

# On the Detection of Digital Face Manipulation

Joel Stehouwer\* Hao Dang\* Feng Liu\* Xiaoming Liu Anil Jain  
Department of Computer Science and Engineering  
Michigan State University, East Lansing MI 48824

## Abstract

Detecting manipulated facial images and videos is an increasingly important topic in digital media forensics. As advanced synthetic face generation and manipulation methods become available, new types of fake face representations are being created and raise significant concerns for their implications in social media. Hence, it is crucial to detect the manipulated face image and locate manipulated facial regions. Instead of simply using a multi-task learning approach to simultaneously detect manipulated images and predict the manipulated mask (regions), we propose to utilize the attention mechanism to process and improve the feature maps of the classifier model. The learned attention maps highlight the informative regions to further improve the binary classification power, and also visualize the manipulated regions. In addition, to enable our study of manipulated facial images detection and localization, we have collected the first database which contains numerous types of facial forgeries. With this dataset, we perform a thorough analysis of data-driven fake face detection. We demonstrate that the use of an attention mechanism improves manipulated facial region localization and fake detection.

## 1. Introduction

Human faces play an important role in human-human communication and association of side information, e.g., gender, age, with identity. For instance, face recognition systems are increasingly utilized in our daily life for a multitude of applications such as phone unlocking, access control, and payment [42]. However, these advances also entice malicious actors to manipulate face images to launch attacks, aiming to be successfully authenticated as the genuine user. Moreover, manipulation of facial content has become ubiquitous, and raises new concerns especially in social media around the world [36, 34, 35]. Recent advances in deep learning have led to a dramatic increase in the realism of face synthesis and enabled a rapid and far-reaching

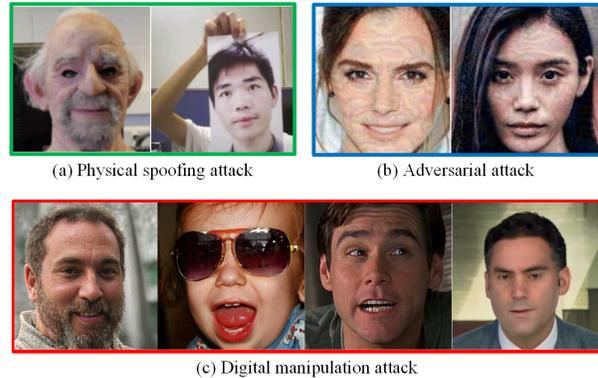


Figure 1. Three types of fake face attacks: (a) physical spoofing attack, (b) adversarial face attack, (c) digital manipulation attack.

dissemination of “fake news” [4]. Therefore, to mitigate the adverse impact caused by fake face attacks, and benefit both *public security and privacy*, it is crucial to develop effective visual forensics solutions against these threats.

As in Fig. 1, there are *three* main types of fake face attacks. i) Physical spoofing attacks can be as simple as printed paper, replaying image/video on a smartphone, or as complicated as a 3D mask [28, 21, 29]. ii) Adversarial face attacks generate high-quality and perceptually imperceptible adversarial examples that can evade automated face matchers [17, 31, 15, 48]. iii) Digital manipulation attacks, made feasible by Variational AutoEncoders (VAEs) [25, 33] and Generative Adversarial Networks (GANs) [16], can generate whole or partially modified photorealistic face images, which are indistinguishable by humans. Among these three types, our research addresses only *digital manipulation attacks*, with the objective of automatically detecting manipulated faces, as well as localizing modified facial regions. We interchangeably use the term “face manipulation detection”, or “face forgery detection” to describe our objective.

Digital facial manipulation methods fall into one of four categories: face expression swap, face identity swap, facial attributes manipulation and entire face synthesis. 3D face reconstruction and animation methods [14, 55] are widely used for facial expression swap, such as *Face2Face* [40].

\*denotes equal contribution by the authors.

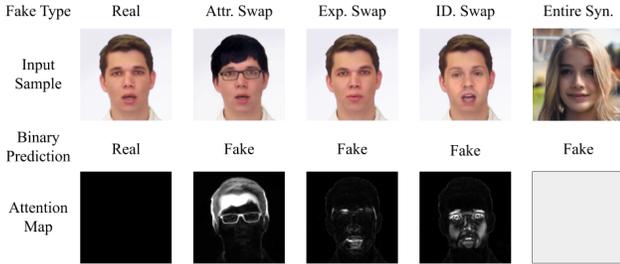


Figure 2. Our face forgery detection aims to tackle faces generated by four types of face manipulation methods. Given a face image, our proposed approach outputs the binary classification result and localizes the manipulated regions via estimated attention map. For real or entirely synthetic face images, our estimated attention maps are all zeros or ones.

These methods enable the transfer of facial expressions from one person to another person in real time with only RGB cameras. Identity swap methods replace the face of a person with the face of another. Examples include *FaceSwap*, which inserts famous actors into movie clips where they never appeared [7] and *DeepFakes*, which performs face swapping using deep learning algorithms. Attributes manipulation edits single or multiple attributes in a face, e.g., gender, age, skin color, hair, and glasses. The adversarial framework of GANs is used for image translation [20, 53, 54] or manipulation in a given context [6, 38], which diversifies facial images synthesis. *FaceApp* [2] has popularized facial attribute manipulation as a consumer-level application, which provides 28 filters to modify specific attributes [2]. The fourth category is entire face synthesis. Fueled by the large amounts of face data, along with the success of GANs, any amateur user is capable of producing a complete synthetic facial image. The realism achieved by the deep learning methods is such that even humans have difficulty assessing if a face image is genuine or manipulated. For instance, websites such as [thispersondoesnotexist.com](http://thispersondoesnotexist.com) and [thecleverest.com/judgefakepeople](http://thecleverest.com/judgefakepeople) offer evidence of the level of realism achieved by GAN-based methods [22, 12, 23].

Research on the face manipulation detection has been seriously hampered by the lack of adequate datasets. Existing approaches are often evaluated on small datasets with limited manipulation types, including Zhou *et al.* [52], Deepfake [26], and FaceForensics/FaceForensics++ [34, 35]. To remedy this issue, we collect a diverse fake face dataset (DFFD) of 2.6 million images from all four categories of digital face manipulations defined above.

Due to the fact that the modification of a face image can be in whole or in part, we assume that a well-learned network would gather different amount of information *spatially*, in order to detect manipulated faces. We hypothesize

that correctly estimating this spatial information can enable the network to focus on these important spatial regions to make its decision. Hence, here we aim to not only detect manipulated faces, but also automatically locate the manipulated regions, by estimating an image-specific attention map, as in Fig. 2. We also demonstrate that this attention map is beneficial to the final task of face forgery detection. In the future, we hope the predicted attention maps for manipulated face images and videos could reveal hints about the type, magnitude, and intention of the manipulation.

In summary, the contributions of this work include:

- ◊ A comprehensive fake face dataset include 0.8M real faces and 1.8M fake faces generated by diverse face modification methods, and an accompanying evaluation protocol.
- ◊ A novel attention-based layer to improve classification performance and produce an attention map indicating the manipulated facial regions.
- ◊ A novel metric, termed Inverse Intersection Non-Containment (IINC), for evaluating attention maps that produces a more coherent evaluation than existing metrics.
- ◊ The state-of-the-art performance of face manipulation detection in comparison to the strong baseline network.

