Face Anti-Spoofing: Model Matters, So Does Data

Xiao Yang[12]*, Wenhan Luo[2]*, Linchao Bao[2], Yuan Gao[2], Dihong Gong[2], Shibao Zheng[1], Zhifeng Li[2]†, Wei Liu[2]†
[1] Department of Electronic Engineering, Shanghai Jiao Tong University, China
[2] Tencent AI Lab, Shenzhen, China
{xiao_yang, sbzh}@sjtu.edu.cn, {whluo.china, linchaobao, ethan.y.gao, gongdihong}@gmail.com,
michaelzfli@tencent.com, wl2223@columbia.edu

Abstract

Face anti-spoofing is an important task in full-stack face applications including face detection, verification, and recognition. Previous approaches build models on datasets which do not simulate the real-world data well (e.g., small scale, insignificant variance, etc.). Existing models may rely on auxiliary information, which prevents these anti-spoofing solutions from generalizing well in practice. In this paper, we present a data collection solution along with a data synthesis technique to simulate digital medium-based face spoofing attacks, which can easily help us obtain a large amount of training data well reflecting the real-world scenarios. Through exploiting a novel Spatio-Temporal Anti-Spoof Network (STASN), we are able to push the performance on public face anti-spoofing datasets over state-of-the-art methods by a large margin. Since the proposed model can automatically attend to discriminative regions, it makes analyzing the behaviors of the network possible. We conduct extensive experiments and show that the proposed model can distinguish spoof faces by extracting features from a variety of regions to seek out subtle evidences such as borders, moire patterns, reflection artifacts, etc.

1. Introduction

Face anti-spoofing [40, 1, 15, 14] is an important, yet challenging problem in the face recognition community. It has wide practical applications in face authentication, security check, and access control. This task is to recognize whether a face is captured from spoof attacks, including printed face, replaying a face video with digital medium, wearing a mask, *etc*. Therefore, face anti-spoofing is quite vital to the security of face recognition systems [39].

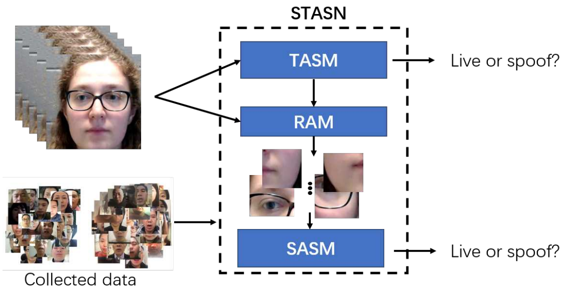

Figure 1. The proposed model STASN includes three components, TASM, RAM and SASM. TASM extracts temporal and global feature representation, SASM learns discrimination from local important regions, attended by RAM. With our collected data, the model achieves the state-of-the-art performance.

Previous methods have made a progress in achieving acceptable accuracy in recent years, while are hardly adoptable in practical applications due to various reasons. For example, one reason may be that some methods are trained using datasets which are of small scale and/or indistinctive variations. This prevents these methods from generalizing well. Moreover, some algorithms rely on additional information, such as rPPG [26] and depth [2]. The performance depends on the quality of the auxiliary information to some extent. The dependence may also bring inconvenience in practice. Besides, existing datasets for face anti-spoofing including NUAA [37], CASIA-MFSD [44], and Replay-Attack [11] were released years ago. More recent MSU-USSA [32] and OULU-NPU datasets [10] do not include significant variations in poses, illuminations, and expressions. Current methods are trained/tested on specific datasets, but are not comprehensively justified in the complex real scenarios.

To make the research on face anti-spoofing more valuable for practical applications, in this paper we present an easy-to-execute solution to obtain a large amount of training data and build a model upon the data to push the limits of the face anti-spoofing performance. Specifically, we download positive samples (live face videos) from the Web, and collect negative samples by recording displayed videos on various digital devices. A novel spoof face synthesis method

*equal contributions
†correspondence authors

is proposed to further accelerate the data acquisition procedure. With the proposed solution, we are able to collect $5,000$ positive videos and $5,000$ negative videos within a month.

On the other hand, we build a model, dubbed Spatio-Temporal Anti-Spoofing Network (STASN), with a spatio-temporal attention mechanism. This model is composed of three modules, Temporal Anti-Spoofing Module (TASM), Region Attention Module (RAM), and Spatial Anti-Spoofing Module (SASM). In TASM, a CNN is used to learn powerful features and the employed LSTM encodes temporal information for video classification. From a spatial perspective, considering that the sole entire image fails to show a sufficient discriminating power, local subtle features are proven to be more useful [2, 25]. Different from previous random method [2] or fixed setting [25], we learn the discriminative region attention via a deep module RAM. This module can capture important local regions of spoof clues with a strong discriminating ability. Feature learning is carried out more effectively by attending these local regions in the so-called SASM. Moreover, local attention structure and LSTM features complement each other to endow different information into spatio-temporal incorporation.

The attention scheme in the proposed model allows us to explore a more intuitive representation which benefits humans in understanding how this problem can be tackled. We dive into the model to conduct a set of investigation studies, and show some interesting (intermediate) results, which would be useful for future research in the community.

Our main contributions are summarized as follows:

- We present an easy-to-execute solution to collect a large amount of data by mimicking the spoofing attacks in real world, which is demonstrated to be vital for the face anti-spoofing research.
- We propose a face anti-spoofing model with a spatio-temporal attention mechanism to fuse global temporal and local spatial information, which allows us to analyze the model's interpretable behaviors.
- We significantly advance the state-of-the-art performances on public face anti-spoofing datasets, thus providing the community a promising direction along with building powerful anti-spoofing solutions in practice.

2. Related Work

Traditional Methods. The difference in textures is one of the main clues to distinguish live faces from spoof faces [7]. Such information has been exploited for face anti-spoofing. For example, many hand-crafted features have been studied in previous works, including LBP [29, 12, 13], HOG [21, 43], DOG [37, 33], SIFT [32], and SURF [9]. In addition, different data domains have been exploited to extract discriminative features. Boulkenafet *et al.* investigated different color spaces such as HSV and YCbCr [6, 8]. Features in the frequency domain were also studied in [22]. The common issue existing in these methods is that these hand-crafted features are not robust to various nuisance variables in the wild, such as illuminations and occlusions.

