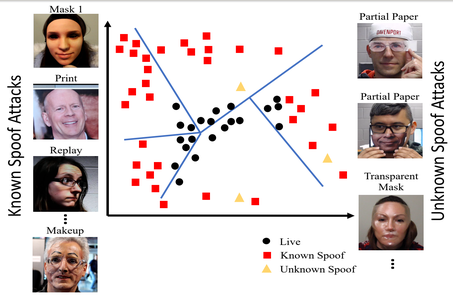
Deep Tree Learning for Zero-Shot Face Anti-Spoofing

Yaojie Liu, Joel Stehouwer, Amin Jourabloo, Xiaoming Liu
Department of Computer Science and Engineering, Michigan State University, MI

CVPR JUNE 16-21, 2019 LONG BEACH • CALIFORNIA

Introduction



- Empower the machine to detect **unknown/unseen** attacks
- Enlarge the study from 2 types to **13 types**
- Collect the first database for Zero-shot Face Anti-spoofing

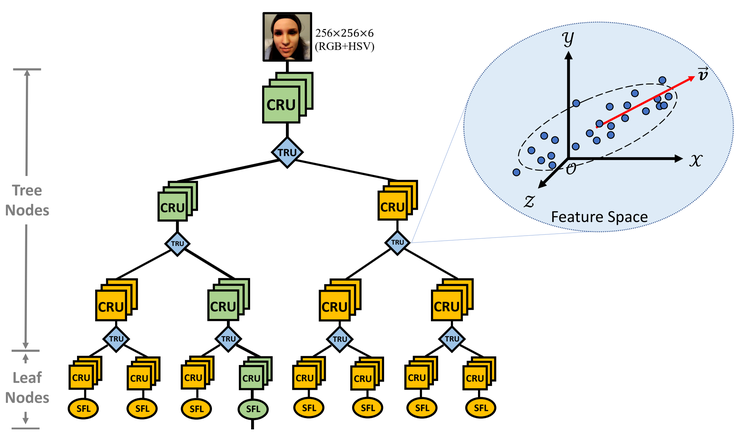
Overall Architecture

➢ Motivations?

- Different spoof attacks require different features/cues to distinguish
- Leverage the knowledge from known attacks

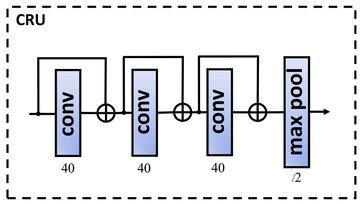
➢ Deep Tree Network

- Learn homogenous features in early stage/ distinct features in later stage
- Unsupervised to build the tree: splitting on the direction of largest data variation
- Supervised to learn the features at the leaf node
- End-to-end training



Convolutional Residual Unit (CRU)

- 3X3 convolution layer w/ 40 channels
- residual connections
- max pooling



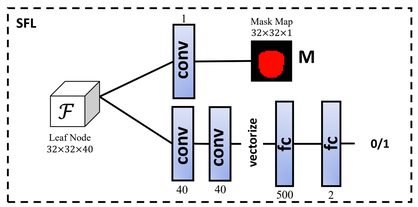
Supervised Feature Learning (SFL)

➢ Classification Supervision

$$\mathcal{L}_{class} = \frac{1}{N}\sum_{I_k \in \mathcal{S}} \Big\{ -y_k \log p_k + (1-y_k)\log(1-p_k) \Big\}$$

➢ Pixel-wise Supervision

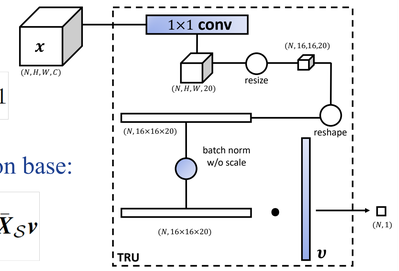
$$\mathcal{L}_{mask} = \frac{1}{N}\sum_{I_k \in \mathcal{S}} \|\mathbf{M}_k - \mathbf{D}_k\|_1$$



Tree Routing Unit (TRU)

➢ Tree routing function:

$$\varphi(\boldsymbol{x}) = (\boldsymbol{x}-\boldsymbol{\mu})^T \cdot \boldsymbol{v}, \quad \|\boldsymbol{v}\| = 1$$



➢ To make the v to be the largest projection base:

$$\arg\max_{\boldsymbol{v},\theta} \lambda = \arg\max_{\boldsymbol{v},\theta} \boldsymbol{v}^T \bar{\boldsymbol{X}}_{\mathcal{S}}^T \bar{\boldsymbol{X}}_{\mathcal{S}} \boldsymbol{v}$$

➢ Tree routing loss:

- build the X_S using the spoof data only

$$\mathcal{L}_{route} = \exp(-\alpha \boldsymbol{v}^T \bar{\boldsymbol{X}}_{\mathcal{S}}^T \bar{\boldsymbol{X}}_{\mathcal{S}} \boldsymbol{v}) + \beta \mathrm{Tr}(\bar{\boldsymbol{X}}_{\mathcal{S}}^T \bar{\boldsymbol{X}}_{\mathcal{S}})$$