## 2. Related Work

**Digital Face Manipulation Methods.** With the rapid progress in computer graphics and computer vision, the quality of digital face manipulation has reached to a level where it is difficult for humans to tell the difference between genuine and manipulated faces [35]. Graphics-based approaches are widely used for identity or expression transfer by first reconstructing 3D models for both source and target faces, and then exploiting the corresponding 3D geometry (identity or expression) to warp between them. In particular, Thies *et al.* [39] present expression swap for facial reenactment using a consumer-level RGB-D camera. *Face2Face* [40] is an advanced real-time face reenactment system using only an RGB camera. Instead of manipulating expression only, the extended work “Deep Video Portraits” [24] can transfer the full 3D head position, head rotation, expression, and eye blinking from a source actor to a portrait video of a target actor. “Synthesizing Obama” [38] animates the face of a person based on an input audio signal. *FaceSwap* can replace the identity of 3D models while preserving the original expressions.

Deep learning techniques, not surprisingly, are now popular to synthesize or manipulate faces [41]. The term *Deepfakes* has widely become a synonym for face identity replacement based on deep learning [35]. There are various public implementations of *Deepfakes*, most recently by ZAO [3] and *FaceAPP* [2]. *FaceAPP* can selectively modify parts of a face [2]. GAN-based methods can produce entire synthetic face images, including non-face background. PG-GAN [22] and StyleGAN [23] improve the image quality

with a progressive growing strategy or a novel generator.

**Fake Face Benchmarks.** Unfortunately, large and diverse datasets for face manipulation detection and evaluation are limited in the community. Zhou *et al.* [52] collected a dataset with face-swapped images generated by an iOS app and an open-source face swap software. Video-based face manipulation became available with the release of FaceForensics [34], which contains 0.5M *Face2Face* manipulated frames from over 1,000 videos. An extended version, FaceForensics++ [35], further augments the collection with *Deepfake* [1] and *FaceSwap* manipulations. However, these datasets are still limited to two fake types (identity and expression swap). To overcome this limitation, we collect the first fake face dataset that includes diverse fake types, *i.e.*, identity and expression swapped images from FaceForensics++, face attribute manipulated images using *FaceAPP*, and complete fake face images using StyleGAN [23] and PGGAN [22]. Our dataset is described in detail in Sec. 4.

**Attention Mechanism Localization.** There are two common approaches to localize manipulated image regions: segmenting the entire image [5, 32], and repeatedly performing binary classification using a sliding window [35]. These localization methods are often implemented via multi-task learning with additional supervision, and they do not directly improve the final image classification performance. In contrast, we propose an *attention mechanism* to automatically detect the manipulated region for face images, which requires very few trainable parameters (either zero or one conv filter in our implementations). In computer vision, attention models have been widely used for image classification [9, 47, 43], image inpainting [27, 51] and object detection [50, 8]. Attention not only serves to select a focused location but also enhances object representations at that location, which is effective for learning generalizable features for the given task. A number of methods [45, 46, 19] utilize the attention mechanism to enhance the accuracy of CNN classification models. Residual Attention Network [43] improves the accuracy of the classification model using 3D self-attention maps. Choe *et al.* [11] propose an attention-based dropout layer to process the feature maps of the model, which can be applied to CNN classifiers to improve the localization accuracy. To our knowledge, this is the *first* paper to apply the attention mechanism to face manipulation detection and localization.

### 3. Proposed Method

We cast the manipulated face detection as a binary classification problem by a CNN-based network. We further propose to utilize the attention mechanism to process the feature maps of the classifier model. The learned attention maps can highlight the regions in an image which influence the CNN’s decision, and further be used to guide the CNN to discover more discriminative features.

### 3.1. Motivation for the Attention Map

Assuming the attention map can highlight the regions of the image that are manipulated, and thereby guide the network to detect these regions. This alone should be useful for the face forgery detection. In fact, each pixel in the attention map would compute a probability that its receptive field corresponds to a manipulated region in the input image. For example, in the Xception network, the input image size is  $299 \times 299$  and the attention map is at a lower resolution, *e.g.*,  $19 \times 19$ . Each pixel in the attention map would describe the probability of manipulation for a local patch of size  $16 \times 16$  in the input image.

The attention map is effective at determining the manipulated regions because of the modification of high frequency noise in local patches of an image. Digital forensics has shown that camera model identification is possible due to “fingerprints” in the high-frequency information of a real image. Because of this, it is feasible to detect abnormalities in this high-frequency information due to algorithmic processing. Hence we insert the attention map into the backbone networks where the receptive field corresponds to appropriately sized local patches. Then, the features before the attention map encode the high-frequency fingerprint for the corresponding patch, which can be used to discriminate between real and manipulated regions at the local level.

Three major factors were considered during the construction and development of our attention map; *i)* explainability, *ii)* usefulness, and *iii)* modularity.

**Explainability:** Due to the fact that a face image can be modified entirely or in part, we produce an attention map that predicts where the *modified* pixels are. In this way, an auxiliary output is produced to explain which spatial regions the network based its decision on. This differs from prior works in that we use the attention map as a mask to remove any irrelevant information from the high-dimensional features *within the network*. During training, for a face image where the entire image is real, the attention map should ignore the entire image. For a modified or generated face, at least some parts of the image are manipulated, and therefore the attention map should focus only on these parts.

**Usefulness:** One requirement of our proposed attention map is that it enhances the final classification performance of the network. This is accomplished by feeding the attention map back into the network to ignore non-activated regions. This follows naturally from the fact that modified images may only be *partially modified*. Through the attention map, we can remove the real regions of a partial fake image so that the features used for final anti-fake classification are purely from modified regions.

**Modularity:** To create a truly utilitarian solution, we take great care to maintain the modularity of the solution. Our proposed attention map can be implemented easily and inserted into existing backbone networks, through the inclu-

