ForgeryNet: A Versatile Benchmark for Comprehensive Forgery Analysis

Yinan He[1,2*] Bei Gan[2*] Siyu Chen[2*] Yichun Zhou[2,3*]
Guojun Yin[2] Luchuan Song[4†] Lu Sheng[3] Jing Shao[2‡] Ziwei Liu[5]
[1]Beijing University of Posts and Telecommunications [2]SenseTime Research [3]Beihang University
[4]University of Science and Technology of China [5]Nanyang Technological University
heyinan@bupt.edu.cn {ganbei,chensiyu,yinguojun,shaojing}@sensetime.com
{buaazyc,lsheng}@buaa.edu.cn slc0826@mail.ustc.edu.cn ziwei.liu@ntu.edu.sg

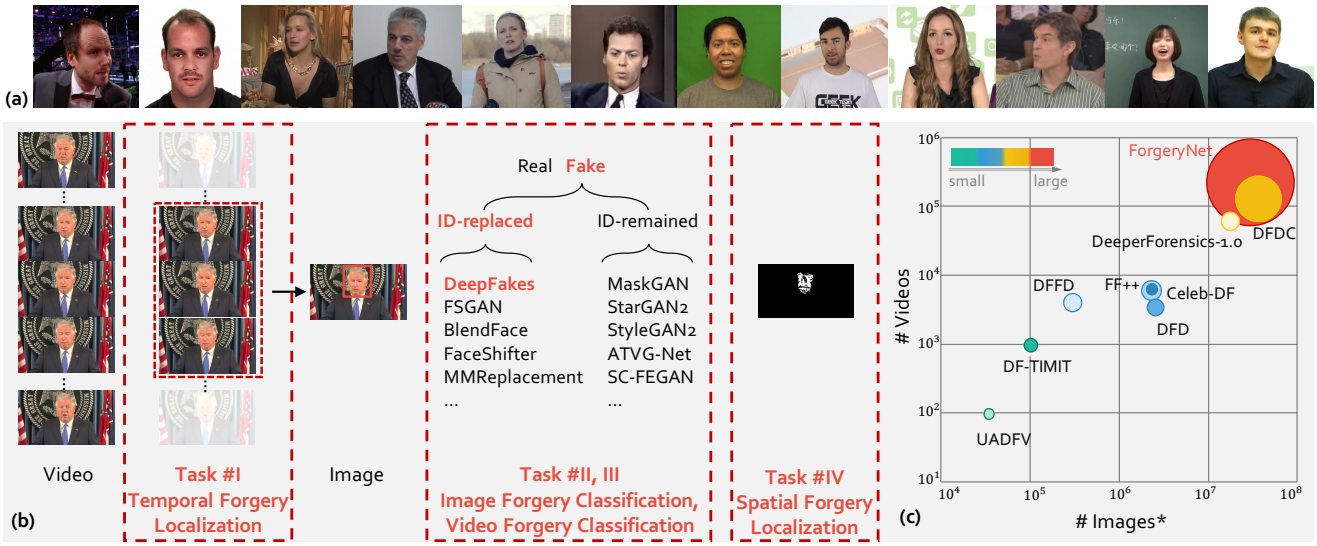

Figure 1: ForgeryNet is a new mega-scale face forgery dataset with comprehensive annotations and four forgery analysis tasks. It contains thousands of subjects, various manipulation methods and diverse re-rendering processes. In (a), can you distinguish which images are forged?

Abstract

The rapid progress of photorealistic synthesis techniques have reached at a critical point where the boundary between real and manipulated images starts to blur. Thus, benchmarking and advancing digital forgery analysis have become a pressing issue. However, existing face forgery datasets either have limited diversity or only support coarse-grained analysis.

To counter this emerging threat, we construct the ForgeryNet dataset, an extremely large face forgery dataset with unified annotations in image- and video-level data across four tasks: 1) ***Image Forgery Classification****, including two-way (real / fake), three-way (real / fake with identity-replaced forgery approaches / fake with identity-remained forgery approaches), and* n*-way (real and* 15 *respective forgery approaches) classification. 2)* ***Spatial Forgery Localization****, which segments the manipulated area of fake images compared to their corresponding real images. 3)* ***Video Forgery Classification****, which re-defines the video-level forgery classification with manipulated frames in random positions. This task is important because attackers in real world are free to manipulate any target frame. and 4)* ***Temporal Forgery Localization****, to localize the temporal segments which are manipulated. ForgeryNet is by far the largest publicly available deep face forgery dataset in terms of data-scale (2.9 million images, 221,247 videos), manipulations (7 image-level approaches, 8 video-level approaches), perturbations (36 independent and more mixed perturbations) and annotations*

*Equal contribution.
†Work done during an internship at SenseTime Research.
‡Corresponding author.
§https://yinanhe.github.io/projects/forgerynet.html

The label of images in Fig. 1(a)(from left to right): fake, fake, fake, fake, fake, fake, fake, real, fake, real, fake, real[0]

(6.3 million classification labels, 2.9 million manipulated area annotations and 221,247 temporal forgery segment labels). We perform extensive benchmarking and studies of existing face forensics methods and obtain several valuable observations. We hope that the scale, quality, and variety of our ForgeryNet dataset will foster further research and innovation in the area of face forgery classification, as well as spatial and temporal forgery localization etc.

1. Introduction

Photorealistic facial forgery technologies, especially recent deep learning driven approaches [23, 38, 49], give rise to widespread social concerns on potential malicious abuse of these techniques to eye-cheatingly forge media (*i.e*., images and videos, *etc*.) of human faces. Therefore, it is of vital importance to develop reliable methods for face forgery analysis[1], so as to distinguish *whether* and *where* an image or video is manipulated.

Most recent progress about face forgery analysis are sparked by gathering of face forgery detection datasets [18, 52] and early attempts of profiling intrinsic characteristics within the forgery images. However, performances on most datasets have already saturated (*i.e*. over 99% accuracy [26, 32, 46, 61]) due to their limited scales (*e.g*. number of images/videos and subject identities) and limited diversity (*e.g*. forgery approaches, scenarios, realistic perturbations, *etc*.). Moreover, in practical applications, it is often required to detect forged faces by locating tampered areas in an image and/or manipulated segments in an untrimmed video, rather than merely providing a binary label.

In this paper, we construct a new mega-scale dataset named ForgeryNet with comprehensive annotations, consisting of two groups (*i.e*. image- and video-level) and four tasks for real-world digital forgery analysis. We carefully benchmark existing forensics methods on ForgeryNet. Extensive experiments and in-depth analysis show that this larger and richer annotated dataset can boost the development of next-generation algorithms for forgery analysis. Specifically, ForgeryNet brings several unique advantages over existing datasets.

(1) Wild Original Data. Most current datasets are captured under controlled conditions (*e.g*. environment, angles and lighting). We collect original data with diversified dimensions of angle, expression, identity, lighting, scenario and *etc*. from four datasets [7, 13, 20, 45]. Note that all the original data have a *Creative Commons Attribution* license that allows to share and adapt the material.

[1]In this paper, the definition of the term "face forgery" refers to an image or a video containing modified identity, expressions or attribute(s) with a learning-based approach, distinguished with 1) a so-called "CheapFakes" [48] that are created with off-the-shelf softwares without learnable components and 2) "DeepFakes" that only refer to manipulations with swapped identities [18].

(2) Various Forgery Approaches. There are at most 8 forgery approaches in all current datasets, while ForgeryNet is manipulated by 15 approaches, including face transfer, face swap, face reenactment and face editing. We choose approaches that span a variety of learning-based models, including encoder-decoder structure, generative adversarial network, graphics formation and RNN/LSTM (Fig. 4).

(3) Diverse Re-rendering Process. In the process of transmission and re-rendering, media data (image/video) always undergo compression, blurring and other operations, which may smooth the traces of forgery and bring more challenge for forgery detection. The ForgeryNet dataset posts 36 perturbations, such as optical distortion, multiplicative noise, random compression, blur, and *etc*. As shown in Fig. 1(c), circle sizes refer to the number of forgery approaches with re-rendering process operations.

(4) Rich Annotations and Comprehensive Tasks. According to the real application scenario, we propose four tasks, as shown in Fig. 1(b): 1) Image Forgery Classification, distinguishes whether an image is forgery or not and meanwhile tells its forgery type (*i.e*. manipulation approaches). We provide three types of annotations including two-way, three-way and n-way classification. Both intra- and cross-forgery evaluations are set on three-way and n-way settings. 2) Spatial Forgery Localization, localizes manipulated areas of forgery images. Due to the fact that a forgery image may contain multiple faces and can be manipulated entirely or in part, it is more substantial to segment modified pixels in addition to only telling that it is forged. 3) Video Forgery Classification, similar to image-level classification, contains three types of annotations. Note that different from existing forgery video datasets, we construct our video dataset with untrimmed videos, each of which has part of the frames manipulated, considering the fact that forgery videos in real world are often manipulated on a certain subject and some key frames. 4) Temporal Forgery Localization, localizes the temporal segments which are manipulated. This is a new task for forgery analysis. Together with Video Forgery Classification and Spatial Forgery Localization, it provides comprehensive spatio-temporal forgery annotations.

2. Related Works

Due to the urgency in detecting face manipulation, many efforts have been devoted to creating face forgery detection datasets. Previous datasets can be grouped down into three generations. Their statistical information is listed in Tab. 1.

The first generation consists of datasets such as DF-TIMIT [36], UADFV [60], SwapMe and FaceSwap [64]. DF-TIMIT manually selects 16 pairs of appearance-similar people from the publicly available VidTIMIT database, and generates 640 videos with faces swapped. UADFV contains 98 videos, *i.e*. 49 real videos from YouTube and 49 fake ones generated by FakeAPP [3]. SwapMe and FaceSwap

Table 1: Comparison of various face forgery datasets. ForgeryNet surpasses any other dataset both in scale and diversity. It provides both video- and image-level data. The forgery data are constructed by 15 manipulation approaches within 4 categories. We also employ 36 types of perturbations from 4 kinds of distortions for post-processing.