In contrast to solely using a still image, researchers attempt to leverage spontaneous face motions in a sequence of frames for face anti-spoofing. For example, eyes blinking has been utilized to detect face liveness in [30, 36]. Kollreider *et al.* used mouth and lip motions for face anti-spoofing [20]. However, spontaneous face motions are often too subtle to be captured by hand-crafted features in practice.

Deep Learning Methods. The strong representation power of modern CNNs has been exploited [27] in face anti-spoofing research [16, 23, 31, 42]. The methods in [23, 31] used a pretrained CaffeNet or VGG-face model as a feature extractor to distinguish live and spoof faces. Multiple spatial scales have been leveraged in [38] to classify live and spoof images. Additional information, such as remote photoplethysmography (rPPG, a heart pulse signal) and spoof noise, has been exploited in [26, 18]. Despite the progress in their performance with regard to traditional methods, the detection accuracy and the robustness to the nuisance variables in the wild are still less satisfied for practical use.

Most recently, improvements have been achieved by simultaneously taking spatial and temporal aspects into account. Our work is most related to [41] in this category, where we both used an LSTM-CNN architecture on multiple frames of a video. In contrast to [41], we further explore local regions fusion and an attention mechanism to boost the performance and allow interpretable analysis.

3. Data

Sufficient data plays an important role in deriving satisfactory models. Typically the data used to train the models is expected to be sufficient and close to the testing data. However, the current data sets in the community are either of small scale or do not mimic the real-world testing data sufficiently well, which limits the potential of models trained on these datasets.

To this end, we collect a set of data ourselves and build models based on the collected data. Generally, live faces (not spoof) are not difficult to obtain. For example, there are a lot of selfie videos on the Web. Additionally, it is also easy to collect videos which are not selfie but include faces. We downloaded a set of positive samples consisting of $5,000$ live face videos using Python scripts. Then a 5$\sim$10 seconds clip is extracted from each video using the Dlib [19] face detector.

The difficulties lie in the acquisition of negative samples,

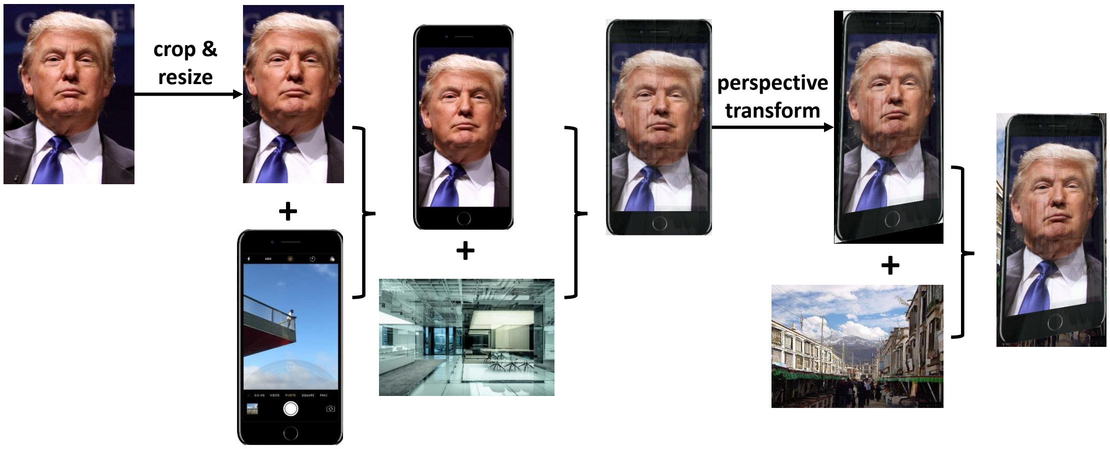

Figure 2. The procedure of synthesizing spoof faces. Please zoom in to check the synthetic reflection artifacts exhibited in the result.

i.e., spoof face videos. As it is known to all, spoof behaviour is rare in the real world, so the spoof face data is scarce and expensive. We adopt the following two methods to acquire negative samples.

3.1. Manually-Mimicked Spoof Faces

To mimic the spoof attacking procedure, we use various kinds of digital device to display positive face videos and different kinds of devices to record the displayed videos. Specifically, the devices displaying the normal videos include three models of iPhone (iPhone 6, iPhone 7, and iPhone X), over 10 models of typical Android phones of popular brands (Samsung, Huawei, Xiaomi, *etc.*), pads (iPad and an Android pad), and desktop/laptop screens. The mobile devices used to record videos include popular iPhone models and Android models. It is important to note that the human workers are required to try their best to mimic the attacking procedure when recording spoof videos. To be specific, they are asked to avoid the artifacts of reflection on the screen, moire pattern, the appearance of device edge/border, *etc.* This kind of careful handling would approximate the spoof videos in deliberated attacking. In this way, we collected $2,500$ negative samples (5$\sim$10 seconds videos) in less than a month with two human workers.

3.2. Machine-Synthesized Spoof Faces

The aforementioned procedure, elaborated by humans, is not sufficiently efficient. To tackle this, we propose a novel and effective method to synthesize a large number of negative samples based on the collected positive samples. We observe that the spoof face videos in the real world usually exhibit either low quality in the form of blurs, or reflections in the display screen plus perspective distortions. Thus we synthesize two sets of negative samples consisting of $2,500$ videos in total. Given a positive (image) sample $\mathbf{X}$, a Gaussian blur kernel G is applied to the image with random strength to blur the image. This procedure is formulated as $\hat{\mathbf{X}} = \mathbf{X} * G(\sigma)$, where σ is the strength of the blur drawn as a random variable.