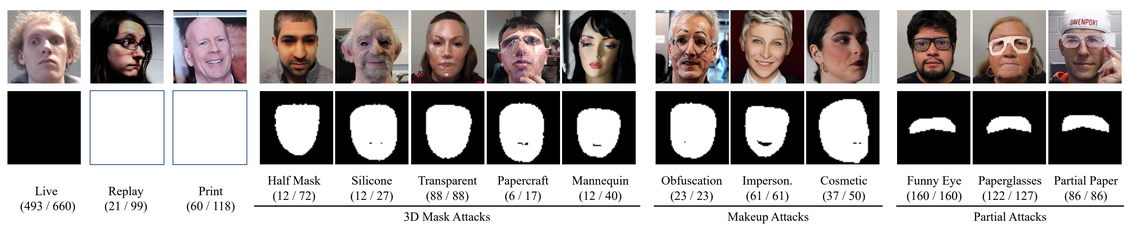
➢ Tree unique loss:

- balance the training data for each split

$$\mathcal{L}_{uniq} = -\frac{1}{N}\sum_{\mathbf{I}_k \in \mathcal{S}} \left\|\bar{\boldsymbol{x}}_k^T \boldsymbol{v}\right\|^2 + \frac{1}{N^-}\sum_{\mathbf{I}_k \in \mathcal{S}^-} \left\|\bar{\boldsymbol{x}}_k^T \boldsymbol{v}\right\|^2$$

Data and Experimental Results

➢ Stats of the Spoof in the Wild II database for ZSFA

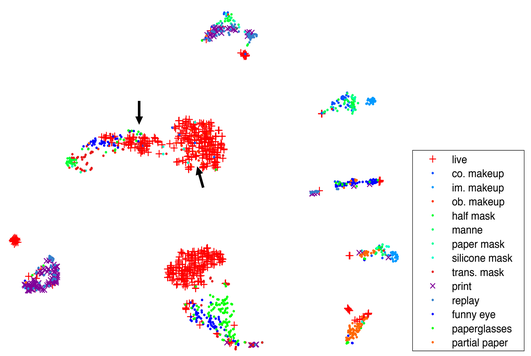


Live (493 / 660) | Replay (21 / 99) | Print (60 / 118) | Half Mask (12 / 72) | Silicone (12 / 27) | Transparent (88 / 88) | Papercraft (6 / 17) | Mannequin (12 / 40) | Obfuscation (23 / 23) | Imperson. (61 / 61) | Cosmetic (37 / 50) | Funny Eye (160 / 160) | Paperglasses (122 / 127) | Partial Paper (86 / 86)

3D Mask Attacks | Makeup Attacks | Partial Attacks

➢ Leave-one-out testing protocols and performance comparison

Methods	Metrics (%)	Replay	Print	Mask Attacks					Makeup Attacks			Partial Attacks			Average
				Half	Silicone	Trans.	Paper	Manne.	Obfusc.	Imperson.	Cosmetic	Funny Eye	Paper Glasses	Partial Paper	
SVM$_{RBF}$+LBP [9]	APCER	19.1	15.4	40.8	20.3	70.3	**0.0**	4.6	96.9	35.3	**11.3**	53.3	58.5	0.6	32.8 ± 29.8
	BPCER	22.1	21.5	21.9	21.4	20.7	23.1	22.9	21.7	12.5	22.2	18.4	20.0	22.9	21.0 ± 2.9
	ACER	20.6	18.4	31.3	21.4	45.5	11.6	13.8	59.3	23.9	16.7	35.9	39.2	11.7	26.9 ± 14.5
	EER	20.8	18.6	36.3	21.4	37.2	7.5	14.1	51.2	19.8	16.1	34.4	33.0	7.9	24.5 ± 12.9
Auxiliary [33]	APCER	23.7	7.3	27.7	**18.2**	97.8	8.3	16.2	100.0	18.0	16.3	91.8	72.2	0.4	38.3 ± 37.4
	BPCER	**10.1**	**6.5**	**10.9**	**11.6**	**6.2**	**7.8**	**9.3**	11.6	**9.3**	**7.1**	**6.2**	**8.8**	**10.3**	$\mathbf{8.9 \pm 2.0}$
	ACER	16.8	6.9	19.3	**14.9**	52.1	8.0	12.8	55.8	13.7	**11.7**	49.0	40.5	**5.3**	23.6 ± 18.5
	EER	14.0	4.3	**11.6**	**12.4**	**24.6**	7.8	10.0	72.3	10.1	**9.4**	21.4	**18.6**	**4.0**	17.0 ± 17.7
Ours	APCER	**1.0**	**0.0**	**0.7**	24.5	**58.6**	0.5	**3.8**	**73.2**	**13.2**	12.4	**17.0**	**17.0**	**0.2**	$\mathbf{17.1 \pm 23.3}$
	BPCER	18.6	11.9	29.3	12.8	13.4	8.5	23.0	**11.5**	9.6	16.0	21.5	22.6	16.8	16.6 ± 6.2
	ACER	**9.8**	**6.0**	**15.0**	18.7	**36.0**	**4.5**	**7.7**	**48.1**	**11.4**	14.2	**19.3**	**19.8**	8.5	$\mathbf{16.8 \pm 11.1}$
	EER	**10.0**	**2.1**	14.4	18.6	26.5	**5.7**	**9.6**	**50.2**	**10.1**	13.2	**19.8**	20.5	8.8	$\mathbf{16.1 \pm 12.2}$



➢ Find more details and source code: