Fixed-Acc and Adaptive-Acc, decreases along with the length of sequence increases. This is because the padded 'no manipulation' class is fewer in the longer manipulation sequence



(a) Failure cases on sequential facial components manipulation dataset

(b) Failure cases on sequential facial attributes manipulation dataset

Fig. 3: Examples of failure cases.

when evaluating under Adaptive-Acc, which is closer to the evaluation under Fixed-Acc.

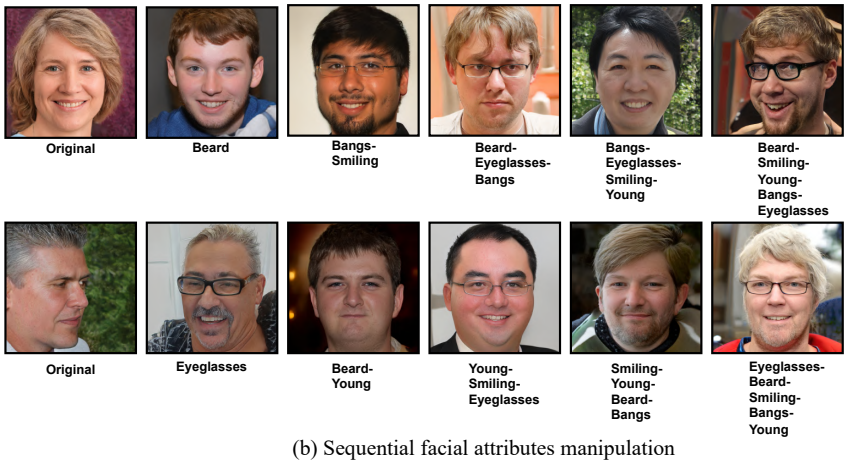
6 Failure Cases

To provide a deeper understanding for our novel task and method, we display some failure cases produced by the proposed method as illustrated in Fig. 3. From Fig. 3, it can be seen that there exist diverse failure cases, including wrong predictions with respect to manipulation type, sequence order, sequence length, etc. This validates that it is quite difficult for our novel research problem since we need to detect facial manipulation sequences in terms of correct manipulation types, orders and lengths simultaneously from hyper-realistic face images with subtle sequential manipulations. This motivates us to continually improve the performance of the proposed SeqFakeFormer to tackle such a challenging task in the future.

7 Sequential Deepfake Dataset

We display more samples from the generated large-scale Sequential Deepfake (Seq-Deepfake) dataset in Fig. 4. As shown in Fig. 4, based on two different facial manipulation methods, facial components manipulation [6] and facial attributes manipulation [5], various sequential facial manipulations are produced with diverse manipulation steps, expressions, ages, and genders.

(a) Sequential facial components manipulation



(b) Sequential facial attributes manipulation

Fig. 4: Illustration of Seq-Deepfake dataset. Samples of Seq-Deepfake are provided with annotations of manipulation sequences.

8 Face Recovery

Fig. 5 shows more samples regarding the face recovery based on correct and wrong facial manipulation sequences. As can be observed in Fig. 5, once we detect the correct facial manipulation sequence, *i.e.* correct manipulations ordered with correct manipulation steps, we can recover original face by performing face attribute manipulation based on the inverse order of detected facial manipulation sequence (process with green arrow). In contrast, recovering the face image with wrongly ordered manipulation sequences may encounter different problems, such as incomplete recovery of age, smile, glasses, bangs, etc. (process with red arrow).

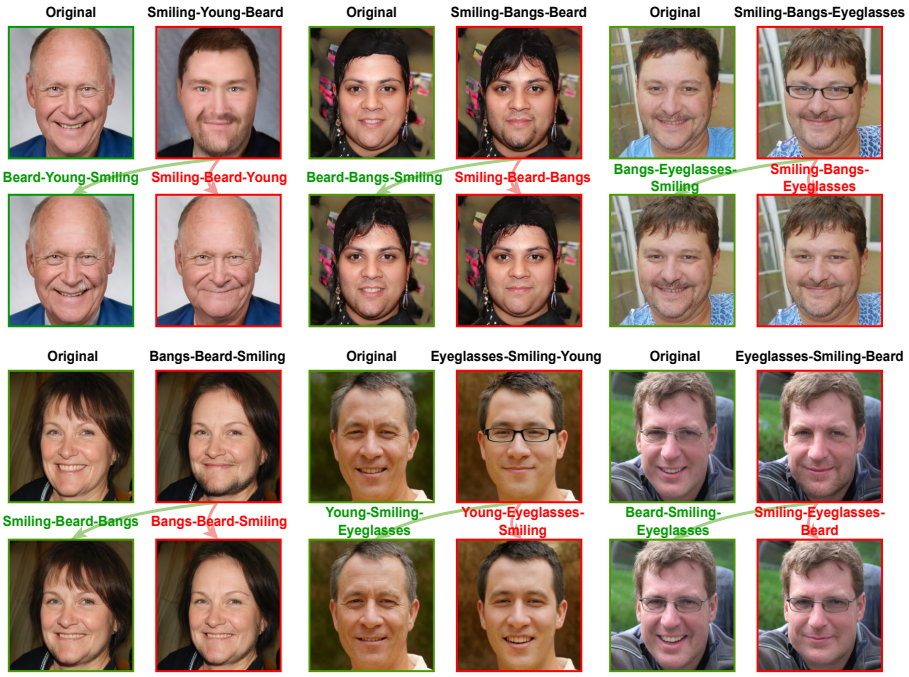


Fig. 5: Face recovery based on correct and wrong facial manipulation sequences.

References

1. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV. pp. 213–229 (2020)
2. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
3. Gao, P., Zheng, M., Wang, X., Dai, J., Li, H.: Fast convergence of detr with spatially modulated co-attention. In: CVPR (2021)
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
5. Jiang, Y., Huang, Z., Pan, X., Loy, C.C., Liu, Z.: Talk-to-edit: Fine-grained facial editing via dialog. In: ICCV (2021)
6. Kim, H., Choi, Y., Kim, J., Yoo, S., Uh, Y.: Exploiting spatial dimensions of latent in gan for real-time image editing. In: CVPR (2021)
7. Luo, Y., Zhang, Y., Yan, J., Liu, W.: Generalizing face forgery detection with high-frequency features. In: CVPR (2021)
8. Wang, H., Liu, W., Bocchieri, A., Li, Y.: Can multi-label classification networks know what they don't know? NeurIPS (2021)
9. Wang, S.Y., Wang, O., Owens, A., Zhang, R., Efros, A.A.: Detecting photoshopped faces by scripting photoshop. In: CVPR (2019)
10. Zhao, H., Zhou, W., Chen, D., Wei, T., Zhang, W., Yu, N.: Multi-attentional deepfake detection. In: CVPR (2021)