On the other hand, the process to augment the positive sample with reflection plus distortion is as follows (also as shown in Fig. 2). 1) We firstly fit the positive sample into the screen of a device template (*e.g.*, an iPhone). The derived image, to mimic the display in a device, is denoted as $\mathbf{X}'$. 2) This image is blended with a random image as a reflection layer image, and we have a new image $\mathbf{X}'_r = (1-\alpha)\mathbf{X}' + \alpha\mathbf{X}_r$, where α is the strength variable of the content in the reflection layer image $\mathbf{X}_r$, randomly drawn from $[0, 0.2]$. 3) As the display device can hardly be posed strictly vertically in front of the live camera, we additionally apply a perspective transformation (with random parameters) $P(\cdot)$ to the image $\mathbf{X}'_r$, yielding a new image $\mathbf{X}'_{rd}$. There will be a mask $\mathbf{M}$ associated with the transformation. 4) We again blend the transformed image $\mathbf{X}'_{rd}$ with a random background image $\mathbf{X}_b$ from a collected image set, considering the mask $\mathbf{M}$ as, $\tilde{\mathbf{X}} = \mathbf{M} \odot \mathbf{X}'_{rd} + (\mathbf{1} - \mathbf{M}) \odot \mathbf{X}_b$, where $\mathbf{1}$ is a 2D matrix with the same resolution size of $\mathbf{X}'_{rd}$, and $\odot$ means the element-wise product between matrices.

4. Model

The input of the anti-spoofing task is a face video, so we propose to mine both the spatial and temporal cues to accomplish this task. Moreover, we discover that, besides the whole image which provides a complete view of the face, local subtle regions are more helpful for our verification task. In light of this, we develop a neural network model called Spatio-Temporal Anti-Spoofing Network (STASN) to learn more discriminative spatio-temporal features to complete our task. As shown in Fig. 3, this model is composed of three modules, Temporal Anti-Spoofing Module (TASM), Region Attention Module (RAM), and Spatial Anti-Spoofing Module (SASM). TASM addresses face anti-spoofing as a video classification problem, making decisions by mining temporal cues. RAM explores the location of potential subtle details and each patch of the attended

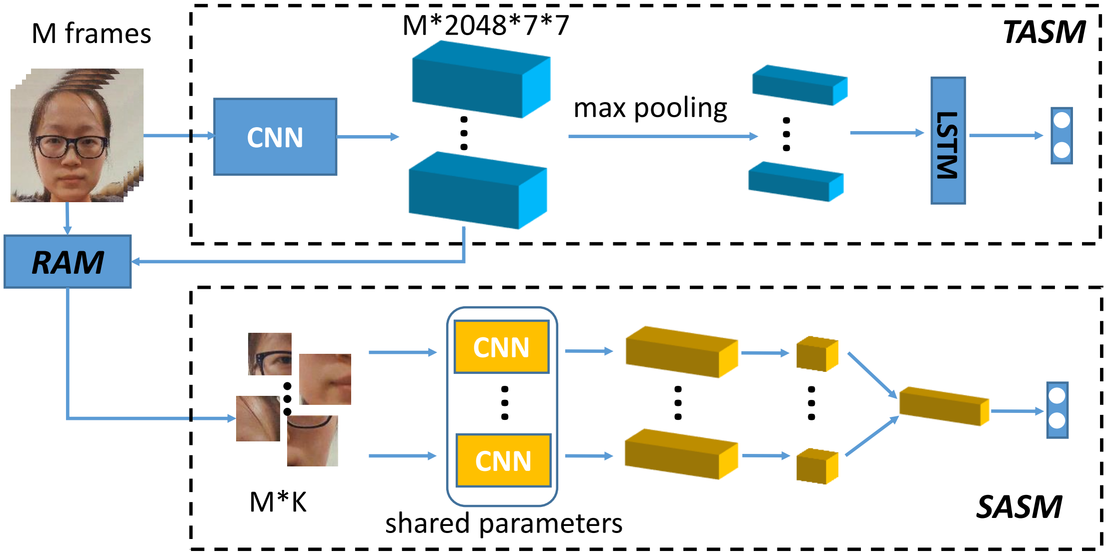

Figure 3. An overview of the STASN model. There are three components, Temporal Anti-Spoofing Module (TASM), Region Attention Module (RAM), and Spatial Anti-Spoofing Module (SASM), in the model. The TASM is a CNN-LSTM structure, taking the frame sequence as input and predicting binary classification results. The RAM learns the offset based on CNN features from TASM and outputs attended regions with respect to sequential images. The attended regions are forwarded in a parameter-shared CNN to give binary predictions.

location is fed into SASM to learn region representations.

4.1. Network

Let $\{(V_i, y_i)\}_{i=1}^N$ denote the set of training data, where V_i represents a training video and y_i denotes its label, 0 representing spoof video and 1 representing normal video. Each video consists of multiple frames, as $V_i = \{\mathbf{X}_{i,j}\}_{j=1}^M$, where $\mathbf{X}_{i,j}$ denotes the j-th frame in video V_i.

Typically, as ground-truth, only the class label is provided and there is no annotation of important regions. Thus we have to discover these regions for specified processing in the face anti-spoofing task. Unlike traditional strategies like random choice [2] or simple fusion of adding convolutional layer [24], we carefully design an attention mechanism, RAM, with fewer parameters and more reasonable initialization settings for locating the discriminative and significant sub-regions. The overview of our STASN is shown in Fig. 3. As described before, STASN includes three modules. Temporal Anti-Spoofing Module (TASM) aims to capture temporal dependencies among video frames. Spatial Anti-Spoofing Module (SASM) consists of K streams and each aims to learn subtle discriminative features. Region Attention Module (RAM) generates attention regions.

Temporal Anti-Spoofing Module (TASM). The TASM is a Conv-LSTM structure composed of a convolutional neural network to extract representative visual features and an LSTM module to encode temporal correlation across multiple frames. We use a 50-layer ResNet pre-trained on the ImageNet dataset as the visual feature extractor. This network is followed by a global average pooling layer. An LSTM module follows the pooling layer, extracting the temporal relationship from different video frames.