Dataset	Video Clips		Still images		Approaches	Subjects	Uniq. Perturb.	Mix Perturb.	Annotations
	Real	Fake	Real	Fake					
UADFV [60]	49	49	241	252	1	49	-	×	591
DF-TIMIT [36]	320	640	-	-	2	43	-	×	1,600
Deep Fake Detection [4]	363	3,068	-	-	5	28	-	×	3,431
Celeb-DF [39]	590	5,639	-	-	1	59	-	×	6,229
SwapMe and FaceSwap [64]	-	-	4,600	2,010	2	-	-	×	6,610
DFFD [14]	1,000	3,000	58,703	240,336	7	-	-	×	8,000
FaceForensics++ [52]	1,000	5,000	-	-	5	-	2	×	11,000
DeeperForensics-1.0 [33]	50,000	10,000	-	-	1	100	7	✓	60,000
DFDC [18]	23,564	104,500	-	-	8	960	19	×	128,064
ForgeryNet (Ours)	**99,630**	**121,617**	**1,438,201**	**1,457,861**	**15**	**5400+**	**36**	✓	**9,393,574**

Figure 2: Representative examples of original data collected from four face datasets respectively.

choose two face swapping Apps [1,2] to create 2010 forgery images in total on 1005 original real images.

The second generation includes Google DeepFake Detection dataset [4] with 3,068 forgery videos by five publicly available manipulation approaches, and Celeb-DF [39] containing 590 YouTube real videos mostly from celebrities and 5,639 manipulated video clips. FaceForensics++ [52] consists of 4000 fake videos manipulated by four approaches (*i.e*. DeepFakes, Face2Face, FaceSwap and NeuralTextures), and 1000 real videos from YouTube. The data scale and quality of the second generation have been improved. However, these datasets still lack diversity in forgery approaches and task annotations, and are not well-suited for challenges encountered in real world.

The third generation datasets are the most recent face forgery datasets, *i.e*. DeeperForensics-1.0 [33], DFDC [18], and DFFD [14] which contains tens of thousands of videos and tens of millions of frames. DeeperForensics-1.0 consists of 60,000 videos for real-world face forgery detection. DFDC contains over 100,000 clips sourced from 960 paid actors, produced with several face replacement forgery approaches including learnable and non-learnable approaches. In a practical application, in addition to classification, it is necessary to locate the manipulated areas or segments in an image or an untrimmed video. A few datasets have taken these tasks into consideration. DFFD provides annotations of spatial forgery at the first time, yet it only presents binary masks without manipulation density.

3. ForgeryNet Construction

Most of existing public face forgery datasets [4, 14, 18, 33, 36, 36, 39, 52, 60, 64] contain only single or no more than

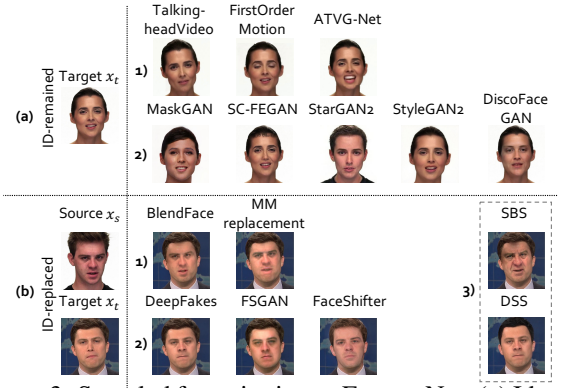

Figure 3: Sampled forgeries in our ForgeryNet. (a) Identity-remained forgery approaches: 1) *Face reenactment*, 2) *Face editing*. (b) Identity-replaced forgery approaches: 1) *Face transfer*, 2) *Face swap*, 3) *Face stacked manipulation*.

10 specific manipulation approaches, and even the largest one [18] only operates 8 manipulations with 19 perturbations on 960 subjects. Moreover, these datasets take forgery analysis solely as a classification task. On the contrary, our proposed ForgeryNet dataset provides 15 manipulation approaches with more than 36 mix-perturbations on over $5,400$[2] subjects, and defines four tasks (*i.e*. image and video classification, spatial and temporal localization) with a total of 9.4M annotations. Our whole dataset consists of two subsets: *Image-forgery* set provides over 2.9M still images and *Video-forgery* set has more than 220k video clips. These two subsets have their real data respectively randomly selected from the original data, and 15 forgery approaches are applied to image-forgery construction while 8 of them also generate the video-forgery data[3]. We compare our ForgeryNet with other publicly available datasets in Tab. 1. Over all the comparison items listed in the table, our dataset surpasses the rest both in scale and diversity.

[2]Some original datasets do not provide the identity annotation.

[3]There are 7 forgery approaches that are only suitable for generating images.

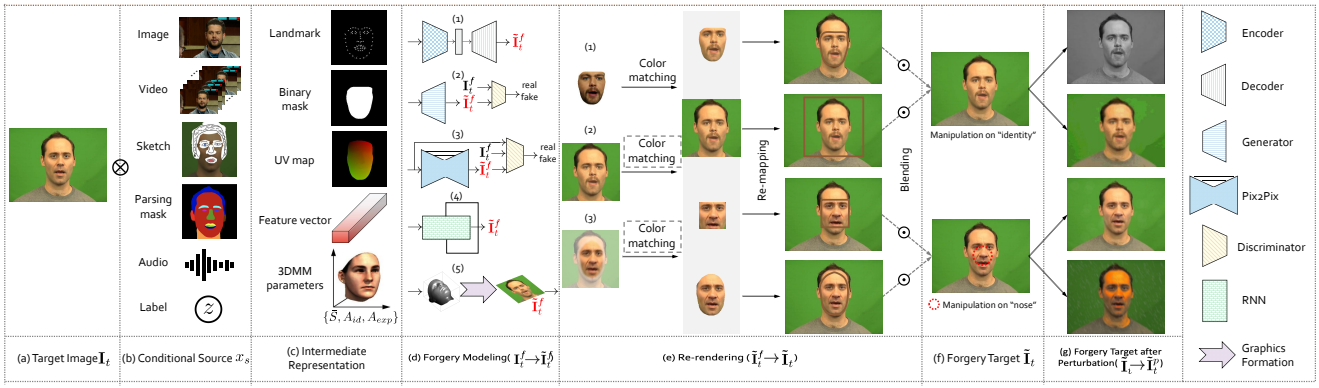

Figure 4: Pipeline of face forgery approaches. (a)-(c) Representation preparation: target image $\mathbf{I}_t$, conditional source x_s and their intermediate representations. (d) Forgery models produce a forged target face $\tilde{\mathbf{I}}_\mathbf{t}^\mathbf{f}$ by processing the representations. (e)-(f) Re-render $\tilde{\mathbf{I}}_\mathbf{t}^\mathbf{f}$ to full image $\mathbf{I}_t$ and get the forgery image $\tilde{\mathbf{I}}_\mathbf{t}$. (g) Apply perturbations to $\tilde{\mathbf{I}}_\mathbf{t}$ to obtain final forgery data.

3.1. Original Data Collection

Source of Original Data. Four face datasets, CREMA-D [7], RAVDESS [45], VoxCeleb2 [13] and AVSpeech [20], are chosen as the original data to boost the diversity in dimensions of face identity, angle, expression, scenarios *etc*.

Note that CREMA-D is made available under the Open Database License, while others are released under a Creative Commons Attribution License. The resolutions of these original data range from 240p to 1080p, and face yaw angles ranging from -90 to 90 degrees are all covered. Representative examples are shown in Fig. 2.

Preprocess Original Data. For further manipulation, we crop original videos into a controllable set of source videos with reasonable lengths. Then we detect and select faces for manipulation and obtain their face attribute labels.

3.2. Forgery Approach

To guarantee the diversity of forgery approaches in the proposed ForgeryNet, we introduce 15 face forgery approaches[4] [9, 11, 17, 23, 34, 35, 37, 38, 47, 49, 56]. They are selected according to perspectives of modeling types, conditional sources, forgery effects and functions. We denote x_t as the *target* subject to be manipulated while the *source* x_s is regarded as the conditional media driving the *target* to change either identity or attributes, or even both.

3.2.1 Forgery Category

According to the visual effects of facial manipulation, we divide the forgery approaches into two categories, *i.e*. *Identity-remained* and *Identity-replaced*. Sampled forgeries in Fig. 3 illustrate these categories and their sub-types.

Identity-remained Forgery Approach in Fig. 3(a) remains the identity of x_t and the identity-agnostic content like expression, mouth, hair and pose of x_t are changed, driven by x_s. We adopt eight approaches and divide them into two sub-types: 1) *Face reenactment* on $x_t(i, a)$ preserves its identity but has its *intrinsic* attributes like pose, mouth and expression manipulated by conditional source x_s and forms $x_t(i, \tilde{a}^s)$, where i refers to identity and a denotes attribute(s). Alternatively, with 2) *Face editing* on $x_t(i, a)$ has its *external* attributes altered, such as facial hair, age, gender and ethnicity, to obtain $x_t(i, \hat{a}^s)$. We also include multiple attribute manipulation with two editing approaches, *e.g*. both hair and eyebrow are manipulated as shown with the first example in Fig. 3(a-2).

Identity-replaced Forgery Approach in Fig. 3(b) replaces the content of x_t with that of x_s preserving the identity of s. Seven approaches are divided into three sub-types as follows. 1) *Face transfer* transfers both identity-aware and identity-agnostic content (*e.g*. expression and pose) from x_s to x_t, resulting in $x_t(\tilde{i}^s, \tilde{a}^s)$. 2) *Face swap* which produces $x_t(\tilde{i}^s, a)$ only swaps identity from the source x_s to the target x_t, and the identity-agnostic content a are preserved. 3) *Face stacked manipulation* refers to a combination of both *Identity-remained* and *Identity-replaced* approaches. We propose two assembles[5], *i.e*. $\langle$*editing* $\rightarrow$ *transfer*$\rangle$ and $\langle$*swap* $\rightarrow$ *editing*$\rangle$, where the former one transfers both the identity and attributes of the manipulated $x_s(i, \hat{a})$ to the target x_t to obtain $x_t(\tilde{i}^s, \tilde{\hat{a}}^s)$ and the latter alters the external attributes of the swapped target $x_t(\tilde{i}^s, a)$ to get $x_t(\tilde{i}^s, \hat{a}^s)$.