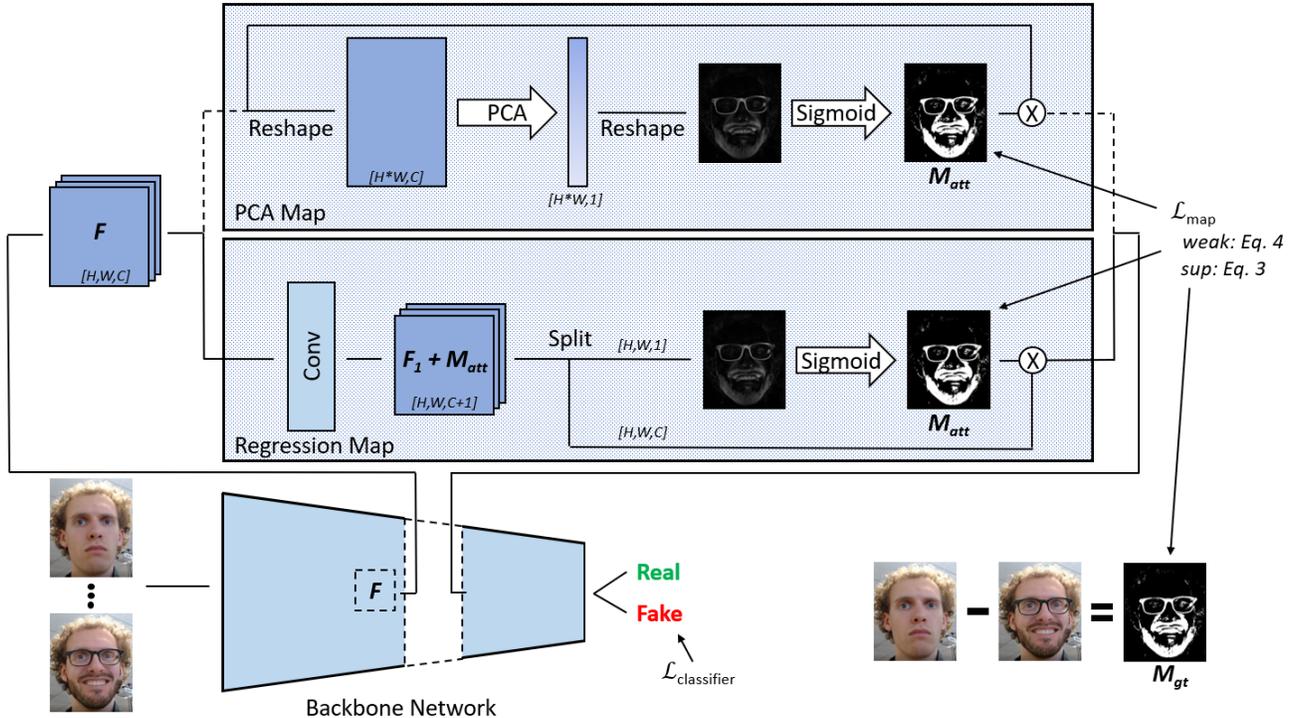


Figure 3. The architecture of our face manipulation detection. Given any backbone network, our proposed attention-based layer can be inserted into the network, where one specific example of this insertion is in Fig. 6. It takes the high-dimensional feature  $\mathbf{F}$  as input, estimates an attention map  $\mathbf{M}_{att}$  using either *PCA* or *regression*-based methods, and channel-wise multiplies with the high-dimensional feature, which is fed back into the backbone. In addition to the binary classification supervision  $\mathcal{L}_{classifier}$ , either supervised or weakly supervised loss,  $\mathcal{L}_{map}$ , can be applied to estimate the attention map, depending on whether ground truth manipulation map  $\mathbf{M}_{gt}$  is available.

sion of a single convolution layer, its associated loss functions, and masking the subsequent high-dimensional features. This can even be done while leveraging pre-trained networks by initializing only the weights that are used to produce the attention map.

### 3.2. Attention-based Layer

As shown in Fig. 3, the attention-based layer can be applied to any feature map of a classification model, and induce the model to learn the manipulated regions of the fake image. Specifically, the input of the attention-based layer is a convolutional feature map  $\mathbf{F} \in \mathbb{R}^{B \times H \times W \times C}$ , where  $B$  is the batch size, and  $H$ ,  $W$ ,  $C$  are height, width, and the number of channels, respectively. Then we can generate an attention map  $\mathbf{M}_{att} \in \mathbb{R}^{B \times H \times W}$  by processing  $\mathbf{F}$ . Here, we propose two approaches to implement the attention-based layer: self-attention PCA projection and direct regression.

**Self-attention PCA Projection.** Instead of simply compressing  $\mathbf{F}$  via channel-wise average pooling to generate the attention map [11], we adopt Principal Component Analysis (PCA) to summarize features over channel so that we can observe the largest variation in the high-dimensional space [10, 29]. Firstly,  $\mathbf{F}$  is reshaped into a matrix  $\mathbf{X} \in$

$\mathbb{R}^{(B \cdot H \cdot W) \times C}$ . We utilize Singular Value Decomposition (SVD) to find the largest eigenvector  $\mathbf{v} \in \mathbb{R}^{C \times 1}$  of the covariance matrix  $(\mathbf{X} - \mu)^T (\mathbf{X} - \mu)$ , where  $\mu$  is the row-wise mean of  $\mathbf{X}$ . By projecting mean-removed  $\mathbf{X}$  onto  $\mathbf{v}$  and applying a Sigmoid activation, we obtain the attention map:

$$\mathbf{M}_{att} = \text{Sigmoid}((\mathbf{X} - \mu)^T \mathbf{v}). \quad (1)$$

Since the entire network is trained to perform binary classification, the feature map  $\mathbf{X}$  includes information discriminative between real and fake facial parts. Performing PCA will discover the feature dimension  $\mathbf{v}$  that is the *most* discriminative between real and fake, and project mean-removed samples of a mini-batch to this dimension. Assuming the real samples are on the negative side of this dimension and fake samples on the positive side, the Sigmoid function turns former to be near 0 and the latter to be bounded at 1. As a result, the intensity of each pixel in the attention map is close to 0 for the real regions, and close to 1 for the fake regions. In other words, the pixel of the attention map indicates the probability of the original image patch being a fake region. Note that within a  $B$ -sample mini-batch, 60% of the samples are fake images and 40% are real ones, in order to balance the numbers of real and

Table 1. Comparison of fake face datasets along different aspects: number of still images, number of videos, number of fake types (identity swap (Id. swap), expression swap (Exp. swap), attributes manipulation, and entire image synthesis (Entire syn.)) and pose variation.

Dataset	Year	# Still images		# Video clips		# Fake types				Pose variation
		Real	Fake	Real	Fake	Id. swap	Exp. swap	Attr. mani.	Entire syn.	
Zhou <i>et al.</i> [52]	2018	2,010	2,010	-	-	2	-	-	-	Unknown
Yang <i>et al.</i> [49]	2018	241	252	49	49	1	-	-	-	Unknown
Deepfake [26]	2018	-	-	-	620	1	-	-	-	Unknown
FaceForensics++ [35]	2019	-	-	1,000	3,000	2	1	-	-	$[-30^\circ, 30^\circ]$
FakeSpotter [44]	2019	6,000	5,000	-	-	-	-	-	2	Unknown
<b>DFFD</b> (proposed)	2019	58,703	240,527	1,000	3,000	2	1	28 + 40	2	$[-90^\circ, 90^\circ]$

fake at the level of image patches, as many fake images are only partially manipulated.

The PCA projection layer uses self-attention which does not require additional trainable parameters. Finally, the attention map is channel-wise multiplied with the feature map  $\mathbf{F}$ . This helps the subsequent backbone network to focus its processing to the non-zeros areas of the attention map, *i.e.*, the fake regions.

**Direct Regression.** Another option to implement the attention-based layer is to compute the spatial attention map using auxiliary convolutional layer(s). Simply, we add one more channel in the previous convolutional layer to generate the feature map of size  $\mathbf{F} \in \mathbb{R}^{B \times H \times W \times (C+1)}$ .  $\mathbf{F}$  can be split into  $\mathbf{F}_1 \in \mathbb{R}^{B \times H \times W \times C}$  and  $\mathbf{M}_{att} \in \mathbb{R}^{B \times H \times W \times 1}$ . The last channel with sigmoid function serves as the attention map. The attention map is applied to the feature map  $\mathbf{F}_1$  by channel-wise multiplication and fed to the subsequent convolutional layers.

This direct regression method is simple, yet effective, for adaptive feature refinement. Later we show that the benefits of our proposed attention layer are realized regardless of the choice of backbone networks. This further validates our claim that the proposed solution is modular and improves the usefulness and flexibility of the attention map.