Region Attention Module (RAM). The region attention module aims to generate important local regions, which are fed into the SASM. Specifically, when the attended positions are located, we crop the corresponding regions to finer scale with higher resolution to extract subtle features. To ensure that the whole network can be optimized during training, we model this process by learning a transformation matrix as,

$$T = \begin{bmatrix} s_h & 0 & a_x \\ 0 & s_w & a_y \end{bmatrix}, \qquad (1)$$

which allows cropping and translation operations. We fix s_h and s_w as predefined constant values to set the region size, and output $2 \times K$ (K is the number of attended regions) parameters so as to locate individual regions within the image boundary. To achieve this goal, we develop a simple yet effective sub-network. As shown in Fig. 4, we take input as features from the *res_conv5* block. It is followed by a depth-wise convolutional layer of 7×7 filter. After that, we perform a 1×1 channel convolution operation and output $2 \times K$ parameters, indicating the offset/translation with regard to the anchor locations. This cross depthwise-channel structure is useful for learning spatial attention locations. The operation also reduces the computational complexity with merely $1/C$ (C is the number of channels) times of parameters needed by conventional convolution.

It is straightforward to initialize the anchor position of attended regions as the center of the image. However, we discover that the optimization of seeking the attention region position easily gets stuck in local overfitting during the

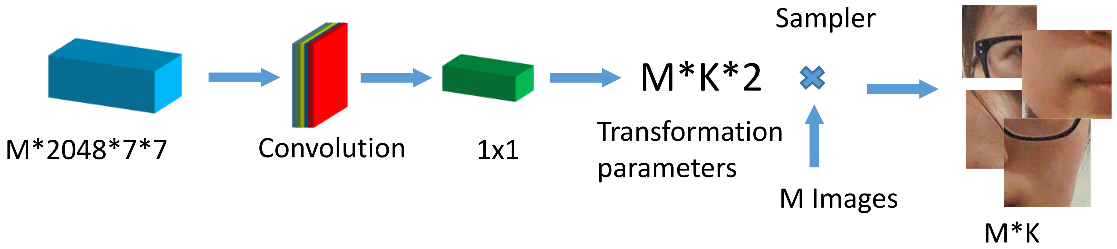

Figure 4. The region attention module. The input are features from the CNN in TASM. By applying depth-wise convolution and conventional convolution, transformation parameters are output. These parameters are referred by a sampler to give the attended local regions.

training stage. Therefore, a mechanism to produce a better initial localization of important regions is expected. We employ Grad-Gram [35] which produces a coarse localization map highlighting the important regions to initialize the position of attention regions. As the RAM structure learns the position offset, the attended regions can be obtained along with the better initialization.

Assuming that we have already the Temporal Anti-Spoofing Module (TASM), given an image $\mathbf{X}$ and its corresponding label y^c, we forward this image through the TASM and compute a raw score considering its label. This score signal is propagated back to the convolutional feature maps F_k of a convolutional layer. Then we derive the gradient of the score regarding F_k as $\frac{\partial y^c}{\partial F^k}$, and the gradient is manipulated by global pooling to obtain the importance weights α_k^c of the feature map F_k for a target class c,

$$\alpha_k^c = \frac{1}{z}\sum_i \sum_j \frac{\partial y^c}{\partial F_{ij}^k}\,, \tag{2}$$

where z is a normalization factor. We can perform a weighted combination of forward activation maps and formulate it as,

$$S_F = ReLU(\sum_k \alpha_k^c F^k). \tag{3}$$

We obtain a score map of the same size as the convolutional feature maps (7×7 in this case), and bilinear interpolation is applied at this score map to make it of the same size as the input image. Then we use the average pooling operation to derive a 4×4 score map, each value indicating the importance of one grid in the 4×4 grids. We select the largest K values, and let their corresponding region positions as the initial important regions in a frame. The RAM structure will learn the corresponding offsets a_x and a_y considering the initialization positions. By doing so, the final important local regions can be obtained.

Spatial Anti-Spoofing Module (SASM). As shown in Fig. 3, the Spatial Anti-Spoofing Module (SASM) is a multi-branch network structure. This module includes K streams of local-region branches. Each stream aims to learn the most discriminative features for one of the K local regions of a face image. To reduce the model parameters, we share convolutional layers among the multiple streams. Global max-pooling operation on the corresponding output map is conducted, and a 1×1 convolutional layer with batch normalization [17] and ReLU reduces the 2048-dim feature to 256-dim. Then we concatenate the number of K 256-dim features to classify spoof faces *versus* live faces.

During the testing phase, to obtain the most powerful discrimination, we combine both temporal and spatial scores to yield the final score.

4.2. Step-Wise Training

To better optimize attention localization and the task of classification, we develop a three-step training algorithm. In the *first* step, we initialize the CNN in TASM by using a ResNet network pre-trained on ImageNet. We then feed video data into TASM with 5 training epochs with the cross-entropy loss and derive a pre-trained temporal path model TASM. In the *second* step, we fix the pre-trained TASM, and train the RAM and SASM together. Specifically, we employ the TASM to obtain a response image of each frame by the Grad-Gram algorithm. Then we use the proposed region attention module to generate local regions, and feed these resized local patches into SASM with the cross-entropy loss. After 5 training epochs, we obtain the pre-trained RAM and SASM. In the *third* step, we train the whole model in another 5 epochs by decreasing the learning rate to $1/10$ of the previous rate to optimize the training performance.

5. Experiments

In this section, we conduct extensive experiments on the task of face anti-spoofing to demonstrate the effectiveness of our method and data. In the following, we sequentially describe the employed datasets & metrics (Sec. 5.1), implementation details (Sec. 5.2) and results (Sec. 5.3 - 5.5).

5.1. Datasets & Metrics

We evaluate the proposed model on four public face anti-spoofing databases, including Replay-Attack [11], CASIA-MFSD [44], Oulu-NPU [10] and the latest SiW [26]. CASIA-MFSD [44] contains 50 subjects, and 12 videos for each subject with 3 different resolutions and illumination conditions. Replay-Attack [11] includes $1,300$ live and

Table 1. The ablation study results on Oulu-NPU in terms of Protocols 1 and 3.