3.2.2 Forgery Pipeline

Although there are a wild variety of architectures designed for the aforementioned approaches, most are created using variations or combinations of generative networks, encoder-decoder networks or graphics formation. We briefly summarize the forgery pipeline in Fig. 4.

The target is always an image marked as $\mathbf{I}_t$, while there are various conditional source formats x_s, including image,

[4]Detailed description of the forgery approaches is provided in the appendix.

[5]StarGAN2-BlendFace-Stack (SBS), DeepFakes-StarGAN2-Stack (DSS)

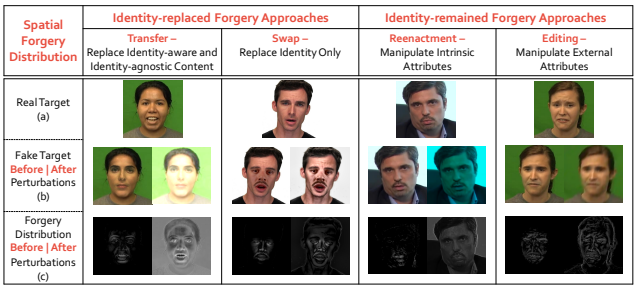

Figure 5: Annotations for Spatial Forgery Localization in ForgeryNet. Examples of (a) real image, (b) forgery image, (c) corresponding spatial annotations.

image sequence, sketch map, parsing mask, audio, label, or even noise. We first detect the *target* face $\mathbf{I}_t^f$, crop and align it, and then transform both the *target* face as well as *source* data to intermediate representations such as UV map, feature bank, 3DMM parameters and *etc*.

Forgery Modeling. These representations are forwarded to the forgery models to obtain a forged target face $\tilde{\mathbf{I}}_t^f$. We include five architecture variants as, 1) *Encoder-Decoder* [5], 2) *Vanilla GAN* [55], 3) *Pix2Pix* [38], 4) *RNN/LSTM* [9], and 5) *Graphics Formation* [19].

Re-rendering Process. To acquire the full forged target, the forged target face $\tilde{\mathbf{I}}_t^f$ is re-rendered back to the target full image $\mathbf{I}_t$ to obtain $\tilde{\mathbf{I}}_t$. In particular, according to different forgery procedures, 1) $\tilde{\mathbf{I}}_t^f$ can be a *face mask*, shown in Fig. 4(e-1), which contains the area from the eyebrows to the face chin. 2) $\tilde{\mathbf{I}}_t^f$ can also be a *face bounding-box*, illustrated in Fig. 4(e-2,3), which keeps the same bounding box as the original target face.

Perturbation. To better reflect real-world data distribution, we apply 36 types of perturbations to the forgery data $\tilde{\mathbf{I}}_t$. We follow common practices in visual quality assessment [54] with distortions of compression, transmission, capture, color, *etc*.

3.3. ForgeryNet Annotation

In contrast to most previous datasets, our ForgeryNet is annotated comprehensively both in image- and video-level across four tasks.

Image Forgery Classification. According to the forgery definition in Sec. 3.2.1, given a forgery image, we provide three types of forgery labels, *i.e*. labels for two-way (real / fake), three-way (real / fake with identity-replaced forgery approaches / fake with identity-remained forgery approaches), and n-way ($n = 16$, real and 15 respective forgery approaches) classification tasks respectively. These annotations make it possible to explore the correlation between different forgery meta-types or approaches.

Spatial Forgery Localization. As shown in Fig. 5, we take the forgery image $\tilde{\mathbf{I}}_t$ and the corresponding real image $\mathbf{I}_t$ to calculate their difference to obtain a *forgery distribution* $\tilde{\mathbf{I}}_t^d$. In this paper, we define the *Spatial Forgery Localization* task as "*localizing the face area manipulated by deep forgery approaches*", and thus the forgery distribution before perturbation $\tilde{\mathbf{I}}_t^d$ is taken as the ground-truth annotation.

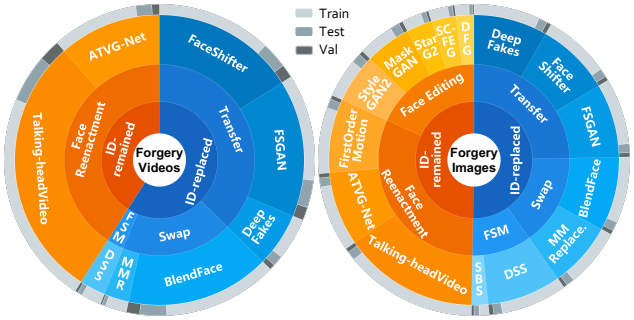

Figure 6: Illustration of image- and video-level sets. From the inside to the outside are categories of *Identity-remained* and *Identity-replaced*, corresponding sub-types, specific forgery approaches and the situation of data split.

Video Forgery Classification & Temporal Forgery Localization. Note that in contrast to all the existing datasets, we construct our video forgery dataset with untrimmed forgery videos $\tilde{\mathbf{V}}_t'$, each of which splices real and manipulated frames together. Same as image-forgery, *Video Forgery Classification* also contains three types of class annotations. We also provide the annotations on locations of manipulated segments in the untrimmed forgery video and propose a new task, *i.e*. *Temporal Forgery Localization*, to localize these forged segments.

4. ForgeryNet Settings

On ForgeryNet, we set up two benchmarks, image and video, with a series of tasks for face forgery analysis.

Dataset Preparation. Both image- and video-level sets are split into training, validation and test subsets with a ratio close to 7:1:2. Forgery data distributions and catagories of the two sets are shown in Fig. 6. Forgery data in each subset have identities matched with the corresponding real subset. The ratio of real to fake in each subset is close to 1:1.

4.1. Image Benchmark Settings

4.1.1 Image Forgery Classification

In order to foster further researches on face forgery classification, we carefully design two protocols to evaluate forensics methods in this area.

Protocol 1: Intra-forgery Evaluation. In intra-forgery evaluation, all the real and fake data in the training set are used to train models, and the validation set is used for evaluation. This protocol has three variants, according to the definition in Sec. 3.3, *i.e*. two-/three-/n-way classification.

Protocol 2: Cross-forgery Evaluation. To further evaluate the generalization ability of training with our data, we

Table 2: **Image Forgery Classification (Protocol 1):** binary classification. We report accuracy and AUC scores of the compared forensics methods.

Method	Param.	Acc	AUC
MobileNetV3 Small [29]	1.7M	76.24	85.51
MobileNetV3 Large [29]	4.2M	78.30	87.56
EfficientNet-B0 [58]	4.0M	79.86	89.31
ResNet-18 [28]	11.2M	78.31	87.75
Xception [12]	20.8M	80.78	90.12
ResNeSt-101 [62]	46.2M	82.06	91.02
SAN19-patchwise [63]	18.5M	80.08	89.38
ELA-Xception [27]	20.8M	73.77	82.69
SNRFilters-Xception [10]	20.8M	81.09	90.52
GramNet [44]	22.1M	80.89	90.20
F^3-Net [50]	57.3M	80.86	90.15

conduct cross-forgery evaluation by training the evaluated forensics method with one certain type of manipulation and testing it with others. The manipulation type can either be general (*e.g. identity-replaced*), or specific (*e.g. ATVG-Net*). Note that this protocol only involves binary classification.

Metrics. For binary classification tasks, we evaluate with Accuracy (Acc) and the Area under ROC curve (AUC). For three- and n-way class settings, we use Accuracy (Acc) and mean Average Precision (mAP) as evaluation metrics.

4.1.2 Spatial Forgery Localization

Compared with classification tasks, spatial forgery localization aims to specify manipulated regions. Images along with forgery masks are used to train the localization model.

Metrics. We utilize three metrics for evaluation: two variants of Intersection over Union (IoU) and L1 distance.

4.2. Video Benchmark Settings

Video Forgery Classification. Evaluation protocols for video forgery classification are generally similar to the ones designed for the image set, except that n=9 for n-class setting. Metrics are the same as those for image classification.

Temporal Forgery Localization. For each video, forensics methods to be evaluated are expected to provide temporal boundaries of forgery segments and the corresponding confidence values. We follow metrics used in ActivityNet [24] evaluation, and employ Interpolated Average Precision (AP) as well as Average Recall@K (AR@K) for evaluating predicted segments with respect to the groundtruth ones.

5. Image Forgery Analysis Benchmark

5.1. Image Forgery Classification

Protocol 1: Intra-forgery Evaluation. For comprehensive evaluation, we provide results of two-way class classification with several representative models of different sizes. Considering the trade-off between performance and efficiency, we use Xception [12] as the baseline model. ELA-Xception [27] and SNRFilters-Xception [10] are two variants of Xception. Smaller models include MobileNetV3 [29], EfficientNet-B0 [58] and ResNet-18 [28]. We select ResNeSt-101 [62] as the large model. We also experiment with recent state-of-the-art methods for face forgery detection, *i.e.* F^3-Net [50] and GramNet [44], as well as a fully-attentional network SAN19 [63].

All experiments are conducted on face images cropped with face bounding boxes enlarged by $1.3\times$. During training, we use several types of data augmentation to mimic distortions caused by compression and packet loss during transmission, so as to improve the generalization of developed models.

As presented in Tab. 2, we list binary classification metrics of all aforementioned forensics methods. We also show the corresponding ROC curves of these methods in Fig. 7(a). For three-way and 16-way classification experiments, as shown in Tab. 3, Acc scores show that classification becomes more difficult when the number of categories increases, yet the mAP metric indicates that the discrimination ability becomes higher instead. Moreover, after mapping back to binary classification, we can also observe slight performance boosts on F^3-Net compared to training results with only binary labels. This suggests that more auxiliary information potentially makes the forensics model more discriminative.

Table 3: **Image Forgery Classification (Protocol 1):** multi-class settings and their mappings to binary classification. We report the accuracy, mAP and AUC scores.

	3-way class		3→2-way class	
	Acc.	mAP	Acc.	AUC
Xception	73.00	89.90	80.17	89.92
GramNet	73.30	90.00	80.75	90.13
F^3-Net	74.45	90.41	81.75	90.63
	16-way class		16→2-way class	
	Acc.	mAP	Acc.	AUC
Xception	58.81	93.16	81.00	90.53
GramNet	56.77	92.27	80.83	90.25
F^3-Net	59.82	92.98	81.88	90.91

Table 4: **Image Forgery Classification (Protocol 2):** binary classification. We report the accuracy and AUC scores. Forensics methods trained with ID-replaced forgery approaches have significant performance drops when tested on unseen ID-remained forgery approaches, and *vice versa*.