### 3.3. Loss Functions

To train the binary classification network, it is possible to begin with a pre-trained backbone network or to learn the backbone network from scratch. Either way, the overall loss for the training is as follows:

$$\mathcal{L} = \lambda_c * \mathcal{L}_{\text{classifier}} + \lambda_m * \mathcal{L}_{\text{map}}, \quad (2)$$

where  $\mathcal{L}_{\text{classifier}}$  is the binary classification loss of Softmax and  $\mathcal{L}_{\text{map}}$  is the attention map loss.  $\lambda_c$  and  $\lambda_m$  are the respective loss weights.

For attention map learning, we consider three different cases: supervised, weakly supervised, and unsupervised.

**Supervised learning.** If the training samples are paired with ground truth attention masks, we can train the network in a supervised fashion, shown in Eqn. 3.

$$\mathcal{L}_{\text{map}} = |\mathbf{M}_{att} - \mathbf{M}_{gt}|, \quad (3)$$

where  $\mathbf{M}_{gt}$  is the ground truth manipulation mask. We use zero-maps as the  $\mathbf{M}_{gt}$  for real faces, and one-maps as the  $\mathbf{M}_{gt}$  for entire synthesized fake faces. For partially manipulated faces, we pair fake images with their corresponding source images, compute the absolute pixel-wise difference in the RGB channels, convert into grayscale, and divide by 255 to produce a map in the range of  $[0, 1]$ . We finally set a threshold of 0.1, which is empirically determined and validated in later experiments, to obtain the binarized modification map as  $\mathbf{M}_{gt}$ . We posit this strong supervision can help attention-based layer to learn the most discriminative regions and features for fake face detection.

**Weakly supervised learning.** For partially manipulated faces, sometimes the source images are not available. Hence, we can not obtain the ground truth manipulation mask as described above. However, we would still like to include these faces in learning the attention maps. To this end, we propose a weak supervision map loss as in Eqn. 4:

$$\mathcal{L}_{\text{map}} = \begin{cases} |\mathbf{M}_{att}^i - 0|, & \text{if real} \\ |\max(\mathbf{M}_{att}^i) - 0.75|, & \text{if fake} \end{cases} \quad (4)$$

where  $\mathbf{M}_{att}^i$  is the attention map of the  $i$ -th sample in a mini-batch. This loss drives the attention map to remain un-activated for real images, *i.e.*, zeros. For fake images, regardless of entire or partial manipulation, the maximum map value should be a sufficiently large value, 0.75 in our experiments. This implies that, for partial manipulation, it is acceptable that most of the map values are zeros, as long as at least one modified local region has a large map value.

**Unsupervised learning.** We can train the network without any map supervision when  $\lambda_m$  is set to 0. With only image-level classification supervision, the attention maps learn informative region automatically. More analysis of these three losses is available in the experiments section.

## 4. Diverse Fake Face Dataset

One of our contributions is the construction of a dataset with diverse types of fake faces, termed Diverse Fake Face Dataset (DFFD). Compared with the previous fake face datasets in Tab. 1, DFFD contains greater diversity in fake

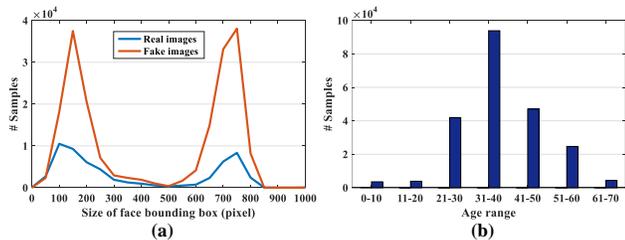


Figure 4. (a) Distribution of the face bounding box sizes (pixel) and (b) Age distribution of our DFFD.

types, which is crucial for research on the detection and localization of face manipulations.

**Data Collection.** In Sec. 1, we introduced four types of digital manipulation methods: identity swap, expression swap, attribute manipulation, and entire synthesized faces. We collect data from all these four categories by adopting respective state-of-the-art methods to generate fake images, described below. For DFFD, we analyze the gender and age distribution, and the face size of all samples, as in Fig. 4. Among all images and video frames, 47.7% are from male subjects, 52.3% are from female, and the majority of samples are from subjects in the range 21 – 50 years of age. For the face size, both real and fake samples have low quality, as well as high quality images. This ensures that the distributions of gender, age, and face size are less biased.

*Real face images.* We utilize FFHQ [23] and CelebA [30] datasets as our real face samples since the faces contained here cover comprehensive variations in race, age, gender, pose, illumination, expression, resolution, and camera capture quality. They have been used in digital fake face detection as real samples. We further utilize the source frames from FaceForensics++ [35] as additional real faces.

*Identity and expression swap.* For facial identity and expression swap, we use all the video clips from FaceForensics++ [35]. The FaceForensics++ contains 1,000 real videos collected from YouTube and their corresponding 3,000 manipulated versions which are divided into two groups: identity swap using *FaceSwap* and *Deepfake* [1], and expression swap using *Face2Face* [40]. From the website<sup>1</sup>, we also collected identity swap data, which are entertainment videos generated by *Deep Face Lab* (DFL)<sup>2</sup>.

*Attributes manipulation.* We adopt two methods *FaceAPP* [2] and *StarGAN* [12] to generate attribute manipulated images, where 4,000 faces of FFHQ and 2,000 faces of CelebA are the input source real images, respectively. *FaceAPP*, as a consumer-level smart phone application, provides 28 filters to modify specified facial attributes, e.g., gender, age, hair, skin color, beard, and glasses. The manipulated images are generated with an automated script

<sup>1</sup>[https://www.patreon.com/ctrl\\_shift\\_face](https://www.patreon.com/ctrl_shift_face)

<sup>2</sup><https://github.com/iperov/DeepFaceLab>

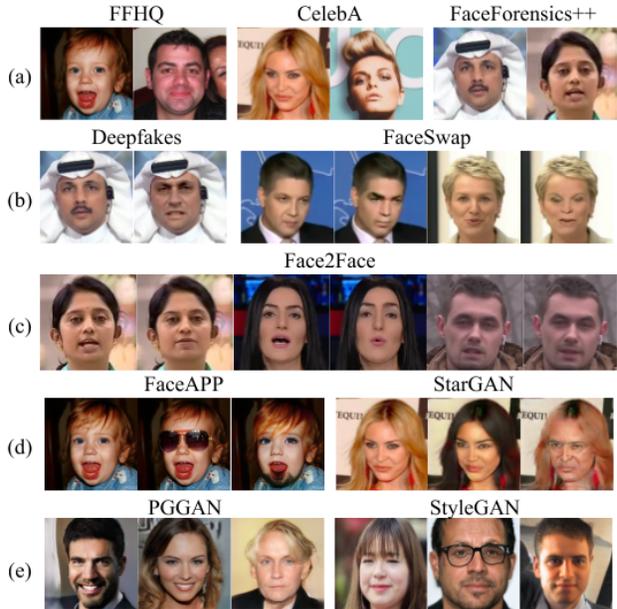


Figure 5. Example faces in our DFFD. (a) Real images/frames from FFHQ, CelebA and FaceForensics++ datasets; (b) Paired face identity swap images from FaceForensics++ dataset; (c) Paired face expression swap images from FaceForensics++ dataset; (d) Attributes manipulated examples by *FaceAPP* and *StarGAN*; (e) Entire synthesized faces by *PGGAN* and *StyleGAN*.

working on Android devices. For each face in FFHQ, we generate three corresponding fake images. Two of them are generated using a single random manipulation filter, and the last one uses multiple random manipulation filters. *StarGAN*, a GAN-based image-to-image translation method, can generate 40 types of facial attribute manipulations. For each face in CelebA, we generate 40 fake images. In total, we collect 92K attribute manipulated images.