Prot.	Method	APCER(%)	BPCER(%)	ACER(%)
1	TASM	1.7	3.3	2.5
	STASN(w/o search)	2.1	2.5	2.3
	STASN	1.2	2.5	1.9
3	TASM	5.8±4.3	1.4±2.4	3.6±1.9
	STASN(w/o search)	5.4±3.8	1.1±1.3	3.3±1.9
	STASN	4.7±3.9	0.9±1.2	2.8±1.6

spoof videos from 50 subjects. These two datasets are used for cross testing. Oulu-NPU [10] consists of 990 real face videos and $3,960$ spoof face videos. There are four testing protocols associated with Oulu-NPU to evaluate the generalization of algorithms. Protocol 1 evaluates on the illumination variation, and Protocol 2 studies the impact of different types of spoof medium. Protocol 3 examines the effect of different camera devices and Protocol 4 investigates all the challenges above. The SiW dataset exhibits variations of different real-world factors. Three protocols are proposed with this dataset, concerning the model performance with regard to face pose and expression variations, cross spoof medium of replay attack, and cross types of attacking (*e.g.* from print attack to replay attack).

The performance metrics we employed are the Attack Presentation Classification Error Rate (APCER) [4], the Bona Fide Presentation Classification Error Rate (BPCER) [4] and Half Total Error Rate (HTER) [4]. The HTER is half of the sum of the False Rejection Rate (FRR) and the False Acceptance Rate (FAR). Besides, we use ACER = (APCER+BPCER)/2.

5.2. Implementation Details

The model is implemented with PyTorch framework. We use $K = 4$ as the number of regions and set $s_h = s_w = 0.25$. After face detection for each frame, we resize each face to the fixed size 224×224. We use Adam to optimize our proposed network with learning rate of $5e-5$ in the first and second step describe in Sec. 4.2. This learning rate is decreased to its $1/10$ magnitude in the third step of fine-tuning. The batch size of the temporal CNN-LSTM network is 10 and the number of frames M is 10.

We also use our collected data to further improve the performance. To be specific, we train a CNN with the same structure as the CNN in the TASM using our own data. We replace the CNN in TASM with the CNN module trained with our own data. A fine-tuning is carried out using the individual public dataset before evaluation. Namely, we use our own data only to learn powerful features as pre-training. We will discuss the significance of using our own data in the final performance later.

5.3. Ablation Study

Advantage of the attention mechanism. It is obvious that, the temporal path TASM could accomplish the task. The spatial path (SASM) along with the attention module RAM further mines the important local regions for the task. Thus by comparing the performance of sole TASM and the whole network will reveal the effectiveness of the employment of the local attended regions. We conduct ablation study using the Protocol 1 and 3 on the Oulu-NPU data set. Table 1 shows the comparison results. TASM indicates the sole temporal path, and STASN is the full method with both the temporal path and the attended spatial path. The STASN outperforms TASM with approximately 30% error reduction in terms of different metrics, indicating that by fusing local patches with attention mechanism the spoof faces can be more accurately classified.

Advantage of the search of initial regions. As mentioned in Sec. 4.1, initializing the local regions in the center of the image along with the learning of the offset is prone to get trapped in local optimum. Thus we employ the Grad-Gram method through the TASM to derive better initial position of the local regions and then learn the offset accordingly. This strategy of seeking initial region is compared with the naive initialization method (without search) in Table 1. As it is shown in Table 1, compared with the strategy without search, seeking better initial region position further reduces the error rates, suggesting its advantage.

5.4. Intra Testing

The intra testing is carried out on both the Oulu-NPU and the SiW datasets. We strictly follow the four testing protocols on Oulu-NPU and the three protocols on SiW for the evaluation. The metric values of APCER, BPCER and ACER are reported as quantitative results. A set of methods are adopted as counterparts to compare with, including CPqD [5], GRADIANT [5], MILHP [25], MixedFASNet [5], MassyHNU [5], Auxiliary [26] and FaceDs [18]. We train a model using the Oulu-NPU dataset, termed as “Ours” in Table 2. Additionally, as described in Sec. 5.2, we derive a model by using our collected data (Sec. 3) for pre-training, and fine-tune with the concerned dataset. This model is termed as “Ours+” in the table.

Table 2 reveals that, 1) compared with the state-of-the-art results, our method (without our data) achieves comparable result. It achieves three best values and two second best values. This suggests that our model works effectively in distinguishing spoof faces, without resorting to other sources of information. 2) With our own data for pre-training, our method beats all the compared methods, with obvious advantage. Especially with Protocol 4, which is the most difficult protocol on this dataset, the reduce of the error rates is significant (by comparing the best with the second best). It reveals the importance of the employment of our own data,

Table 2. The intra-testing results of four protocols on Oulu-NPU in terms of different metrics. For each metric, smaller value means better performance. The best performance is shown in bold, and the second best results are indicated by underline. This also applies to the following tables.

Prot.	Method	APCER(%)	BPCER(%)	ACER(%)
1	CPqD [5]	2.9	10.8	6.9
	GRADIANT [5]	1.3	12.5	6.9
	MILHP [25]	8.3	**0.8**	4.6
	Auxiliary [26]	1.6	1.6	1.6
	FaceDs [18]	**1.2**	1.7	1.5
	Ours	**1.2**	2.5	1.9
	Ours+	**1.2**	**0.8**	**1.0**
2	MixedFASNet [5]	9.7	2.5	6.1
	MILHP [25]	5.6	5.3	5.4
	FaceDs [18]	4.2	4.4	4.3
	Auxiliary [26]	2.7	2.7	2.7
	GRADIANT [5]	3.1	1.9	2.5
	Ours	4.2	**0.3**	2.2
	Ours+	**1.4**	0.8	**1.1**
3	MixedFASNet [5]	5.3±6.7	7.8±5.5	6.5±4.6
	MILHP [25]	1.5±1.2	6.4±6.6	4.0±2.9
	GRADIANT [5]	2.6±3.9	5.0±5.3	3.8±2.4
	FaceDs [18]	4.0±1.8	3.8±1.2	3.6±1.6
	Auxiliary [26]	2.7±1.3	3.1±1.7	2.9±1.5
	Ours	4.7±3.9	**0.9±1.2**	2.8±1.6
	Ours+	**1.4±1.4**	3.6±4.6	**2.5±2.2**
4	MassyHNU [5]	35.8±35.3	8.3±4.1	22.1±17.6
	MILHP [25]	15.8±12.8	8.3±15.7	12.0±6.2
	GRADIANT [5]	5.0±4.5	15.0±7.1	10.0 ±5.0
	Auxiliary [26]	9.3±5.6	10.4±6.0	9.5±6.0
	FaceDs [18]	1.2±6.3	6.1±5.1	5.6±5.7
	Ours	6.7±10.6	8.3±8.4	7.5±4.7
	Ours+	**0.9±1.8**	**4.2±5.3**	**2.6±2.8**

Table 3. The results on SiW regarding its three protocols.