		ID-replaced		ID-remained	
		Acc.	AUC	Acc.	AUC
Xception	ID-replaced	84.13	92.80	64.62	74.86
	ID-remained	67.28	75.83	81.17	90.71
GramNet	ID-replaced	82.82	92.54	62.72	74.28
	ID-remained	67.50	76.19	80.60	90.28
F^3-Net	ID-replaced	83.84	92.73	64.33	73.82
	ID-remained	68.44	77.24	81.18	90.29

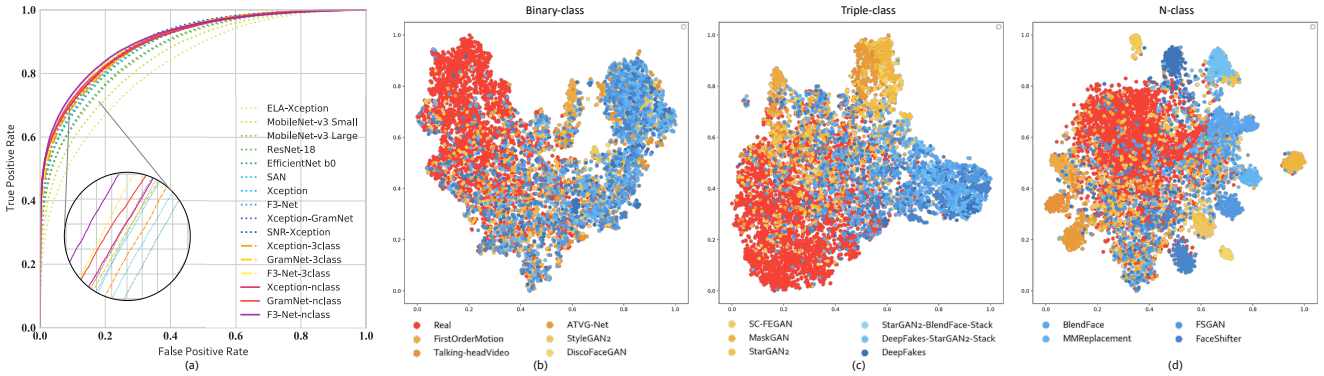

Figure 7: **Image Forgery Classification (Protocol 1):** (a) We show the ROC curves of the compared methods under the setting of binary classification. (b)-(d) t-SNE feature visualization of the data manipulated by different forgery approaches, trained with binary, three-way and n-way classification respectively.

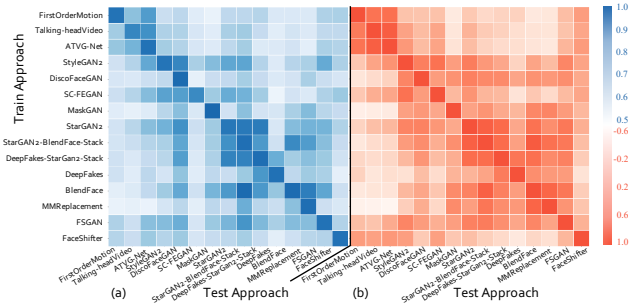

Figure 8: **Image Forgery Classification (Protocol 2):** (a) AUC score map, and (b) correlation map according to the AUC scores. X-axis denotes the tested forgery approach and Y-axis denotes the forgery approach for training.

Protocol 2: Cross-forgery Evaluation. For this protocol, we show the generalization ability of forensics methods across forgery approaches. Tab. 4 lists the results of models trained on *ID-replaced* but evaluated on *ID-remained*, and *vice versa*. The more exhaustive cross-forgery setting with 15 specific forgery approaches is also evaluated and shown in Fig. 8. We observe from these results that intra-forgery testing naturally performs the best. From Fig. 8(a), we can also see that training on *ATVG-Net, StyleGAN2* or *BlendFace* gives the best generalization performance on average. On the other hand, *DiscoFaceGAN* is the most generalizable forgery approach, while *SC-FEGAN* is the most difficult approach to generalize to. There is another interesting finding that forgery approaches with stronger similarity tend to induce better cross-forgery performance. For example, *DiscoFaceGAN* is a *StyleGAN*-based approach, thus training on the latter approach produces favorable results on the former. Similarly, *StarGAN2* and the two face stack manipulations which both involve *StarGAN2* generalize well to each other. In addition, as shown in Fig. 8(b), forgery approaches belonging to the same meta-category usually have higher correlations mutually. For example, for meta-category *Face reenactment*, if a forensics method can obtain good performance on *ATVG-Net*, it may also work for *FirstOrderMotion* and *Talking-headVideo*.

Table 5: **Spatial Forgery Localization.** We compare results with three metrics, *i.e.*, IOU, IOU_{diff} and L1 distance.

Method	IoU		IoU_{diff}			Loss_{l1}
	0.1	0.2	0.01	0.05	0.1	
Xception+Reg.	89.55	93.70	67.57	83.25	89.22	0.0131
Xeption+Unet [51]	95.99	98.76	79.71	92.70	97.13	0.0134
HRNet [59]	96.27	98.78	88.73	92.99	96.27	0.0114

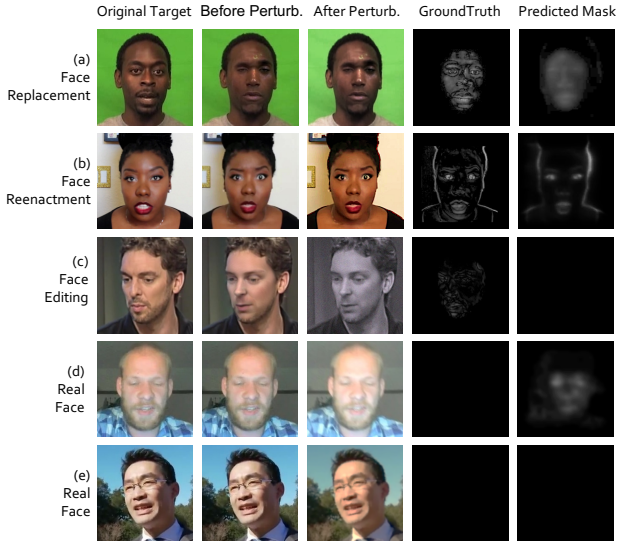

Figure 9: **Spatial Forgery Localization.** Examples of predicted manipulation masks by HRNet.

5.2. Spatial Forgery Localization

We evaluate pixel regression and other two segmentation methods for the spatial localization task. UNet [51] is a popular segmentation architecture, which has been widely used. For comparison, we also adopt HRNet [59] because of its superior performance on other datasets.

In Tab. 5, HRNet outperforms other methods. Especially in terms of IoU_{diff} with threshold 0.01, HRNet surpasses

Table 6: **Video Forgery Classification (Protocol 1):** binary classificaiton. We report accuracy and AUC scores under two crop strategies. Video-level classification has better results than the image-level setting.

Method	Parameters	Single-crop		Multi-crop	
		Acc	AUC	Acc	AUC
X3D-M [21]	2.9M	87.93	93.75	88.97	96.99
Slow-only [22]	31.6M	86.76	92.64	87.37	95.96
TSM [40]	23.5M	88.04	93.05	89.11	96.25
SlowFast [22]	33.6M	88.78	93.88	89.92	97.28

Table 7: **Video Forgery Classification (Protocol 1):** multi-class settings and their mappings to binary classification. We report the accuracy, mAP and AUC scores.

Method	3-way class		3→2-way class	
	Acc.	mAP	Acc.	AUC
X3D-M [21]	84.00	94.55	87.69	93.78
SlowFast [22]	85.73	94.89	89.11	94.37
	9-way class		9→2-way class	
	Acc.	mAP	Acc.	AUC
X3D-M [21]	76.91	95.06	87.51	93.81
SlowFast [22]	80.86	95.92	89.45	94.25

other methods by more than 10%. We also present predicted manipulation maps for several test samples. In Fig. 9(c), the slight beard change is hard to detect, while in Fig. 9(d), a real image is misjudged as manipulated.

6. Video Forgery Analysis Benchmark

6.1. Video Forgery Classification

In this section, we select several typical video backbones of different sizes: X3D-M [21], Slow-only R-50 [22], TSM [40], and SlowFast R-50 [22]. We sample 16 frames with temporal stride 4 as input to all models.

Binary classfication results of video-level forensics methods are listed in Tab. 6. Compared to image-level evaluation, video-level Acc and AUC are generally higher. SlowFast [22] obtains the best performance on video classification, while X3D-M [21], with only a very small number of parameters, also gives satisfying results. We select these two as representatives of large and small models respectively in subsequent experiments, as displayed in Tab. 7 and Tab. 8. Cross-forgery evaluation results are worse than their image counterparts, suggesting harder generalization with temporal information.

6.2. Temporal Forgery Localization

We experiment with both frame-based and video-based models for temporal localization. For frame-based model, after binarizing frame predictions with a fixed threshold (0.25), we select consecutive fake sequences, with different tolerance levels for real frames in the middle, as final proposals. The confidence of a proposal is simply the average of the original frame scores. We adopt Boundary-Sensitive Network (BSN) [42] and Boundary-Matching Network (BMN) [41] on top of X3D-M and SlowFast features as the video-based models.

Table 8: **Video Forgery Classification (Protocol 2):** binary classification. Forensics methods trained with ID-replaced forgery approaches have substantial performance drops (even more significant than their image-level counterparts) when tested on unseen ID-remained forgery approaches, and *vice versa*.

		ID-replaced		ID-remained	
		Acc.	AUC	Acc.	AUC
X3D-M	ID-replaced	87.92	92.91	55.25	65.59
	ID-remained	55.93	62.87	88.85	95.40
SlowFast	ID-replaced	88.26	92.88	52.64	64.83
	ID-remained	52.70	61.50	87.96	95.47

Table 9: **Temporal Forgery Localization.** We show AP, AR and mAP scores of all compared methods.

	AR		AP			avg.
	2	5	0.5	0.75	0.9	AP
Xception [12]	25.83	73.95	68.29	62.84	58.30	62.83
X3D-M+BSN [42]	81.33	86.88	80.46	77.24	55.09	70.29
X3D-M+BMN [41]	88.44	91.99	90.65	88.12	74.95	83.47
SlowFast+BSN [42]	83.63	88.78	82.25	80.11	60.66	73.42
SlowFast+BMN [41]	90.64	93.49	92.76	91.00	80.02	86.85

Tab. 9 compares these methods on the validation set. In particular, video-based methods perform significantly better than the frame-based method, demonstrating the importance of applying a boundary-aware network. Additionally, BMN outperforms BSN with large margins, and achieves ∼87 average AP. This is of great significance since it shows our model is capable of effectively locating manipulated media in a large video database. We hope our results can inspire more future works on forgery localization.

7. Conclusion

In this paper, we present ForgeryNet, a new mega-scale benchmark for both image- and video-level face forgery analysis. Compared with existing datasets for face forgery, ForgeryNet possesses more variety and is more comprehensive in terms of wild sources, various manipulation approaches, diverse re-rendering process and richness of annotations. We further introduce four possible applications with ForgeryNet: image and video classification, spatial and temporal localization. The results obtained in these tasks help us better understand facial forgery towards real-world scenarios. For future works, we welcome interested researchers to contribute more novel facial forgery approaches. More forgery analysis can also be studied on our dataset to improve the defense capabilities.

Acknowledgments This work is supported by key research and development program of Guangdong Province, China, under grant 2019B010154003, as well as NTU NAP and A*STAR through the Industry Alignment Fund - Industry Collaboration Projects Grant, the National Natural Science Foundation of China under Grant No. 61906012.

References

[1] Faceswap. https://github.com/MarekKowalski/FaceSwap/. 3

[2] Swapme. https://itunes.apple.com/us/app/swapme-by-faciometrics/. 3

[3] Fakeapp. https://www.fakeapp.com/, 2018. 2

[4] Google ai blog. contributing data to deepfake detection research. https://ai.googleblog.com/2019/09/contributing-data-to-deepfake-detection.html, 2019. 3

[5] faceswap. https://github.com/deepfakes/faceswap, 2020. 5, 11

[6] Hassan Foroosh Ankit Sharma. Slim-cnn: A light-weight cnn for face attribute prediction. *arXiv preprint arXiv:1907.02157*, 2019. 11

[7] Houwei Cao, David G Cooper, Michael K Keutmann, Ruben C Gur, Ani Nenkova, and Ragini Verma. Crema-d: Crowd-sourced emotional multimodal actors dataset. *IEEE Trans. Affect. Comput.*, 5(4):377–390, 2014. 2, 4, 10

[8] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017. 16

[9] Lele Chen, Ross K Maddox, Zhiyao Duan, and Chenliang Xu. Hierarchical cross-modal talking face generation with dynamic pixel-wise loss. In *CVPR*, 2019. 4, 5, 11, 12

[10] Mo Chen, Vahid Sedighi, Mehdi Boroumand, and Jessica Fridrich. Jpeg-phase-aware convolutional neural network for steganalysis of jpeg images. In *the 5th ACM Workshop*, 2017. 6, 14

[11] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *CVPR*, pages 8188–8197, 2020. 4, 11, 12

[12] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, 2017. 6, 8, 14

[13] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition. *arXiv preprint arXiv:1806.05622*, 2018. 2, 4, 10, 11

[14] Hao Dang, Feng Liu, Joel Stehouwer, Xiaoming Liu, and Anil K Jain. On the detection of digital face manipulation. In *CVPR*, pages 5781–5790, 2020. 3

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009. 14

[16] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *CVPR*, 2020. 11

[17] Yu Deng, Jiaolong Yang, Dong Chen, Fang Wen, and Xin Tong. Disentangled and controllable face image generation via 3d imitative-contrastive learning. In *CVPR*, 2020. 12

[18] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge dataset. *arXiv preprint arXiv:2006.07397*, 2020. 2, 3, 15

[19] Bernhard Egger, William AP Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, et al. 3d morphable face models—past, present, and future. *TOG*, 39(5):1–38, 2020. 5, 11

[20] A. Ephrat, I. Mosseri, O. Lang, T. Dekel, K Wilson, A. Hassidim, W. T. Freeman, and M. Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. *arXiv preprint arXiv:1804.03619*, 2018. 2, 4, 10, 11

[21] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *CVPR*, 2020. 8, 15

[22] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019. 8, 15

[23] Ohad Fried, Ayush Tewari, Michael Zollhöfer, Adam Finkelstein, Eli Shechtman, Dan B Goldman, Kyle Genova, Zeyu Jin, Christian Theobalt, and Maneesh Agrawala. Text-based editing of talking-head video. *TOG*, 38(4):1–14, 2019. 2, 12

[24] Bernard Ghanem, Juan Carlos Niebles, Cees Snoek, Fabian Caba Heilbron, Humam Alwassel, Victor Escorcia, Ranjay Krishna, Shyamal Buch, and Cuong Duc Dao. The activitynet large-scale activity recognition challenge 2018 summary. *arXiv preprint arXiv:1808.03766*, 2018. 6, 15

[25] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 14

[26] Luca Guarnera, Oliver Giudice, and Sebastiano Battiato. Deepfake detection by analyzing convolutional traces. In *CVPRW*, 2020. 2

[27] Teddy Surya Gunawan, Siti Amalina Mohammad Hanafiah, Mira Kartiwi, Nanang Ismail, Nor Farahidah Za'bah, and Anis Nurashikin Nordin. Development of photo forensics algorithm by detecting photoshop manipulation using error level analysis. *IJEECS*, 7(1):131–137, 2017. 6, 14

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6, 14

[29] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *ICCV*, 2019. 6, 13

[30] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 13

[31] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018. 14

[32] Nils Hulzebosch, Sarah Ibrahimi, and Marcel Worring. Detecting cnn-generated facial images in real-world scenarios. In *CVPRW*, 2020. 2

[33] Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection. In *CVPR*, 2020. 3, 10, 15

[34] Youngjoo Jo and Jongyoul Park. Sc-fegan: Face editing generative adversarial network with user's sketch and color. In *ICCV*, 2019. 4, 11, 12

[35] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *CVPR*, 2020. 4, 12

[36] Pavel Korshunov and Sébastien Marcel. Deepfakes: a new threat to face recognition? assessment and detection. *arXiv preprint arXiv:1812.08685*, 2018. 2, 3

[37] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *CVPR*, 2020. 4, 11, 12

[38] Lingzhi Li, Jianmin Bao, Hao Yang, Dong Chen, and Fang Wen. Faceshifter: Towards high fidelity and occlusion aware face swapping. *arXiv preprint arXiv:1912.13457*, 2019. 2, 4, 5, 11, 12

[39] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *CVPR*, 2020. 3

[40] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *ICCV*, 2019. 8, 15

[41] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *ICCV*, 2019. 8, 16

[42] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *ECCV*, 2018. 8, 16

[43] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 11

[44] Zhengzhe Liu, Xiaojuan Qi, and Philip HS Torr. Global texture enhancement for fake face detection in the wild. In *CVPR*, 2020. 6, 14

[45] Steven R Livingstone and Frank A Russo. The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english. *PLOS ONE*, 13(5):e0196391, 2018. 2, 4, 10

[46] Francesco Marra, Cristiano Saltori, Giulia Boato, and Luisa Verdoliva. Incremental learning for the detection and classification of gan-generated images. In *WIFS*, 2019. 2

[47] Yuval Nirkin, Yosi Keller, and Tal Hassner. Fsgan: Subject agnostic face swapping and reenactment. In *ICCV*, 2019. 12

[48] Britt Paris and Joan Donovan. Deepfakes and cheap fakes. *United States of America: Data & Society*, 2019. 2

[49] Ivan Petrov, Daiheng Gao, Nikolay Chervoniy, Kunlin Liu, Sugasa Marangonda, Chris Umé, Jian Jiang, Luis RP, Sheng Zhang, Pingyu Wu, et al. Deepfacelab: A simple, flexible and extensible face swapping framework. *arXiv preprint arXiv:2005.05535*, 2020. 2, 12

[50] Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *ECCV*, 2020. 6, 14

[51] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 7, 14

[52] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *CVPR*, pages 1–11, 2019. 2, 3, 10, 15

[53] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, 2018. 14

[54] Muhammad Shahid, Andreas Rossholm, Benny Lövström, and Hans-Jürgen Zepernick. No-reference image and video quality assessment: a classification and review of recent approaches. *EURASIP J IMAGE VIDE*, 2014(1):40, 2014. 5

[55] Yujun Shen, Bolei Zhou, Ping Luo, and Xiaoou Tang. Facefeat-gan: a two-stage approach for identity-preserving face synthesis. *arXiv preprint arXiv:1812.01288*, 2018. 5, 11

[56] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. In *NeurIPS*, 2019. 4, 11, 12

[57] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V Le. Mnasnet: Platform-aware neural architecture search for mobile. In *CVPR*, 2019. 14

[58] Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 2019. 6, 14

[59] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *TPAMI*, 2020. 7, 14

[60] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses. In *ICASSP*, 2019. 2, 3

[61] Ning Yu, Larry Davis, and Mario Fritz. Attributing fake images to gans: Analyzing fingerprints in generated images. *arXiv preprint arXiv:1811.08180*, 2018. 2

[62] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. *arXiv preprint arXiv:2004.08955*, 2020. 6, 14

[63] Hengshuang Zhao, Jiaya Jia, and Vladlen Koltun. Exploring self-attention for image recognition. In *CVPR*, 2020. 6, 14

[64] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Two-stream neural networks for tampered face detection. In *CVPRW*, 2017. 2, 3

Appendix

A. Original Data Collection

In contrast to previous facial forgery datasets [33, 52] which only involve original data taken from certain briefing scenarios or TV shows, we choose four face datasets [7, 13, 20, 45] as the original data with diversified face identities, angles, expressions, actions, *etc*., for the sake of building a wild and diverse forgery dataset.

(1) *CREMA-D* [7] is a dataset of $7,442$ video clips from 48 male and 43 female actors with a variety of ethnicities, ages ranging from 20 to 74, and six different emotions.

(2) *RAVDESS* [45] consists of $7,356$ files including both video footages and sound tracks from 24 professional actors with eight emotions, vocalizing two lexically-matched statements in a neutral North American accent.

(3) *VoxCeleb2* [13] is constructed with over one million YouTube videos with utterances of $6,112$ celebrities.
(4) *AVSpeech* [20] is a dataset of 290k YouTube video clips of $3 \sim 10$ seconds long. Note that the speakers talk with no audio background interference, *i.e*. the only audible sound in the soundtrack of a video belongs to a single visible and speaking person.

B. Original Data Preprocessing

The selected in-the-wild videos vary in length (2 seconds $\sim$ 1 hour), FPS ($20 \sim 30$), semantic annotations, and number of faces appearing in one frame. For further manipulation, we preprocess the original data into a controllable source video set:

(1) *Video-Origin & Image-Origin:* Due to the large amount of videos in VoxCeleb2 and AVSpeech, we respectively pick $43,941$ and $43,584$ videos with length over 6 seconds. The videos are chosen randomly, yet in VoxCeleb2 we guarantee all $6,112$ identities are included in the selected video set. All the selected videos from these two datasets are then truncated into $6 \sim 10$ seconds to enrich length variations, while those from CREMA-D and RAVDESS are retained without cropping due to their short duration ($2 \sim 5$ seconds). The images of `image-origin` are extracted from the aforementioned `video-origin` set with 20 FPS.

(2) *Target Face:* We detect faces from images in `image-origin` by RetinaFace [16] for future manipulation. As shown in Fig. 2 in the main paper, in some scenarios, multiple faces co-occur in a single frame, such as "conversation between two or more people" or "crowd gathering". To determine the target face for forgery, we first use a simple IoU (Intersection-over-Union) based tracking to acquire face tubes each with faces of the same person identity. We select the face which appears most frequently in the video, *i.e*. has the longest face tube.

(3) *Attribute Prediction:* To manipulate facial attributes, the deep models require attribute labels as a conditional input. However, data in `video/image-origin` lack attribute labels due to limited annotations (*e.g*. only "emotions" and "age") of the original datasets. To this end, we predict the attribute labels with Slim-CNN [6, 43], a state-of-the-art face attribute classification method.

C. Forgery Approach

To guarantee the diversity of forgery approaches in the proposed ForgeryNet, we introduce 15 face forgery approaches [9, 11, 17, 23, 34, 35, 37, 38, 47, 49, 56], which are shown in the main paper. We conclude five architecture variants as, 1) *Encoder-Decoder* [5] is used to disentangle the identity from identity-agnostic attributes and then modify/swap the encodings of the target before passing them through the decoder. 2) *Vanilla GAN* [55] consists of a generator and a discriminator which work against each other. After training, the discriminator is discarded and the generator is used to generate content. 3) *Pix2Pix* [38] is a popular improvement on GANs which enables translations from one image domain to another. The generator is an encoder-decoder network with skip connections from encoder to decoder which enable the generator to produce high fidelity imagery by bypassing some compression layers when needed. In addition to the above three variants, which are the basic elements for generating a forgery image, some sequential and dynamic-length data (*e.g*. audio and video) are often handled by 4) *RNN/LSTM* [9], and 5) *Graphics Formation* [19]. The latter represents a simulation of the classical image formation process of computer graphics, that is, reconstructing a 3D face model with 3DMM parameters, which are the output of a classical analysis-by-synthesis algorithm, and then rendering the generated 3D face model into a 2D image.

D. Re-rendering Process

(1) For the *face mask* condition shown in Fig. 4 (e-1) in the main paper, we first align the landmarks of $\tilde{\mathbf{I}}_t^f$ and $\mathbf{I}_t^f$ to align their masks $\tilde{\mathbf{I}}_t^m$ and $\mathbf{I}_t^m$, and then calculate an optimal transformation to align $\tilde{\mathbf{I}}_t^f$ back to the $\mathbf{I}_t$. Color matching is then operated on the re-aligned face to make $\tilde{\mathbf{I}}_t^f$ more adaptable to $\mathbf{I}_t^f$[6]. The following step is blending, with the objective of making $\tilde{\mathbf{I}}_t^f$ seamlessly fit the target full image $\mathbf{I}_t$. We corrode and blur the smaller mask between $\tilde{\mathbf{I}}_t^m$ and $\mathbf{I}_t^m$, and perform the Poisson blending along the outer contour of $\tilde{\mathbf{I}}_t^f$ to get the full forgery image $\tilde{\mathbf{I}}_t$.
(2) For the *face bounding-box* condition, an easy way is to directly substitute the bounding-box in the original target image $\mathbf{I}_t^b$ with a forgery one $\tilde{\mathbf{I}}_t^b$, and simply perform the Poisson blending along the edge of the bounding-box as shown in Fig. 4 (e-2) in the main paper. However, some GAN-based approaches always induce some unexpected details outside the face region, especially some background clutters with jittery and blurred information. Meanwhile, some graphic-based approaches cannot infer the texture of non-face regions such as hair. To this end, we first calculate the convex hull of the face area through the face landmarks to obtain the face mask $\tilde{\mathbf{I}}_t^m$, and then turn to the re-rendering solution for the *face mask* condition described above, as is illustrated in Fig. 4 (e-3) in the main paper.

Each frame of a video is re-rendered through the aforementioned steps. However, the obtained re-rendered frame sequence often contains frequent jitters due to misalignment and forgery effect. To generate a realistic and smooth video,

[6] *Identity-remained* forgery do not have this step since it only changes local intrinsic or external attributes. Moreover, some editing even aims at altering colors such as lip or eye color.

Table 10: **Summary of the four types of forgery approaches.** In this table, the input, output, architecture, resolution, modification ability, and whether to retrain in inference of each forgery approach are presented. S/T represents the modality of x_s and x_t. v:=video, i:=image, a:=audio, m:= mask, s:=sketch, l:= noise, S:=single identity, M:=multiple identity

	Method	S/T	CG/GAN	Input	Modification	Resolution	Retraining
Face Reenactment	FirstOrderMotion [56]	v/i	GAN	M/M	pose,expression	256*256	No need
	ATVG-Net [9]	v/i	GAN	M/M	pose,expression	128*128	No need
	Talking-head Video [23]	a/v	CG+GAN	M/S	mouth	256*256	1∼3 portraits
Face Editing	StarGAN2 [11]	i/i	GAN	M/M	attribute transfer	256*256	portraits
	StyleGAN2 [35]	l/i	GAN	M/M	rebuild from latent	1024*1024	portraits
	MaskGAN [37]	m,i/i	GAN	M/M	editing record	512*512	portraits,mask
	SC-FEGAN [34]	s,i/i	GAN	M/M	sketch record	512*512	portraits,sketch
	DiscoFaceGAN [17]	i/i	CG+GAN	M/M	3dmm attributes	1024*1024	portraits
Face Transfer	BlendFace	v/v	CG	M/M	identity, expression	Any	No need
	MMReplacement	i/i	CG	M/M	identity, expression	Any	at least 1 protrait
Face Swap	FSGAN [47]	v/v	GAN	M/M	identity	256*256	No need
	DeepFakes [49]	v/v	GAN	S/S	identity	192*192	2k∼5k portraits
	FaceShifter [38]	i/i	GAN	M/M	identity	256*256	No need

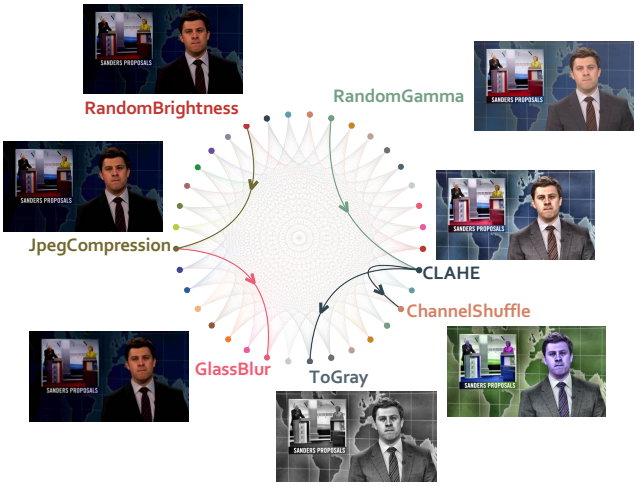

Figure 10: **Perturbations in ForgeryNet.** Different perturbations are marked in different colors. This example shows the effects of one or mixed perturbations. Arrows indicate the mixture order. The image on the left is first added "GlassBlur" followed by "JpegCompression" and at last "RandomBrightness".

we apply slight motion blur as well as compression or super-resolution to the frame sequence.

E. Perturbation

Fig. 10 presents an overview of perturbations. For example, "GlassBlur" and "JpegCompression" can simulate distortion of information in video capture and storage in the real world. Some color distortions such as "RandomBrightness" and "ChannelShuffle" provide diversity in color distributions to adapt to different color renderings of a video.

Mixed perturbations with $2 \sim 4$ distortions are randomly applied to approximately 98% data, while another 1% are added with a single perturbation. The rest 1% are remained unchanged. Each perturbation has $1 \sim 5$ intensity levels. Types and levels of the applied perturbations are all chosen at random, and are applied at the video level, *i.e.* all frames of a video share the same type of perturbation with the same level. Meanwhile, to avoid severe distribution bias, we guarantee each pair of perturbation types co-occurs at least once. The variety of perturbations improves the diversity and realness of ForgeryNet to better imitate the data distribution in real-world scenarios.

F. ForgeryNet Annotation

Image Forgery Classification. The annotations for this task have been elaborated in Sec. 3.3 in the main paper, where we introduce three types of forgery labels, *i.e.* labels for two-way (real / fake), three-way (real / fake with identity-replaced forgery approaches / fake with identity-remained forgery approaches), and n-way ($n = 16$, real and 15 respective forgery approaches) classification tasks respectively.

Spatial Forgery Localization. Due to the fact that forgery images contain various numbers of faces and each face can be manipulated completely or partially, it is more substantial to specify the manipulated area in addition to the classification labels. We convert the forgery image $\tilde{\mathbf{I}}_t$ and the corresponding real image $\mathbf{I}_t$ into two gray-scale images to calculate their pixel-by-pixel absolute differences. We then normalize the difference map within the face area of the real image $\mathbf{I}_t^f$ to obtain a *forgery distribution* $\tilde{\mathbf{I}}_t^d$. As shown in Fig. 5 (a) in the main paper, stronger response suggests the area is manipulated with heavier intensity. Note that

we perform perturbations on the forgery image which cause further modifications in the whole image. The perturbed forgery area distributes all over the whole image rather than merely the face region. In the main paper, compared to Fig. 5 (b) which shows a near-uniform distribution of forgery area both inside and outside the faces, the distribution before perturbation in Fig. 5 (a) shows its advantages in two aspects: 1) the forgery area focuses more on face area, which is consistent with how these deep forgery techniques actually work, and 2) the forgery distribution behaves distinctive among different types of forgery approaches. Take *face reenactment* and *face transfer* as an example, the former has particularly high response on lip and also some medium response around head since the audio- or video-source always drives the lip and pose of the target being manipulated, while the latter replaces both identity-aware and identity-agnostic contents of the target and leads to more even response inside the face. In this paper, we define the *spatial forgery localization* task as "*localizing the face area manipulated by deep forgery approaches*", and thus the forgery distribution before perturbation $\tilde{\mathbf{I}}_t^d$ is taken as the ground-truth annotation.

Video Forgery Classification & Temporal Forgery Localization. As is mentioned in Sec. 3.3 in the main paper, in contrast to all existing datasets, we construct our video forgery dataset with untrimmed forgery videos $\tilde{\mathbf{V}}_t'$, each of which splices real and manipulated segments together. This is based on the consideration that forgery videos in the real world often only involve manipulation on a certain subject at some key frames. Specifically, for each pair of forgery video $\tilde{\mathbf{V}}_t$ and its corresponding real video $\mathbf{V}_t$, we first randomly select $1 \sim 4$ segments from the forgery video $\tilde{\mathbf{V}}_t$, and then fill the rest with the corresponding real segments $\mathbf{V}_t$. Each forgery/real segment in $\tilde{\mathbf{V}}_t'$ has no fewer than 9 frames.

Same as image-forgery, the *Video Forgery Classification* also contains three types of class annotations. We also provide the annotations of each fragment in the untrimmed forgery video and propose a new task, *i.e. Temporal Forgery Localization*, to localize the temporal segments which are manipulated.

G. ForgeryNet Split

We first split the identities of the original videos into two subsets, training and evaluation, roughly according to a proportion of 7:3. This guarantees that any person appearing in a training video does not show up in the evaluation set. Note that the AVSpeech dataset does not provide annotations on person identity, so we have to assume that different videos contain different people, and directly split the videos. The evaluation subset is then further divided into validation and test with an approximate ratio of 1:2, and there may be some identity overlaps between the validation and test subsets. The real data for our image set is sampled from the frames extracted with these original videos according to some fixed proportion. Finally, we apply our 15 forgery approaches to generate manipulated data within each subset respectively, *e.g.* the sources and targets for generating validation forgery data must all come from the validation subset of the original videos.

H. Image Forgery Analysis Benchmark

H.1. Metrics

Image Forgery Classification. We detail calculation methods of the metrics listed in Sec. 4.1.1 in the main paper. For k-way classification ($k = 2,\ 3,\ 16$), we use Accuracy (Acc) balanced over classes, *i.e.* we first calculate k accuracy values from the k classes respectively, and then take the uniform average of them as the final balanced accuracy. We also evaluate the standard Area under ROC curve (AUC) for binary classification. In terms of the other settings with more than two classes, we turn to mean Average Precision (mAP) to measure the discrimination ability of the forensics method. More specifically, the AP of some class i is simply the AUC calculated with class i as the sole positive class and all others being negative. After obtaining k APs, we average them to get mAP. Apart from Acc and mAP, we also compute binary metrics for 3-way or n-way classification, and we sum up probabilities predicted for all forgery categories as the final fake confidence.

Spatial Forgery Localization. As is mentioned in Sec. 4.1.2 in the main paper, we choose three metrics for evaluating predicted maps in our spatial localization task: two variants of Intersection over Union (IoU) and L1 distance. Let N denote the number of pixels, and τ be a predefined threshold.

- IoU $= \frac{1}{N}\sum_{i=1}^{N} |\mathbb{I}[\text{pred}_i \geq \tau] - \mathbb{I}[\text{gt}_i \geq \tau]|$ (*e.g.* $\tau = 0.1$) represents the accuracy over all spatial grids.
- $\text{IoU}_{\text{diff}} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}[|\text{pred}_i - \text{gt}_i| \leq \tau]$ (e.g. $\tau = 0.05$) indicates whether the predicted value of each pixel is close to the groundtruth.
- L1 distance $\text{Loss}_{l1} = \frac{1}{N}\sum_{i=1}^{N} |\text{pred}_i - \text{gt}_i|$ also implies how close is the predicted map to the groundtruth one.

H.2. Models

Image Forgery Classification. There are in total 11 image-level classification methods.

- **MobileNetV3** [29] is an efficient mobile model, combining the following three layers: depthwise separable convolutions from MobileNetV1 [30], the linear bottleneck and inverted residual structure from

MobileNetV2 [53], and lightweight attention modules based on squeeze and excitation from MnasNet [57]. We use both MobileNetV3-Small and MobileNetV3-Large for evaluation.

- **EfficientNet-B0** [58] is the baseline network of the EfficientNet family, which is developed by leveraging a multi-objective neural architecture search based on mobile inverted bottleneck MBConv [53] with squeeze-and-excitation optimization [31] added to it.

- **ResNet-18** [28] is the smallest ResNet architecture with 17 convolutional layers and one fully connected layer for final output.

- **Xception** [12] is a deep convolutional network architecture based on Inception replaced with depthwise separable convolutions. Xception is regarded as our default baseline in further experiments.

- **ResNeSt-101** [62] is a new variant of ResNet. It introduces a modular Split-Attention block that enables attention across different feature-map groups and stacks these blocks ResNet-style to get better performance with similar number of parameters.

- **SAN19-patchwise** [63] takes patchwise self-attention as the basic building block for image recognition. Specifically, we uses SAN19 which roughly corresponds to ResNet-50 to evaluate.

- **ELA-Xception** and **SNRFilters-Xception** differ from Xception in the fact that they do not directly take RGB images as input. More specifically, the input for ELA-Xception is the resulting difference image from Error Level Analysis (ELA) [27]. SNRFilters-Xception, as its name suggests, applies a set of 5×5 high pass kernels [10] to the original input image, and then concatenate the 4 output images along the channel dimension (the number of input channels of the first convolution in Xception is changed to 12 accordingly).

- **Gram-Net** designs Gram Block to leverage global image texture information for fake image detection. The original paper [44] adds Gram Blocks to the ResNet architecture. Yet in our benchmark, we apply them to our baseline model Xception for the sake of fair comparison.

- **$\mathbf{F^3}$-Net** [50] explores frequency information for face forgery detection by taking advantages of two frequency-aware clues: frequency-aware decomposed image components and local frequency statistics. Note that F^3-Net also uses Xception as the backbone network.

Spatial Forgery Localization. We select 3 representative models for spatial localization.

- **Xception+Regression** uses Xception as the backbone network, and adds an extra deconvolution layer after the final feature map to form a direct regression branch which outputs the spatial forgery map.

- **Xception+UNet** [51] supplements a usual contracting network by successive layers where pooling operations are replaced by upsampling operators. A successive convolutional layer can learn to assemble a precise output based on this information. For fair comparison, UNet also uses Xception as its encoder network.

- **HRNet** [59] starts from a high-resolution convolution stream, gradually adds high-to-low resolution convolution streams, and connects the multi-resolution streams in parallel. We use the HRNet-W48 instantiation.

H.3. Implementation Details

Training. For classification methods, we use the default cross-entropy loss for training. As for localization methods, we also add a segmentation loss in addition to the classification loss. There are two choices for the segmentation loss: (1) binary cross entropy loss with soft targets averaged over all spatial locations; (2) MSE loss with respect to groundtruth targets. We select one of these two losses for each localization model based on validation results.

All models use ImageNet [15] for pre-training. We train both classification and localization models end-to-end using synchronous SGD for optimization. The mini-batch size is set to 128. We adopt a multistep learning rate schedule with 100k iterations in total, and the learning rate is decreased by a factor of 0.5 at steps 20k, 40k, 60k, 70k, 80k and 90k. The base learning rate for each model is selected from the set $\{0.01, 0.02, 0.05\}$ according to validation performance. We use linear warm-up [25] from 0.01 during the first 1k iterations. The weight decay is set to 10^{-4} and we apply Nesterov momentum of 0.9. We use face images cropped with provided square bounding boxes (detected boxes enlarged $1.3\times$) for training. For data augmentation, we with 99% probability randomly select one perturbation from some set of perturbation methods, and apply it to the input image. Apart from random perturbation, for a model with input spatial size $S \times S$, we scale the side length to a random value in range $[S, 8S/7]$, and then randomly crop out a $S \times S$ region. Note that for five Xception-based classification models $S = 299$, for three localization models $S = 256$, and for the other six classification models $S = 224$. We also apply random horizontal flip before feeding the input to the model.

Inference. We only perform single-crop inference, and directly scale the input face image to the input spatial size $S \times S$ of the model.

Table 11: **Ablation study on augmentation (image).** We report accuracy and AUC scores of Protocol 1 binary classification on the validation set with three different levels of augmentation.

	weak aug		normal aug		enhanced aug	
	Acc.	AUC	Acc.	AUC	Acc.	AUC
Xception	66.73	74.75	73.70	82.56	80.78	90.12

Table 12: **Cross-dataset experiments.** We report frame-level AUC scores. Each row corresponds to a model trained with one of the datasets. Underlined values are results of models trained and tested on the same dataset, and the bold ones emphasize best cross-dataset performances.

	DF1.0	FF++	DFDC(val)	DFDC(test)	ForgeryNet
FF++ [52]	85.41	99.43	59.77	62.19	63.80
DFDC [18]	79.60	71.34	90.12	93.50	**68.93**
ForgeryNet	**90.09**	**85.06**	**69.68**	**71.08**	90.09

H.4. More Experiments

Ablation Study on Augmentation. We experiment on three different levels of augmentation: weak, normal and enhanced. Weak augmentation does not add random perturbation mentioned in Appendix H.3, while normal and enhanced settings include different numbers of common perturbation methods in the perturbation set for augmentation. Results of Xception trained on these types of data augmentation are shown in Tab. 11. It can be seen that exerting appropriate augmentation to the training set significantly improves the performance of an image forgery classification model.

Cross-dataset Experiments. We provide cross-dataset testing results with our ForgeryNet (image forgery binary classfication only) as well as three public deepfake datasets - FF++ (c23) [52], DFDC [18], and DeeperForensics-1.0 (DF1.0) [33] which are only used for testing. For evaluation, we use (1) test set of FF++ (c23); (2) both validation and test set (only the released half) of DFDC; (3) a subset of DF1.0 which corresponds to the test set of FF++; (4) test set of our image benchmark. For video datasets, we extract frames with temporal stride 30 for frame-level testing. We present the numbers in Tab. 12. ForgeryNet shows the best cross-dataset performances on all other test sets, which indicates the strong generality of our dataset.

I. Video Forgery Analysis Benchmark

I.1. Metrics

Video Forgery Classification. The metrics for this task are the same as those for image classification.

Temporal Forgery Localization. For the temporal localization task, the goal is to generate proposals which have high temporal overlap with the groundtruth (manipulated segments) as well as high recall. We give specifics on our employed metrics for evaluating predicted segments with respect to the groundtruth ones, which are Average Precision at some tIoU threshold (AP@t, *e.g.* $t = 0.5$), average AP, as well as Average Recall@K (AR@K, *e.g.* $K = 5$). Note that these metrics mostly reference ActivityNet [24] evaluation. In details, we choose 10 equally-spaced tIoU threshold values between 0.5 and 0.95 (inclusive) with a step size of 0.05. Under a certain tIoU threshold value t, we may match our predicted segments with the groundtruth according to the condition that tIoU $\geq t$. Recall@K with tIoU threshold t is defined as the proportion of groundtruth which can be matched with some prediction, after preserving only K predicted segments per video on average. AP@t, on the other hand, is the Area under ROC curve computed with predictions and their associated confidence scores, treating the predictions which are matched to some groundtruth segment with tIoU threshold t as positive. Finally, average AP and AR@K are simply the uniform average of APs and Recall@Ks computed at the 10 tIoU thresholds, respectively. Note that both real and fake videos are included in our evaluation, although the real ones do not contain any forgery segment (Recall is not be affected by real videos, but AP is).

I.2. Models

Video Forgery Classification. We choose four typical models for video classification.

- **TSM** [40] inserts Temporal Shift Modules to 2D CNNs to achieve temporal modeling at zero computation and zero parameters. We follow its default setting with ResNet-50 as the backbone network.
- **SlowFast** [22], featuring its two-pathway design with different input temporal strides, is one of the state-of-the-art architectures for action recognition. We choose its R-50 instantiation (without Non-Local blocks), and set the fast-to-slow ratio $\alpha = 4$.
- **Slow-only** is basically the slow pathway of SlowFast, and we also use the R-50 instantiation. Note that with the same number of input frames, Slow-only is actually heavier than SlowFast since the slow branch of the latter only use $1/\alpha$ of the frames.
- **X3D-M** [21] is one member of the X3D family, a series of efficient video networks obtained by progressive expansion along multiple axes. It is able to achieve performances nearly comparable with SlowFast R-50 on common video benchmarks while having much fewer parameters.

Temporal Forgery Localization. As described in Sec. 6.2 in the main paper, we include a frame-based method, where we use Xception as the frame prediction model. The logic of this method can be briefly stated as the following:

1. For a video with T frames, we run the Xception model to get frame-level scores, and then binarize them with threshold 0.25, acquiring a sequence of T binary predictions (real/fake).

2. We enumerate tolerance value in the set $\{1, 3, 5, 7\}$. For a tolerance value t, we inspect the sequence of T predictions, and selects manipulated segments with at least 5 frames satisfying that the length of consecutive real frames in the middle does not exceed t. The confidence score of a segment is simply the average of its frame-level scores.

3. We combine segments predicted with different tolerance levels, and remove duplicates to form the final predictions.

For two video-based methods (BSN [42] and BMN [41]), we use SlowFast and X3D-M for extracting clip features, forming four different "feature+method" pairs. Note that for these feature extraction models, we use fewer input frames for training than their classification counterparts to increase temporal locality. Accordingly, the fast-to-slow ratio α of SlowFast is decreased to 2.

I.3. Implementation Details

Training. For classification methods and feature extraction models for localization, we use the default cross-entropy loss for training. The frame-based localization method directly uses the Xception model trained with the image binary classification task, and does not need any extra training. BSN and BMN have their own training loss functions and procedures which we do not alter.

All models use Kinetics-400 [8] for pre-training. We train them end-to-end using synchronous SGD for optimization. The mini-batch size is set to 64. We adopt a multistep learning rate schedule with 50k iterations in total, and the learning rate is decreased by a factor of 0.5 at steps 20k, 30k, 40k and 45k. The base learning rate is set to 0.02. We use linear warm-up from 10^{-3} during the first 500 iterations. All classification models take 16 frames with a temporal stride of 4 as input, yet the feature extraction models (SlowFast and X3D-M) for BSN and BMN use only continuous 8 frames as input for better temporal sensitivity. We use temporal random crop for training, *i.e.* for a model requiring T frames $\times$ stride τ, we randomly sample a segment of length $T \times \tau$ from the video. In some rare cases where the entire video has less than $T \times \tau$ frames, we use loop padding to fill the rest. The input spatial size is fixed to $S = 224$. Other training details are the same as those for image experiments.

For BSN and BMN, since the feature extraction models take 8 frames as input, we extract features with stride 4. We set the temporal scale parameter to 50, and linearly interpolate the extracted features to the 51 endpoints. We only use fake videos for training video-based localization models. We train TEM and PEM in BSN for 20 epochs each. We train BMN for 9 or 18 epochs according to validation performance. The mini-batch size is set to 128. Other hyper-parameters follow the original settings of BSN and BMN.

Inference. We scale the input to $S \times S$ spatially. On the temporal dimension, we use two settings for classification inference (suppose input temporal sampling is $T \times \tau$): (1) single-crop, or to be more specific, temporally center crop $T \times \tau$ frames; (2) multi-crop, *i.e.* crop multiple segments of length $T \times \tau$ to cover the entire video.

For temporal localization, we only keep top 10 predictions per video in terms of confidence score, and for video-based methods, relevant hyper-parameters are the same as training.

Table 13: **Ablation study on augmentation (video).** We report accuracy and AUC scores of Protocol 1 binary classification on the validation set with three different levels of augmentation.

	weak aug		normal aug		enhanced aug	
	Acc.	AUC	Acc.	AUC	Acc.	AUC
SlowFast	84.39	91.61	87.75	93.22	88.78	93.88

Table 14: **Experiemnts on temporal shuffling.** We report accuracy and AUC scores of Protocol 1 binary classification on the validation set with three different levels of temporal shuffling.

	shuffle 16		shuffle 64		shuffle all	
	Acc.	AUC	Acc.	AUC	Acc.	AUC
SlowFast	88.63	94.11	86.24	93.00	85.04	91.74

I.4. More Experiments

Ablation Study on Augmentation. We conduct similar experiments on augmentation with the same settings as Appendix H.4. As presented in Tab. 13, we observe that our video-level forgery classification method is less affected by augmentation than its image-level counterpart.

Temporal Shuffling Experiments. To verify the effect of continuous temporal information for video forgery classification, we train the SlowFast model with different levels of temporal random shuffling to disrupt temporal continuity: shuffle every 16 frames, shuffle every 64 frames, and shuffle all frames. The results in Tab. 14 indicate that temporal disruptions have considerable, but not very major impact on the performance video classification, implying the video model may have leveraged other sources of information than the continuous temporal flow. An interesting finding is that a weak level of random shuffling (shuffle 16) even slightly boosts the AUC score compared to the setting without shuffling recorded in Tab. 13.

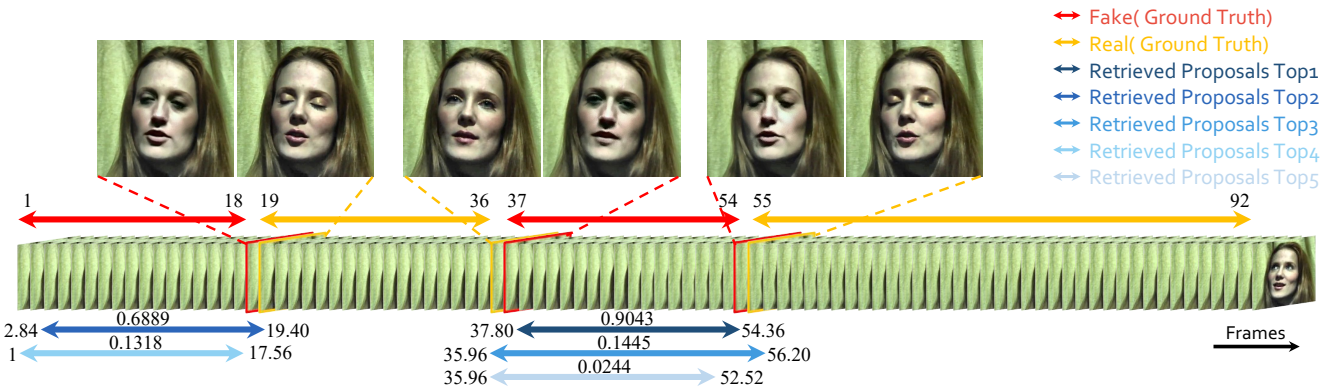

Figure 11: **Example of temporal forgery localization.** We show top-5 predictions of the model SlowFast+BMN. All endpoints of the two manipulated segments are localized with high precision.

I.5. Temporal Localization Analysis

We present an example of temporal forgery localization in Fig. 11. This data sample demonstrates the ability of a boundary-aware model to locate the transitions between real and fake. All endpoints are accurately pointed out by the BMN model. Note that there exist some highly similar predictions, yet are suppressed by a SoftNMS process.