*Entire face synthesis.* Recent works in image synthesis, *PGGAN* [22] and *StyleGAN* [23], achieve remarkable results in realistic face image synthesis. *PGGAN* proposes a progressive training methodology both for generator and discriminator, which can produce high-quality images. *StyleGAN* redesigns the generator by borrowing from style transfer literature. Consequently, we use the pre-trained model of *PGGAN* and *StyleGAN* to create 200k and 100k high-quality entire fake images, respectively. Figure 5 shows some examples of DFFD.

**Pre-processing.** We use *insightFace* [18] to estimate the bounding box and 5 landmarks for each image. We discard images whose detection or alignment fails. We further generate ground truth manipulation masks for fake images as described in Sec. 3.3. To enforce consistency, if a fake face image is derived from a source real face image, we use the same landmarks of the real face image for face cropping.

**Protocols.** As shown in Tab. 2, we collect 781,727 sam-

Table 2. Statistics of our DFFD composition and protocol. The total number of available samples, training samples, validation samples, and testing samples are 2,653,734, 124,731, 12,738 and 161,761, respectively.

Dataset		# Total Samples	# Training	# Validation	# Testing	Average face width (pixel)	
Real	FFHQ [23]	70,000	10,000	999	9,000	750	
	CelebA [30]	202,599	9,974	997	8,979	200	
	Original @ FaceForensics++ [35]	509,128	10,230	998	7,526	200	
Fake	Id. Swap	DFL	49,235	10,006	1,007	38,222	200
		Deepfakes @ FaceForensics++ [35]	509,128	10,230	999	7,517	200
		FaceSwap @ FaceForensics++ [35]	406,140	8,123	770	6,056	200
	Exp. Swap	Face2Face @ FaceForensics++ [35]	509,128	10,212	970	7,554	200
		FaceAPP [2]	18,416	6,000	1,000	5,000	700
	Attr. Manip.	StarGAN [12]	79,960	10,000	1,000	35,960	150
		PGGAN [22]	200,000	19,957	1,998	17,950	750
	Entire Syn.	StyleGAN [23]	100,000	19,999	2,000	17,997	750

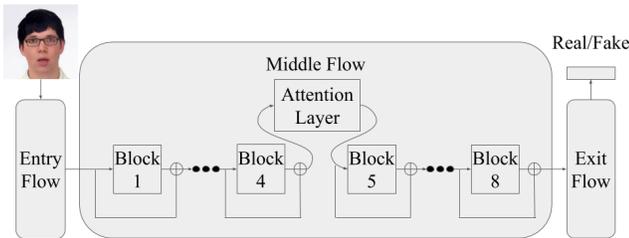


Figure 6. The overall architecture of XceptionNet and its enhancement with our proposed attention later. The original XceptionNet has entry flow, middle flow, and exit flow, where the middle flow is composed of 8 blocks and our attention layer (with detailed in Fig. 3) could be added after one of the blocks.

ples for real image/frames, and 1,872,007 samples for fake ones. Within these samples, we randomly select a subset of 58,703 real images/frames and 240,527 fake ones to make the size of our dataset manageable and to ensure the sizes of each sub-category is balanced. For video samples, we extract one frame per second in order to reduce the size without decreasing the diversity of DFFD. We randomly split the data into 50% for training, 5% for validation and 45% for testing. All fake images manipulated from the same real image are in the same set as the source image.

## 5. Experimental Result

We formulate the face manipulation detection as a binary classification problem. Here, we present the performance of the backbone networks with and without the proposed attention map for both binary classification and manipulated region localization. For all experiments, we utilize the protocols defined in Sec. 4. Unless otherwise stated, all results are computed using the entirety of the test set.

### 5.1. Experimental Setup

**Implementation Details:** In our experiments, we add the attention map and its associated loss functions to the XceptionNet [13], VGG Net [37], and a custom network.

The overall architecture of XceptionNet and how it is enhanced by our attention layer are shown in Fig. 6. Values of the loss weighting  $\lambda$  are set equal to each other and the batch size is 16, where each mini-batch consists of 6 real and 10 fake images. Depending on the backbone architecture, we train for 75k-150k iterations, which requires less than 8 hours on an NVidia GTX 1080Ti. We choose the best model based on the validation set.

**Metrics.** For classification, we report EER (Equal Error Rate), AUC (Area Under Curve) of ROC (Receiver Operating Characteristic) curves, TDR (True Detect Rate) at FDR (False Detect Rate) of 0.01% (denoted as  $TDR_{0.01\%}$ ), and TDR at FDR of 0.1% (denoted as  $TDR_{0.1\%}$ ), respectively. For localization, with known ground-truth masks, we report Pixel-wise Binary Classification Accuracy (PBCA), which treats each pixel as an independent sample to measure classification accuracy, Intersection over Union (IoU), and Cosine similarity, which is the Cosine between two. We also propose a novel metric, Inverse Intersection Non-Containment (IINC) for evaluating manipulated face region prediction, as described in Sec. 5.4.

### 5.2. Ablation Study

Here, we study three aspects of experiments, namely: *i*) benefit of attention map, *ii*) attention map placement, and *iii*) effect of backbone network architectures.

**Benefit of Attention map.** In this experiment, we utilize the state-of-the-art XceptionNet [13] architecture as our backbone network. It is based on depth-wise separable convolution layers with residual connections. Here, XceptionNet is pre-trained on ImageNet and fine-tuned on our dataset. During fine-tuning, the attention layer is inserted between Block 4 and Block 5 in the middle flow, as in Fig. 6. The last layer is modified into a fully connected layer with two outputs.

In Tab. 3, we show a comparison of the PCA projection and direct regression attention with different loss function strategies, including without supervision (*w/o sup.*), weakly supervised loss (*weak sup.*) and supervised loss (*sup.*).

Table 3. Ablation for the benefit of the attention map, with various combinations of map generation methods and supervisions. Boldface, blue and red indicates top three performance, respectively.

Map Supervision	AUC	EER	TDR <sub>0.01%</sub>	TDR <sub>0.1%</sub>	PBCA
Xception	99.61	2.88	<b>77.42</b>	85.26	—
+ Reg. Map, <i>w/o sup.</i>	<b>99.76</b>	<b>2.16</b>	77.07	<b>89.70</b>	12.89
+ Reg. Map, <i>weak sup.</i>	<b>99.66</b>	2.57	46.57	75.20	30.99
+ Reg. Map, <i>sup.</i>	99.64	<b>2.23</b>	<b>83.63</b>	<b>90.78</b>	<b>88.44</b>
+ Reg. Map, <i>sup.</i> - map	99.69	2.73	48.54	72.94	<b>88.44</b>
+ PCA Map, <i>w/o sup.</i>	99.02	4.94	69.53	78.53	20.95
+ PCA Map, <i>weak sup.</i>	<b>99.71</b>	<b>2.36</b>	72.07	86.71	<b>59.20</b>
+ PCA Map, <i>sup.</i>	99.64	2.39	<b>78.00</b>	<b>89.58</b>	<b>85.80</b>
+ PCA Map, <i>sup.</i> - map	99.53	2.90	52.26	64.21	<b>85.80</b>

Table 4. The performance of the attention map at different placements in the middle flow of the XceptionNet architecture.

Map position	AUC	EER	TDR <sub>0.01%</sub>	TDR <sub>0.1%</sub>	PBCA
Block1	99.82	<b>1.69</b>	71.46	<b>92.80</b>	83.30
Block2	<b>99.84</b>	1.72	67.95	90.14	87.41
Block3	99.50	2.82	49.06	72.50	88.14
Block4	99.64	2.23	<b>83.83</b>	90.78	88.44
Block5	99.49	2.62	82.70	89.03	88.40
Block6	99.72	2.28	63.08	86.02	87.41
Block7	99.78	1.79	28.51	88.98	88.39
Block8	98.62	4.42	74.24	79.95	<b>88.96</b>

Given the overall strong performance for all entries and the preferred operational point of low FDR in practices, we focus on the TDR metrics for comparing various methods. First of all, the *sup* cases outperform the weak supervision or without supervision, in both the binary classification and the map classification accuracy. Second, when the ground truth map supervision is not available, the proposed PCA-projection is superior to the direct regression. We hypothesize part of the reason is that PCA-projection has no trainable parameters in the attention-based layer. This shows the strength of learning few parameters when there is a lack of supervision. Due to the superior performance of the regression attention map with supervision, we use this method for all other experiments.

Instead of using the softmax output, an alternative is to use the average of the estimated attention map for binary classification, since loss functions desire low attention values for real faces while higher values for fake faces. The performance of this alternative is shown in the rows for ‘+ Reg. Map, *sup.* - map’ and ‘+ PCA Map, *sup.* - map’ in Tab. 3. The substantial lower performance comparing to using softmax output shows that it is important to have the feedback from the attention map to the subsequent network through channel-wise multiplication.

**Attention Layer Placement.** In this experiment, we investigate the effects of the placement of the attention layer within the XceptionNet middle flow. As shown in Fig. 6, the attention layer can be placed after any of the eight blocks. Tab. 4 shows that the attention layer placement can signif-

Table 5. The performance of three backbone architectures with and without the attention maps.

Network	AUC	EER	TDR <sub>0.01%</sub>	TDR <sub>0.1%</sub>
Xception	99.61	2.88	77.42	85.26
Xception + Reg. Map	99.64	2.23	83.83	90.78
Xception + PCA Map	99.64	2.39	78.00	89.58
VGG	96.95	8.43	0.00	51.14
VGG + Reg. Map	99.46	3.40	44.16	61.97
VGG + PCA Map	98.81	5.20	0.00	75.39
Custom	98.08	6.54	29.91	52.92
Custom + Reg. Map	98.66	5.41	25.21	58.45
Custom + PCA Map	99.20	4.33	34.06	71.14

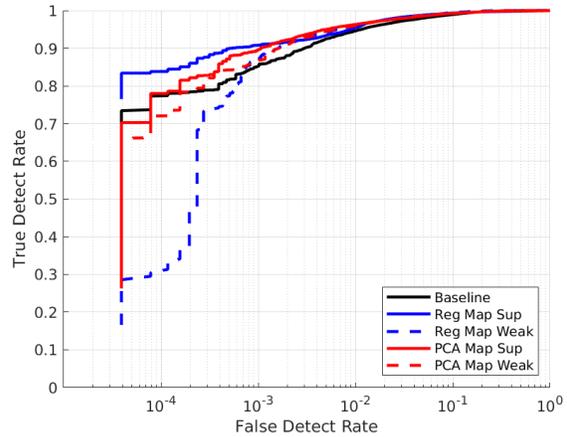


Figure 7. Fake face detection ROCs with various XceptionNet models.

icantly influence the binary classification task. Two trends are evident. Placement of the attention layer early in the network improves the EER. Later placement improves the map classification accuracy, since it increases the receptive field of each pixel in the attention map, and hence can rely on more information to make a decision locally. However, middle placement exhibits strong performance for all metrics, without significant degradation. In the rest of our experiments, results are reported based on ‘Block4’, *i.e.*, placement after Block4.

**Dependency on Backbone Networks.** We use VGG16 [37], XceptionNet [13], and our custom network as backbone networks. For VGG16, the attention layer is inserted after the third convolution block, which produces a 28x28 feature map. The custom network is composed of 3 convolution blocks, each with 3 convolution layers followed by max pooling. The outputs of each block are concatenated together and passed through 2 more convolution layers. These features are used to produce the attention map of size 16x16, and after channel-wise multiplication with it, are further processed to produce the final binary classification. Tab. 5 results validate that using attention map does improve the backbone network performance no matter which backbone network we use. In all three cases, the



Figure 8. Failure examples of the Xception with Regression Map under supervision. From left to right, the columns are top 3 worst samples of real, identity manipulated, expression manipulated, completely generated, and attribute modified, respectively.

Table 6. Fake face detection performance of the Xception Regression Map with supervision for each fake type.

Fake Type	AUC	EER	TDR <sub>0.01%</sub>	TDR <sub>0.1%</sub>
ID Manip.	99.43	3.11	65.16	77.76
EXP Manip.	99.40	3.40	71.23	80.87
Attr. Manip.	99.92	1.09	81.32	90.93
Entire Syn.	100.00	0.05	99.89	99.96

application of attention maps improved the performance of backbone architectures. For the XceptionNet and VGG, the regression map shows better performance, but for the custom network, the PCA map shows better performance. This is due to the smaller size of the custom network and the fact that it is trained from scratch, rather than fine-tuned, on our dataset. Training from scratch allows the PCA map to guide the feature learning from the early stage to maximize its impact, whereas for fine-tuning, the PCA map must gradually adapt to the pre-trained feature extraction.

### 5.3. Fake Face Detection

In Fig. 7, we show the ROCs of the XceptionNet with and without the attention map. From this figure, it is clear that the direct regression approach for the attention map produces the best performing network at very low FDR, which is not only the most challenging scenario, but also the most relevant to the practical applications. This also demonstrates that our proposed approach substantially outperforms the conventional XceptionNet-based manipulation detection, especially at lower FDR.

For our best performing model, Xception Regression Map with supervision, we conduct failure analysis in two aspects. (i) Fig. 8 shows the worst 3 test samples among the real test faces and each fake types. For example, the images in the first column have the lowest Softmax probability of being the real class. Among these samples, some have heavy makeup, and others are of low image quality. (ii) Tab. 6 shows the accuracy of testing samples in each

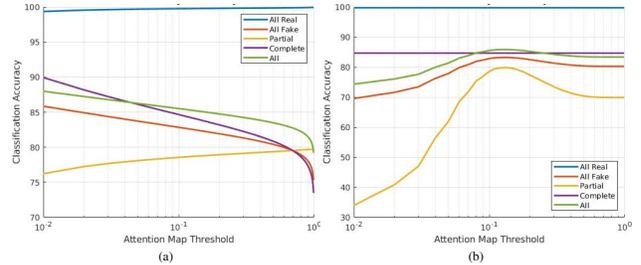


Figure 9. The attention map estimation performance of the proposed method when using different thresholds to binarize the predicted map (a) and the ground truth map (b). The threshold for the other map in either case was 0.1.

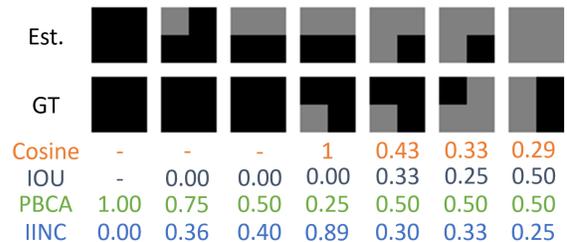


Figure 10. A toy-example comparison of 4 metrics for evaluating the attention map estimation. GT and Est. are the ground truth and estimated binary map, respectively, where white is the manipulated pixel and black is the real pixel. The IOU and Cosine metrics do not adequately reflect the differences in the first 3 examples. Similarly, the PBCA is not useful for the last 3 examples. In contrast, the proposed IINC metric is discriminative in all cases.

fake type. The completely synthesized images appear to be the easiest for detection. This is due to the artificial “fingerprint” these methods leave on the generated images, which is easily distinguishable from real images. In contrast, identity and expression manipulated images are the most challenging to detect, where image quality could be one potential reason, as in the second and third columns of Fig. 8.

### 5.4. Attention Map Estimation

The accuracy of attention map estimation of course depends on the threshold used to binarize the map. In Fig. 9, we show the accuracy of the attention map estimation, as a function of the threshold. This demonstrates that our choice of 0.1 as a threshold for both maps is appropriate.

We utilize three popular metrics for evaluating the attention maps: Intersection Over Union (IOU), Cosine Similarity, and Pixel-wise Binary Classification Accuracy (PBCA). However, these three metrics are inadequate for robust evaluation of these diverse maps. Thus, we propose a novel metric defined in Eqn. 5, termed Inverse Intersection Non-

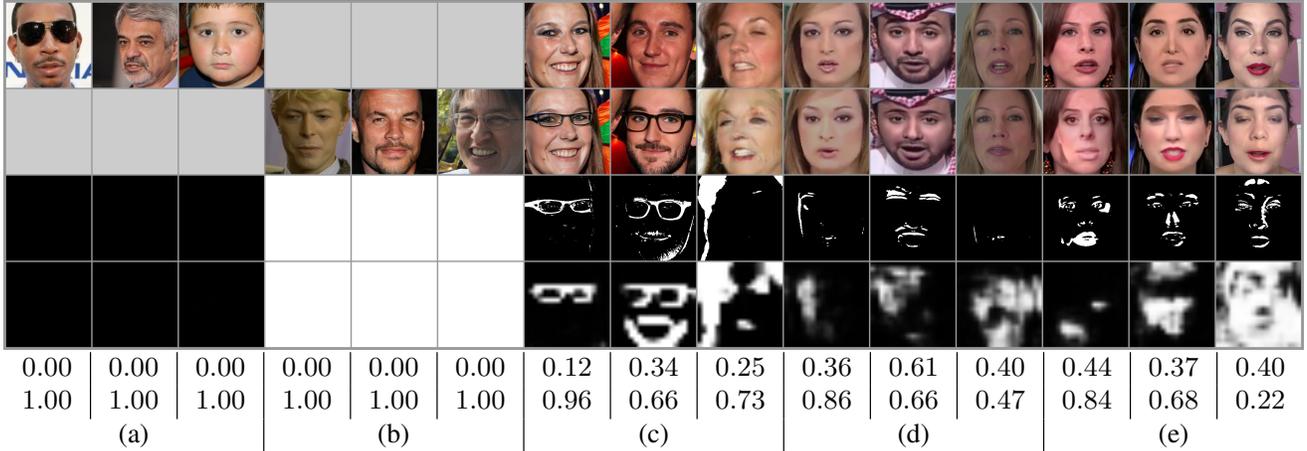


Figure 11. Example source images, manipulated images, ground truth manipulation masks, and estimated attention maps of (a) real, (b) entirely synthesized, (c) attribute manipulation, (d) expression manipulation, and (e) identity manipulation. Below the images are the IINC and PBCA between the maps.

Containment (IINC), to evaluate the predicted maps:

$$\text{IINC} = \frac{1}{3 - |\mathbf{U}|} * \begin{cases} 0 & \text{if } \overline{\mathbf{M}}_{gt} = 0 \text{ and } \overline{\mathbf{M}}_{att} = 0 \\ 1 & \text{if } \overline{\mathbf{M}}_{gt} = 0 \text{ xor } \overline{\mathbf{M}}_{att} = 0 \\ \left(2 - \frac{|\mathbf{I}|}{|\mathbf{M}_{att}|} - \frac{|\mathbf{I}|}{|\mathbf{M}_{gt}|}\right) & \text{otherwise,} \end{cases} \quad (5)$$

where  $\mathbf{I}$  and  $\mathbf{U}$  are the intersection and union between the ground truth map,  $\mathbf{M}_{gt}$ , and the predicted attention map,  $\mathbf{M}_{att}$ , respectively.  $\overline{\mathbf{M}}$  and  $|\mathbf{M}|$  are the mean and  $L_1$  norm of  $\mathbf{M}$ , respectively. The two fractional terms measure the ratio of the area of the intersection with respect to the area of each map, respectively. This improves upon other metrics by measuring the non-overlap ratio of both maps, rather than their combined overlap, as in the IOU.

The benefits of the IINC metric as compared with other metrics are shown in Fig. 10. Note that the IOU and Cosine similarity metrics are not useful at all for the first 3 samples, where the scores are the same, but the maps have vastly different properties. Similarly, the PBCA is not useful for the last 3 cases, because the ratio of mis-classification is not represented in the score. E.g., the last case overestimates by a factor of 100% and the third to last overestimates by 200%, while the second to last both over- and under-estimates by 150%. The IINC provides the optimal ordering by producing the same order as the IOU when it is useful (last 4 cases) and similarly with the PBCA when it is useful (first 3 cases). This shows that IINC is a more useful and robust metric for comparing the attention maps than any of the previous metrics.

The ability of our best performing model (Xception Regression Map with supervision) to predict the attention maps is shown in Tab. 7. In Fig. 11, we show the IINC and PBCA for test images. The ordering of the IINC scores lines up with a qualitative human analysis. The first cases in (d) and (e) are examples where the PBCA is high simply be-

Table 7. Evaluating attention maps with 4 metrics. Note IOU and Cosine similarity are not appropriate to evaluate real images.

Data	IINC	IOU	Cosine Sim.	PBCA
All Real	0.015	—	—	0.998
All Fake	0.147	0.715	0.192	0.828
Partial	0.311	0.401	0.429	0.786
Complete	0.077	0.847	0.095	0.847
All	0.126	—	—	0.855

cause the majority of each map is non-activated. The IINC is more discriminative in these cases due to the non-overlap between the maps. For the third cases in (d) and (e), the IINC produces the same score because the maps display the same behavior (a large amount of over-activation), whereas the PBCA prefers the example in (d) because the maps have fewer activated pixels.

## 6. Conclusion

This paper tackles the digitally manipulated face image detection and localization problem. The proposed method leverages attention mechanism to process the feature maps of the binary classification model. The learned attention maps can highlight the informative face regions for improving the classification ability, and can also highlight the manipulated regions. In addition, we collect the first forgery face dataset that contains diverse types of fake faces. Finally, we empirically show that the use of attention mechanism improves fake detection, and manipulated facial region localization. This is the first unified approach that tackles a diverse set of face manipulation methods, and also achieves the state-of-the-art performance in comparison to prior network architectures.

## References

- [1] Deepfakes github. <https://github.com/deepfakes/faceswap>. Accessed: 2019-09-11. 3, 6
- [2] FaceApp. <https://faceapp.com/app>. Accessed: 2019-09-04. 2, 6, 7
- [3] ZAO. <https://apps.apple.com/cn/app/zao/id1465199127>. Accessed: 2019-09-16. 2
- [4] Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li. Protecting world leaders against deep fakes. In *ICCVW*, pages 38–45, 2019. 1
- [5] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *PAMI*, 39(12):2481–2495, 2017. 3
- [6] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B Tenenbaum, William T Freeman, and Antonio Torralba. Gan dissection: Visualizing and understanding generative adversarial networks. *arXiv preprint arXiv:1811.10597*, 2018. 2
- [7] Jennifer Finney Boylan. Will deep-fake technology destroy democracy? *The New York Times*, Oct, 17, 2018. 2
- [8] Juan C Caicedo and Svetlana Lazebnik. Active object localization with deep reinforcement learning. In *ICCV*, pages 2488–2496, 2015. 3
- [9] Chunshui Cao, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, et al. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *CVPR*, pages 2956–2964, 2015. 3
- [10] Tsuhan Chen, Yufeng Jessie Hsu, Xiaoming Liu, and Wende Zhang. Principle component analysis and its variants for biometrics. In *ICIP*, volume 1, 2002. 4
- [11] Junsuk Choe and Hyunjung Shim. Attention-based dropout layer for weakly supervised object localization. In *CVPR*, pages 2219–2228, 2019. 3, 4
- [12] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, pages 8789–8797, 2018. 2, 6, 7
- [13] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, pages 1251–1258, 2017. 7, 8
- [14] Kevin Dale, Kalyan Sunkavalli, Micah K Johnson, Daniel Vlasic, Wojciech Matusik, and Hanspeter Pfister. Video face replacement. In *ACM Transactions on Graphics (TOG)*, volume 30, page 130, 2011. 1
- [15] Debayan Deb, Jianbang Zhang, and Anil K Jain. Advfaces: Adversarial face synthesis. *arXiv preprint arXiv:1908.05008*, 2019. 1
- [16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014. 1
- [17] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1
- [18] Jia Guo, Jiankang Deng, Niannan Xue, and Stefanos Zafeiriou. Stacked dense u-nets with dual transformers for robust face alignment. In *BMVC*, 2018. 6
- [19] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018. 3
- [20] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, pages 1125–1134, 2017. 2
- [21] Amin Jourabloo, Yaojie Liu, and Xiaoming Liu. Face deep-faking: Anti-spoofing via noise modeling. In *ECCV*, pages 290–306, 2018. 1
- [22] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 2, 3, 6, 7
- [23] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 2, 3, 6, 7
- [24] Hyeonwoo Kim, Pablo Carrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Niessner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt. Deep video portraits. *Transactions on Graphics (TOG)*, 37(4):163, 2018. 2
- [25] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 1
- [26] Pavel Korshunov and Sébastien Marcel. Deepfakes: a new threat to face recognition? assessment and detection. *arXiv preprint arXiv:1812.08685*, 2018. 2, 5
- [27] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *ECCV*, pages 85–100, 2018. 3
- [28] Yaojie Liu, Amin Jourabloo, and Xiaoming Liu. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In *CVPR*, pages 389–398, 2018. 1
- [29] Yaojie Liu, Joel Stehouwer, Amin Jourabloo, and Xiaoming Liu. Deep tree learning for zero-shot face anti-spoofing. In *CVPR*, pages 4680–4689, 2019. 1, 4
- [30] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, December 2015. 6, 7
- [31] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 1
- [32] Huy H Nguyen, Fuming Fang, Junichi Yamagishi, and Isao Echizen. Multi-task learning for detecting and segmenting manipulated facial images and videos. *arXiv preprint arXiv:1906.06876*, 2019. 3
- [33] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, pages 1278–1286, 2014. 1
- [34] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics: A large-scale video dataset for forgery detection in human faces. *arXiv preprint arXiv:1803.09179*, 2018. 1, 2, 3

- [35] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. *arXiv preprint arXiv:1901.08971*, 2019. [1](#), [2](#), [3](#), [5](#), [6](#), [7](#)
- [36] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36, 2017. [1](#)
- [37] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. [7](#), [8](#)
- [38] Supasorn Suwajanakorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Synthesizing obama: learning lip sync from audio. *Transactions on Graphics (TOG)*, 36(4):95, 2017. [2](#)
- [39] Justus Thies, Michael Zollhöfer, Matthias Nießner, Levi Valgaerts, Marc Stamminger, and Christian Theobalt. Real-time expression transfer for facial reenactment. *ACM Trans. Graph.*, 34(6):183–1, 2015. [2](#)
- [40] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2Face: Real-time face capture and reenactment of RGB videos. In *CVPR*, 2016. [1](#), [2](#), [6](#)
- [41] Luan Tran and Xiaoming Liu. On learning 3D face morphable model from in-the-wild images. *TPAMI*, June 2019. doi:10.1109/TPAMI.2019.2927975. [2](#)
- [42] Luan Tran, Xi Yin, and Xiaoming Liu. Representation learning by rotating your faces. *TPAMI*, September 2018. doi:10.1109/TPAMI.2018.2868350. [1](#)
- [43] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *CVPR*, pages 3156–3164, 2017. [3](#)
- [44] Run Wang, Lei Ma, Felix Juefei-Xu, Xiaofei Xie, Jian Wang, and Yang Liu. Fakespotter: A simple baseline for spotting ai-synthesized fake faces. *arXiv preprint arXiv:1909.06122*, 2019. [5](#)
- [45] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803, 2018. [3](#)
- [46] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, pages 3–19, 2018. [3](#)
- [47] Tianjun Xiao, Yichong Xu, Kuiyuan Yang, Jiaying Zhang, Yuxin Peng, and Zheng Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *CVPR*, pages 842–850, 2015. [3](#)
- [48] Han Xu, Yao Ma, Haochen Liu, Debayan Deb, Hui Liu, Jiliang Tang, and Anil Jain. Adversarial attacks and defenses in images, graphs and text: A review. *arXiv preprint arXiv:1909.08072*, 2019. [1](#)
- [49] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses. In *ICASSP*, pages 8261–8265, 2019. [5](#)
- [50] Donggeun Yoo, Sunggyun Park, Joon-Young Lee, Anthony S Paek, and In So Kweon. Attentionnet: Aggregating weak directions for accurate object detection. In *ICCV*, pages 2659–2667, 2015. [3](#)
- [51] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *CVPR*, pages 5505–5514, 2018. [3](#)
- [52] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Two-stream neural networks for tampered face detection. In *CVPRW*, pages 1831–1839, 2017. [2](#), [3](#), [5](#)
- [53] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, pages 2223–2232, 2017. [2](#)
- [54] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *NIPS*, pages 465–476, 2017. [2](#)
- [55] Michael Zollhöfer, Justus Thies, Pablo Garrido, Derek Bradley, Thabo Beeler, Patrick Pérez, Marc Stamminger, Matthias Nießner, and Christian Theobalt. State of the art on monocular 3d face reconstruction, tracking, and applications. In *Computer Graphics Forum*, volume 37, pages 523–550, 2018. [1](#)