Prot.	Method	ACER(%)
1	Auxiliary [26]	3.58
	Ours	1.00
	Ours+	**0.30**
2	Auxiliary [26]	0.57±0.69
	Ours	0.28±0.05
	Ours+	**0.15±0.05**
3	Auxiliary [26]	8.31±3.80
	Ours	12.10±1.50
	Ours+	**5.85±0.85**

consisting of both manually-mimicked data and machine-synthesized data.

For the latest SiW data, we also follow the three testing protocols strictly and report the results in Table 3. Regarding the first and second protocols, our method beats the state of the art [26] with a significant advantage. For the third protocol, the proposed model does not outperform it. However, with the collected useful data, our model achieves the best performance with regard to all protocols, showing the effectiveness of both our model and data.

Table 4. Cross testing comparison on the CASIA-MFSD dataset *versus* the Replay-Attack dataset in terms of HTER.

Method	Train	Test	Train	Test
	CASIA	Replay	Replay	CASIA
	MFSD	Attack	Attack	MFSD
Motion [13]	50.2%		47.9%	
LBP-TOP [13]	49.7%		60.6%	
Motion-Mag [3]	50.1%		47.0%	
Spectral cubes [34]	34.4%		50.0%	
LBP [6]	47.0%		39.6%	
Color Texture [8]	30.3%		37.7%	
CNN [42]	48.5%		45.5%	
Auxiliary [26]	27.6%		28.4%	
FaceDs [18]	28.5%		41.1%	
Ours	31.5%		30.9%	
Ours+	**18.7%**		**25.0%**	

5.5. Cross Testing

Cross testing aims to justify the generalization potential of the concerned model. To testify the generalization ability of our model, we also conduct this kind of cross testing. To make it more specific, two testing settings are applied. The first one is training model on the CASIA-MFSD dataset and testing on the Replay-Attack dataset. The second one is exchanging the training dataset and the testing dataset.

Results comparing with previous methods are shown in Table 4. Our method without our own data beats most of the traditional methods plus a CNN method. The performance of our model is comparable with the latest results of FaceDs [18], while slightly worse than those of Auxiliary [26]. As we have mentioned before, additional information like depth is employed to aid the classification, thus this model is expected to achieve better performance than ours. However, with our data, our model outperforms Auxiliary [26] and achieves the best performance. This verifies that the large amount of synthesized data indeed improves the generalization potential of the model in the case of cross testing.

6. Analysis

With the significant advance over the state-of-the-art performance, we are obliged to understand more behind the proposed model: is it robust enough to classify live and spoof faces? What cues is it looking for to make the decisions? Fortunately, with the attention mechanism in the proposed model, we are able to conduct visualization experiments and reveal more interesting findings as follows.

How does the model behave? Firstly, we aim to investigate the behaviors of the derived model. For example, what is the boundary of the two classes, *i.e.* spoof faces and live faces. To this end, we conduct feature dimension reduction of face features and plot them as scatter dots. We try to discover if there are any patterns in the scatter plots.

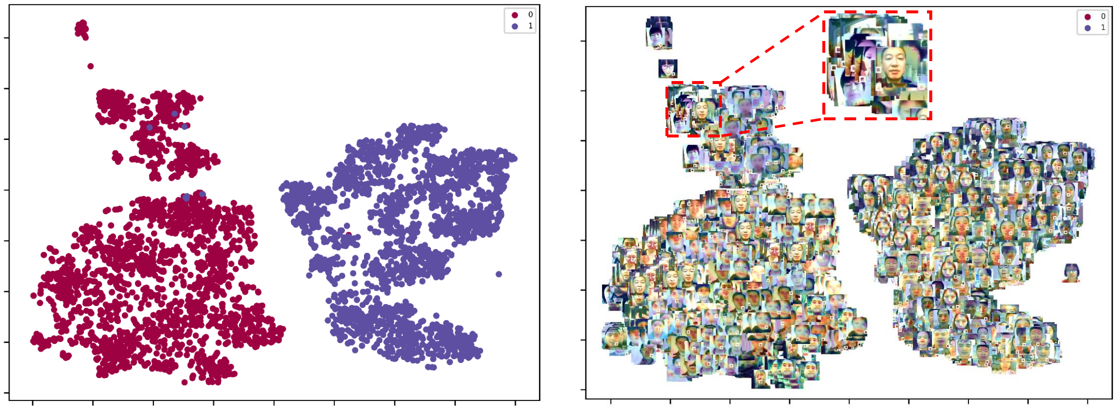

Figure 5. The 2D visualization of spoof faces and live faces. The dimension-reduced 2D face features are plotted in the left side. The corresponding faces with the selected most important region are shown in the right side. Please zoom in to check details. Best viewed in color.

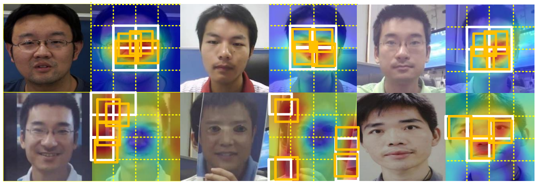

Figure 6. The attended local regions by RAM on live faces (top) and spoof faces (bottom), respectively. Please notice that the attended regions are consistently in the center on live face images, while are diverse on spoof face images, indicating cues such as borders. Best viewed in color.

To better visualize the faces, t-SNE [28] is adopted as the method of feature dimension reduction. Specifically, we use the output $Z \in \mathbb{R}^{2048\times 7\times 7}$ of the final convolutional layer in TASM as features. The Grad-Gram [35] is applied to that feature map Z to select the most important part $Z_p \in \mathbb{R}^{2048}$. Then t-SNE projects high-dimension features Z_p to two dimensions by best preserving the KL divergence distance. The left side of Fig. 5 shows the plotted results, whereas 0 means spoof faces and 1 indicates live faces. The right side shows the corresponding faces with the annotated receptive field (indicated by a white box) of the selected most important region for dimension reduction. By observing this figure, we have the following findings. 1) Though there are few faces plotted together with different labels, in general the faces are clearly separated. 2) Moreover, it is not difficult to find that there are clusters in each class. We further find that these clusters exhibit similar cues. This finding is more evident in spoof faces. For instance, as shown in the zoomed-in part in Fig. 5, faces with evident display device border are prone to be plotted together. 3) If we zoom in to check the live faces, it is interesting that the selected most important region of the close faces is also close spatially in the image space. This proves the consistency of the model, *i.e.* similar faces have similar important local regions. Overall, the faces of important areas are well classified in our task, so the additional branch of region classification with attention is important, and not affected by the overall spatial distributions.

What kind of regions are attended? We are also curious about the attended regions and why these regions are attended. Fig. 6 shows the located four discriminative regions by the Region Attention Module (RAM). Top and bottom rows show the live faces and spoof faces, respectively.

The white boxes represent the position of initial regions and the yellow boxes are the positions output by RAM. Obviously, live faces are attended more with the regions near the tip of the nose. However, for spoof faces, RAM is more inclined to capture a variety of other clues, such as borders, moire patterns, reflection artifacts, *etc*. This is consistent with human perceptions, as human also rely on such a kind of clues to make the decisions.

7. Conclusions

In this paper, we proposed a practical solution to build a powerful and robust face anti-spoofing model. This model, namely Spatio-Temporal Anti-Spoofing Network (STASN), considers both global temporal and local spatial cues to distinguish live faces *versus* spoof faces. Specifically, STASN was trained on a large amount of data collected using the proposed data acquisition methods. The performances of our model on public face anti-spoofing datasets demonstrate its superiority over the state of the art. Our study shows that the research of face anti-spoofing on a large amount of training data is more practical for real-world applications, as models trained on datasets which do not simulate the real-world data well may be of less significance and impact in practice.

Acknowledgements

Xiao Yang is an intern with Tencent AI Lab while doing this work. This work was supported by the NSFC (No. 61671289 and No. 61571261) and STCSM Grant 15DZ2270400.

References

[1] A. Agarwal, R. Singh, and M. Vatsa. Face anti-spoofing using haralick features. In *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–6, Sep. 2016. 1

[2] Yousef Atoum, Yaojie Liu, Amin Jourabloo, and Xiaoming Liu. Face anti-spoofing using patch and depth-based cnns. In *International Joint Conference on Biometrics (IJCB)*, pages 319–328, 2017. 1, 2, 4

[3] Samarth Bharadwaj, Tejas I Dhamecha, Mayank Vatsa, and Richa Singh. Computationally efficient face spoofing detection with motion magnification. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 105–110, 2013. 7

[4] ISO/IEC JTC 1/SC 37 Biometrics. *Information technology – Biometric presentation attack detection – Part 1: Framework*. International Organization for Standardization, 2016. 6

[5] Zinelabdine Boulkenafet, Jukka Komulainen, Zahid Akhtar, Azeddine Benlamoudi, Djamel Samai, Salah Eddine Bekhouche, Abdelkrim Ouafi, Fadi Dornaika, Abdelmalik Taleb-Ahmed, Le Qin, et al. A competition on generalized software-based face presentation attack detection in mobile scenarios. In *International Joint Conference on Biometrics (IJCB)*, pages 688–696, 2017. 6, 7

[6] Zinelabidine Boulkenafet, Jukka Komulainen, and Abdenour Hadid. Face anti-spoofing based on color texture analysis. In *International Conference on Image Processing (ICIP)*, pages 2636–2640, 2015. 2, 7

[7] Z. Boulkenafet, J. Komulainen, and A. Hadid. Face spoofing detection using colour texture analysis. *IEEE Transactions on Information Forensics and Security*, 11(8):1818–1830, Aug 2016. 2

[8] Zinelabidine Boulkenafet, Jukka Komulainen, and Abdenour Hadid. Face spoofing detection using colour texture analysis. *IEEE Transactions on Information Forensics and Security*, 11(8):1818–1830, 2016. 2, 7

[9] Zinelabidine Boulkenafet, Jukka Komulainen, and Abdenour Hadid. Face antispoofing using speeded-up robust features and fisher vector encoding. *Signal Processing Letters*, 24(2):141–145, 2017. 2

[10] Zinelabinde Boulkenafet, Jukka Komulainen, Lei Li, Xiaoyi Feng, and Abdenour Hadid. Oulu-npu: A mobile face presentation attack database with real-world variations. In *International Conference on Automatic Face & Gesture Recognition (FG)*, pages 612–618, 2017. 1, 5, 6

[11] Ivana Chingovska, André Anjos, and Sébastien Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *Proceedings of the 11th International Conference of the Biometrics Special Interes Group*, number EPFL-CONF-192369, 2012. 1, 5

[12] Tiago de Freitas Pereira, André Anjos, José Mario De Martino, and Sébastien Marcel. Lbp- top based countermeasure against face spoofing attacks. In *Asian Conference on Computer Vision (ACCV)*, pages 121–132. Springer, 2012. 2

[13] Tiago de Freitas Pereira, André Anjos, José Mario De Martino, and Sébastien Marcel. Can face anti-spoofing countermeasures work in a real world scenario? In *International Conference on Biometrics (ICB)*, pages 1–8, 2013. 2, 7

[14] M. De Marsico, M. Nappi, D. Riccio, and J. Dugelay. Moving face spoofing detection via 3d projective invariants. In *2012 5th IAPR International Conference on Biometrics (ICB)*, pages 73–78, March 2012. 1

[15] N. Erdogmus and S. Marcel. Spoofing face recognition with 3d masks. *IEEE Transactions on Information Forensics and Security*, 9(7):1084–1097, July 2014. 1

[16] Litong Feng, Lai-Man Po, Yuming Li, Xuyuan Xu, Fang Yuan, Terence Chun-Ho Cheung, and Kwok-Wai Cheung. Integration of image quality and motion cues for face anti-spoofing: A neural network approach. *Journal of Visual Communication and Image Representation*, 38:451–460, 2016. 2

[17] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, pages 448–456, 2015. 5

[18] Amin Jourabloo, Yaojie Liu, and Xiaoming Liu. Face despoofing: Anti-spoofing via noise modeling. *European Conference on Computer Vision (ECCV)*, 2018. 2, 6, 7

[19] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009. 2

[20] Klaus Kollreider, Hartwig Fronthaler, Maycel Isaac Faraj, and Josef Bigun. Real-time face detection and motion analysis with application in liveness assessment. *IEEE Transactions on Information Forensics and Security*, 2(3):548–558, 2007. 2

[21] Jukka Komulainen, Abdenour Hadid, and Matti Pietikainen. Context based face anti-spoofing. In *IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems*, pages 1–8, 2013. 2

[22] Jiangwei Li, Yunhong Wang, Tieniu Tan, and Anil K Jain. Live face detection based on the analysis of fourier spectra. In *Biometric Technology for Human Identification*, volume 5404, pages 296–304. International Society for Optics and Photonics, 2004. 2

[23] Lei Li, Xiaoyi Feng, Zinelabidine Boulkenafet, Zhaoqiang Xia, Mingming Li, and Abdenour Hadid. An original face anti-spoofing approach using partial convolutional neural network. In *International Conference on Image processing theory tools and applications (IPTA)*, pages 1–6, 2016. 2

[24] Wei Li, Xiatian Zhu, and Shaogang Gong. Harmonious attention network for person re-identification. 1:2, 2018. 4

[25] Chen Lin, Zhouyingcheng Liao, Peng Zhou, Jianguo Hu, and Bingbing Ni. Live face verification with multiple instantialized local homographic parameterization. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, pages 814–820, 2018. 2, 6, 7

[26] Yaojie Liu, Amin Jourabloo, and Xiaoming Liu. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 389–398, 2018. 1, 2, 5, 6, 7

[27] Wenhan Luo, Peng Sun, Fangwei Zhong, Wei Liu, Tong Zhang, and Yizhou Wang. End-to-end active object track-

ing and its real-world deployment via reinforcement learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019. 2

[28] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. 8

[29] Jukka Määttä, Abdenour Hadid, and Matti Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *International Joint Conference on Biometrics (IJCB)*, pages 1–7, 2011. 2

[30] Gang Pan, Lin Sun, Zhaohui Wu, and Shihong Lao. Eyeblink-based anti-spoofing in face recognition from a generic web camera. 2007. 2

[31] Keyurkumar Patel, Hu Han, and Anil K Jain. Cross-database face antispoofing with robust feature representation. In *Chinese Conference on Biometric Recognition*, pages 611–619. Springer, 2016. 2

[32] Keyurkumar Patel, Hu Han, and Anil K Jain. Secure face unlock: Spoof detection on smartphones. *IEEE Transactions on Information Forensics and Security*, 11(10):2268–2283, 2016. 1, 2

[33] Bruno Peixoto, Carolina Michelassi, and Anderson Rocha. Face liveness detection under bad illumination conditions. In *IEEE International Conference on Image Processing (ICIP)*, pages 3557–3560, 2011. 2

[34] Allan Pinto, Helio Pedrini, William Robson Schwartz, and Anderson Rocha. Face spoofing detection through visual codebooks of spectral temporal cubes. *IEEE Transactions on Image Processing*, 24(12):4726–4740, 2015. 7

[35] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, Dhruv Batra, et al. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *International Conference on Computer Vision (ICCV)*, pages 618–626, 2017. 5, 8

[36] Lin Sun, Gang Pan, Zhaohui Wu, and Shihong Lao. Blinking-based live face detection using conditional random fields. In *International Conference on Biometrics (ICB)*, pages 252–260. Springer, 2007. 2

[37] Xiaoyang Tan, Yi Li, Jun Liu, and Lin Jiang. Face liveness detection from a single image with sparse low rank bilinear discriminative model. In *European Conference on Computer Vision (ECCV)*, pages 504–517. Springer, 2010. 1, 2

[38] Sergey Tulyakov, Xavier Alameda-Pineda, Elisa Ricci, Lijun Yin, Jeffrey F Cohn, and Nicu Sebe. Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2396–2404, 2016. 2

[39] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1

[40] D. Wen, H. Han, and A. K. Jain. Face spoof detection with image distortion analysis. *IEEE Transactions on Information Forensics and Security*, 10(4):746–761, April 2015. 1

[41] Zhenqi Xu, Shan Li, and Weihong Deng. Learning temporal features using lstm-cnn architecture for face anti-spoofing. In *Asian Conference on Pattern Recognition (ACPR)*, pages 141–145, 2015. 2

[42] Jianwei Yang, Zhen Lei, and Stan Z Li. Learn convolutional neural network for face anti-spoofing. *arXiv preprint arXiv:1408.5601*, 2014. 2, 7

[43] Jianwei Yang, Zhen Lei, Shengcai Liao, and Stan Z Li. Face liveness detection with component dependent descriptor. *International Conference on Biometrics (ICB)*, 1:2, 2013. 2

[44] Zhiwei Zhang, Junjie Yan, Sifei Liu, Zhen Lei, Dong Yi, and Stan Z Li. A face antispoofing database with diverse attacks. In *International Conference on Biometrics (ICB)*, pages 26–31, 2012. 1, 5