Source-Free Domain Adaptation with Contrastive Domain Alignment and Self-supervised Exploration for Face Anti-Spoofing: Supplementary Material

Yuchen Liu[1†], Yabo Chen[2†], Wenrui Dai[2], Mengran Gou[3], Chun-Ting Huang[3], and Hongkai Xiong[1]

[1]Department of Electronic Engineering, Shanghai Jiao Tong University
[2]Department of Computer Science and Engineering, Shanghai Jiao Tong University
{liuyuchen6666, chenyabo, daiwenrui, xionghongkai}@sjtu.edu.cn
[3]Qualcomm AI Research {mgou, chunting}@qti.qualcomm.com

A. Proof of Proposition 1

Proposition 1. *Given a trained model $f_s = h_s \circ g_s$, where g_s is the feature extractor and h_s is the one-layer classifier, the ℓ_2-normalized weight vectors $\{\mathbf{w}_s^{real}, \mathbf{w}_s^{fake}\}$ of the classifier are the equivalent representation of the feature embeddings $\{\mathbf{z}_s^{real}, \mathbf{z}_s^{fake}\}$ of the source prototypes for calculating the supervised contrastive loss.*

Proof. Given the input data $\{\mathbf{x}_S^{real}, \mathbf{x}_S^{fake}\}$, we obtain the ℓ_2-normalized feature embeddings $\{\mathbf{z}_s^{real}, \mathbf{z}_s^{fake}\}$ as $\mathbf{z}_s^{real} = g_s(\mathbf{x}_S^{real})$ and $\mathbf{z}_s^{fake} = g_s(\mathbf{x}_S^{fake})$. $\mathbf{z}_s^{real}$ and $\mathbf{z}_s^{fake}$ are then fed into the one-layer classifier h_s to produce $[\mathbf{z}_s^{real} \cdot \mathbf{w}_s^{real}, \mathbf{z}_s^{real} \cdot \mathbf{w}_s^{fake}]$ and $[\mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{real}, \mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{fake}]$, respectively. Here, $\cdot$ represents the inner product of two vectors. The BCE loss is formulated as

$$\begin{aligned}\mathcal{L}_{\mathrm{BCE}} = &-\log \frac{\exp(\mathbf{z}_s^{real} \cdot \mathbf{w}_s^{real})}{\exp(\mathbf{z}_s^{real} \cdot \mathbf{w}_s^{real}) + \exp(\mathbf{z}_s^{real} \cdot \mathbf{w}_s^{fake})} \\ &-\log \frac{\exp(\mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{fake})}{\exp(\mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{real}) + \exp(\mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{fake})}.\end{aligned} \tag{S-1}$$

When the training on the source data converges, $\mathcal{L}_{\mathrm{BCE}}$ approaches the minimum. We assume that

$$\begin{cases} -\log \dfrac{\exp(\mathbf{z}_s^{real} \cdot \mathbf{w}_s^{real})}{\exp(\mathbf{z}_s^{real} \cdot \mathbf{w}_s^{real}) + \exp(\mathbf{z}_s^{real} \cdot \mathbf{w}_s^{fake})} < \epsilon_1 & \text{(S-2a)} \\ -\log \dfrac{\exp(\mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{fake})}{\exp(\mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{real}) + \exp(\mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{fake})} < \epsilon_2 & \text{(S-2b)} \end{cases}$$

Equations (S-2a) and (S-2b) can be rewritten as

$$\begin{cases} \log[1 + \exp(\mathbf{z}_s^{real} \cdot \mathbf{w}_s^{fake} - \mathbf{z}_s^{real} \cdot \mathbf{w}_s^{real})] < \epsilon_1 & \text{(S-3a)} \\ \log[1 + \exp(\mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{real} - \mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{fake})] < \epsilon_2 & \text{(S-3b)} \end{cases}$$

Note that $\mathbf{z}_s^{real}$, $\mathbf{z}_s^{fake}$, $\mathbf{w}_s^{real}$, *and* $\mathbf{w}_s^{fake}$ *have unit* ℓ_2*-norm. We obtain* $|\mathbf{z}_s^{\kappa}| = 1$, $|\mathbf{w}_s^{\kappa}| = 1$ *and* $\mathbf{z}_s^{\kappa} \cdot \mathbf{w}_s^{\kappa} \in [-1, 1]$. *Here* $|\cdot|$ *denotes the* ℓ_2*-norm and* κ *indicates real or fake. As the training on the source data converges,* $\epsilon_1 \to \log(1 + e^{-2})$ *and* $\epsilon_2 \to \log(1 + e^{-2})$. *From Equations* (S-3a) *and* (S-3b),

$$\begin{cases} \mathbf{z}_s^{real} \cdot \mathbf{w}_s^{fake} - \mathbf{z}_s^{real} \cdot \mathbf{w}_s^{real} < \log(e^{\epsilon_1} - 1) & \text{(S-4a)} \\ \mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{real} - \mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{fake} < \log(e^{\epsilon_2} - 1) & \text{(S-4b)} \end{cases}$$

Equivalently, we obtain that

$$\begin{cases} -1 - \log(e^{\epsilon_1} - 1) \leq \mathbf{z}_s^{real} \cdot \mathbf{w}_s^{real} \leq 1 & \text{(S-5a)} \\ -1 - \log(e^{\epsilon_2} - 1) \leq \mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{fake} \leq 1 & \text{(S-5b)} \end{cases}$$

Without loss of generality, we calculate the supervised contrastive loss for the real face in the target domain (with the weight vector of the classifier or the feature vector of the source data).

$$\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s) = -\log \frac{\exp(\mathbf{z}_t^{real} \cdot \mathbf{z}_s^{real}/\tau)}{\exp(\mathbf{z}_t^{real} \cdot \mathbf{z}_s^{real}/\tau) + \exp(\mathbf{z}_t^{real} \cdot \mathbf{z}_s^{fake}/\tau)}, \tag{S-6}$$

and

$$\mathcal{L'}_{\mathrm{SCL}}(\mathbf{w}_s) = -\log \frac{\exp(\mathbf{z}_t^{real} \cdot \mathbf{w}_s^{real}/\tau)}{\exp(\mathbf{z}_t^{real} \cdot \mathbf{w}_s^{real}/\tau) + \exp(\mathbf{z}_t^{real} \cdot \mathbf{w}_s^{fake}/\tau)}, \tag{S-7}$$

where $\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)$ *is the supervised loss associated with the feature vector of source data and* $\mathcal{L'}_{\mathrm{SCL}}(\mathbf{w}_s)$ *is the supervised loss associated with the weight vector of classifier. In the rest of this section, we prove that i)* $\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)$ *is equivalent to* $\mathcal{L'}_{\mathrm{SCL}}(\mathbf{w}_s)$, *ii)* $\nabla_{\mathbf{z}_t}\mathcal{L'}_{\mathrm{SCL}}(\mathbf{w}_s)$ *and* $\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)$ *have the same direction, and iii)* $|\nabla_{\mathbf{z}_t}\mathcal{L'}_{\mathrm{SCL}}(\mathbf{w}_s)|$ *is equivalent to* $|\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)|$.

i) $\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)$ ***is equivalent to*** $\mathcal{L'}_{\mathrm{SCL}}(\mathbf{w}_s)$***.***

According to Equations (S-6) *and* (S-7), *we consider* $|\mathbf{z}_t^{real} \cdot \mathbf{z}_s^{real} - \mathbf{z}_t^{real} \cdot \mathbf{w}_s^{real}|$ *and* $|\mathbf{z}_t^{real} \cdot \mathbf{z}_s^{fake} - \mathbf{z}_t^{real} \cdot \mathbf{w}_s^{fake}|$ *to compare* $\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)$ *and* $\mathcal{L'}_{\mathrm{SCL}}(\mathbf{w}_s)$.

$$|\mathbf{z}_t^{real} \cdot \mathbf{z}_s^{real} - \mathbf{z}_t^{real} \cdot \mathbf{w}_s^{real}| \leq |\mathbf{z}_t^{real}| \cdot |\mathbf{z}_s^{real} - \mathbf{w}_s^{real}| = |\mathbf{z}_s^{real} - \mathbf{w}_s^{real}|, \tag{S-8}$$

and

$$|\mathbf{z}_t^{real} \cdot \mathbf{z}_s^{fake} - \mathbf{z}_t^{real} \cdot \mathbf{w}_s^{fake}| \leq |\mathbf{z}_t^{real}||\mathbf{z}_s^{fake} - \mathbf{w}_s^{fake}| = |\mathbf{z}_s^{fake} - \mathbf{w}_s^{fake}|. \tag{S-9}$$

From Equations (S-5a) *and* (S-5b), *we obtain that*

$$|\mathbf{z}_s^{real} - \mathbf{w}_s^{real}|^2 = 2 - 2 \cdot \mathbf{z}_s^{real} \cdot \mathbf{w}_s^{real} \leq 4 + 2\log(e^{\epsilon_1} - 1), \tag{S-10}$$

and

$$|\mathbf{z}_s^{fake} - \mathbf{w}_s^{fake}|^2 = 2 - 2 \cdot \mathbf{z}_s^{fake} \cdot \mathbf{w}_s^{fake} \leq 4 + 2\log(e^{\epsilon_2} - 1). \tag{S-11}$$

Let us define

$$\begin{cases} \gamma_1 \triangleq \sqrt{4 + 2\log(e^{\epsilon_1} - 1)} \geq |\mathbf{z}_s^{real} - \mathbf{w}_s^{real}| & \text{(S-12a)} \\ \gamma_2 \triangleq \sqrt{4 + 2\log(e^{\epsilon_2} - 1)} \geq |\mathbf{z}_s^{fake} - \mathbf{w}_s^{fake}| & \text{(S-12b)} \end{cases}$$

For simplicity, we denote $A \triangleq \mathbf{z}_t^{real} \cdot \mathbf{z}_s^{fake}$, $B \triangleq \mathbf{z}_t^{real} \cdot \mathbf{z}_s^{real}$, $A' \triangleq \mathbf{z}_t^{real} \cdot \mathbf{w}_s^{fake}$, *and* $B' \triangleq \mathbf{z}_t^{real} \cdot \mathbf{w}_s^{real}$. *Equations* (S-6) *and* (S-7) *can be rewritten as*

$$\begin{cases} \mathcal{L}_{\text{SCL}}(\mathbf{z}_s) = \log[1 + e^{(A-B)/\tau}] & \text{(S-13a)} \\ \mathcal{L'}_{\text{SCL}}(\mathbf{w}_s) = \log[1 + e^{(A'-B')}/\tau] & \text{(S-13b)} \end{cases}$$

From Equations (S-8)–(S-12b), *we obtain that*

$$|A - A'| \leq \gamma_2, \quad |B - B'| \leq \gamma_1. \tag{S-14}$$

Consequently, we have

$$\begin{cases} A - \gamma_2 \leq A' \leq A + \gamma_2 & \text{(S-15a)} \\ B - \gamma_1 \leq B' \leq B + \gamma_1 & \text{(S-15b)} \end{cases}$$

Therefore,

$$A - B - \gamma_1 - \gamma_2 \leq A' - B' \leq A - B + \gamma_1 + \gamma_2 \tag{S-16}$$

Considering that $f(x) = \log(1 + e^{\frac{x}{\tau}})$ *is monotonically increasing for* $\tau > 0$, *we have*

$$\log\left(1 + e^{\frac{A-B}{\tau}} e^{\frac{-\gamma_1-\gamma_2}{\tau}}\right) \leq \log\left(1 + e^{\frac{A'-B'}{\tau}}\right) \leq \log\left(1 + e^{\frac{A-B}{\tau}} e^{\frac{\gamma_1+\gamma_2}{\tau}}\right) \tag{S-17}$$

Now we calculate the ratio η *of* $\mathcal{L}_{\text{SCL}}(\mathbf{z}_s)$ *and* $\mathcal{L'}_{\text{SCL}}(\mathbf{w}_s)$.

$$\eta = \frac{\mathcal{L}_{\text{SCL}}(\mathbf{z}_s)}{\mathcal{L'}_{\text{SCL}}(\mathbf{w}_s)} = \frac{\log[1 + e^{(A-B)/\tau}]}{\log[1 + e^{(A'-B')/\tau}]} \tag{S-18}$$

According to Equation (S-17), *we obtain that*

$$\begin{aligned} &\frac{\log[1 + e^{\frac{(A-B)}{\tau}}]}{\log[1 + e^{\frac{(A'-B')}{\tau}}]} \leq \frac{\log[1 + e^{\frac{(A-B)}{\tau}}]}{\log[1 + e^{\frac{(A-B)}{\tau}} e^{\frac{(-\gamma_1-\gamma_2)}{\tau}}]} \\ &\leq \frac{\log[1 + e^{\frac{(A-B)}{\tau}}]}{\log[e^{\frac{(-\gamma_1-\gamma_2)}{\tau}} + e^{\frac{(A-B)}{\tau}} e^{\frac{(-\gamma_1-\gamma_2)}{\tau}}]} = \frac{\log[1 + e^{\frac{(A-B)}{\tau}}]}{\log[1 + e^{\frac{(A-B)}{\tau}}] + \frac{-\gamma_1-\gamma_2}{\tau}}. \end{aligned} \tag{S-19}$$

Let us denote $C = \log[1 + e^{\frac{(A-B)}{\tau}}]$. *From Equation* (S-19),

$$\frac{\log[1 + e^{\frac{(A-B)}{\tau}}]}{\log[1 + e^{\frac{(A'-B')}{\tau}}]} \leq \frac{C}{C + \frac{-\gamma_1-\gamma_2}{\tau}} = 1 + \frac{\gamma_1 + \gamma_2}{\tau C - \gamma_1 - \gamma_2}. \tag{S-20}$$

Similarly, we can obtain that

$$\frac{\log[1+e^{\frac{(A-B)}{\tau}}]}{\log[1+e^{\frac{(A'-B')}{\tau}}]} \geq 1-\frac{\gamma_1+\gamma_2}{\tau C+\gamma_1+\gamma_2}. \tag{S-21}$$

From Equations (S-20) and (S-21), we obtain that

$$1-\frac{\gamma_1+\gamma_2}{\tau C+\gamma_1+\gamma_2} \leq \eta \leq 1+\frac{\gamma_1+\gamma_2}{\tau C-\gamma_1-\gamma_2} \tag{S-22}$$

When $\gamma_1 \to 0$ and $\gamma_2 \to 0$, we have $1-\frac{\gamma_1+\gamma_2}{\tau C+\gamma_1+\gamma_2} \to 1$ and $1+\frac{\gamma_1+\gamma_2}{\tau C-\gamma_1-\gamma_2} \to 1$. According to the Sandwich Theorem, we have $\eta \to 1$. Therefore, $\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)$ is equivalent to $\mathcal{L}'_{\mathrm{SCL}}(\mathbf{w}_s)$ when sufficiently trained.

ii)* $\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)$ *and* $\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{w}_s)$ *have the same direction.

Since $\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s) \propto (\mathbf{z}_s^{fake}-\mathbf{z}_s^{real})$ and $\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{w}_s) \propto (\mathbf{w}_s^{fake}-\mathbf{w}_s^{real})$, the inner product of $\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)$ and $\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{w}_s)$ is proportional to

$$\begin{aligned}(\mathbf{z}_s^{fake}-\mathbf{z}_s^{real})\cdot(\mathbf{w}_s^{fake}-\mathbf{w}_s^{real}) =& \mathbf{z}_s^{fake}\cdot\mathbf{w}_s^{fake}-\mathbf{z}_s^{fake}\cdot\mathbf{w}_s^{real} \\ &-\mathbf{z}_s^{real}\cdot\mathbf{w}_s^{fake}+\mathbf{z}_s^{real}\cdot\mathbf{w}_s^{real}.\end{aligned} \tag{S-23}$$

According to Equations (S-4a) and (S-4b), we have

$$(\mathbf{z}_s^{fake}-\mathbf{z}_s^{real})\cdot(\mathbf{w}_s^{fake}-\mathbf{w}_s^{real}) \geq -\log(e^{\epsilon_1}-1)-\log(e^{\epsilon_2}-1). \tag{S-24}$$

Since $|\mathbf{z}_s^{fake}-\mathbf{z}_s^{real}| \leq |\mathbf{z}_s^{fake}|+|\mathbf{z}_s^{real}| = 2$ and $|\mathbf{w}_s^{fake}-\mathbf{w}_s^{real}| \leq |\mathbf{w}_s^{fake}|+|\mathbf{w}_s^{real}| = 2$, we have

$$\begin{aligned}\cos\theta = \frac{(\mathbf{z}_s^{fake}-\mathbf{z}_s^{real})\cdot(\mathbf{w}_s^{fake}-\mathbf{w}_s^{real})}{|\mathbf{z}_s^{fake}-\mathbf{z}_s^{real}|\cdot|\mathbf{w}_s^{fake}-\mathbf{w}_s^{real}|} &\geq \frac{-\log(e^{\epsilon_1}-1)-\log(e^{\epsilon_2}-1)}{4} \\ &= 1-\frac{1}{8}(\gamma_1^2+\gamma_2^2).\end{aligned} \tag{S-25}$$

As $\gamma_1 \to 0$ and $\gamma_2 \to 0$, $\cos\theta \to 1$. This result demonstrates the gradients of the two losses are in the same direction.

***iii)* $|\nabla_{\mathbf{z}_t}\mathcal{L}'_{\mathrm{SCL}}(\mathbf{w}_s)|$ *is equivalent to* $|\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)|$.**

The gradients $|\nabla_{\mathbf{z}_t}\mathcal{L}'_{\mathrm{SCL}}(\mathbf{w}_s)|$ and $|\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)|$ can be calculated as

$$\nabla_{\mathbf{z}_t}\mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s) = \frac{e^{\frac{A-B}{\tau}}}{1+e^{\frac{A-B}{\tau}}}\frac{1}{\tau}(\mathbf{z}_s^{fake}-\mathbf{z}_s^{real}), \tag{S-26}$$

and

$$\nabla_{\mathbf{z}_t}\mathcal{L}'_{\mathrm{SCL}}(\mathbf{w}_s) = \frac{e^{\frac{A'-B'}{\tau}}}{1+e^{\frac{A'-B'}{\tau}}}\frac{1}{\tau}(\mathbf{w}_s^{fake}-\mathbf{w}_s^{real}). \tag{S-27}$$

We further compare $|\mathbf{w}_s^{fake}-\mathbf{w}_s^{real}|^2$ and $|\mathbf{z}_s^{fake}-\mathbf{z}_s^{real}|^2$.

$$\begin{aligned}|\mathbf{w}_s^{fake}-\mathbf{w}_s^{real}|^2-|\mathbf{z}_s^{fake}-\mathbf{z}_s^{real}|^2 =& |\mathbf{w}_s^{fake}|^2+|\mathbf{w}_s^{real}|^2-|\mathbf{z}_s^{fake}|^2-|\mathbf{z}_s^{real}|^2 \\ &+2(\mathbf{z}_s^{fake}\cdot\mathbf{z}_s^{real}-\mathbf{w}_s^{fake}\cdot\mathbf{w}_s^{real})\end{aligned} \tag{S-28}$$

Since $|\mathbf{w}_s^{fake}| = |\mathbf{w}_s^{real}| = |\mathbf{z}_s^{fake}| = |\mathbf{z}_s^{real}| = 1$,

$$|\mathbf{w}_s^{fake} - \mathbf{w}_s^{real}|^2 - |\mathbf{z}_s^{fake} - \mathbf{z}_s^{real}|^2 = 2(\mathbf{z}_s^{fake} \cdot \mathbf{z}_s^{real} - \mathbf{w}_s^{fake} \cdot \mathbf{w}_s^{real}). \tag{S-29}$$

Supposing that $\mathbf{z}_s^{fake} = \mathbf{w}_s^{fake} + \zeta_1$ *and* $\mathbf{z}_s^{real} = \mathbf{w}_s^{real} + \zeta_2$, *we have* $|\zeta_1| \leq \gamma_1$ *and* $|\zeta_2| \leq \gamma_2$ *from Equations* (S-12a) *and* (S-12b)*. Then, we obtain the equivalent form as*

$$|\mathbf{w}_s^{fake} - \mathbf{w}_s^{real}|^2 - |\mathbf{z}_s^{fake} - \mathbf{z}_s^{real}|^2 = 2(\zeta_1 \cdot \mathbf{w}_s^{real} + \zeta_2 \cdot \mathbf{w}_s^{fake} + \zeta_1 \cdot \zeta_2). \tag{S-30}$$

In Equation (S-30), $-|\zeta_1||\mathbf{w}_s^{real}| \leq \zeta_1 \cdot \mathbf{w}_s^{real} \leq |\zeta_1||\mathbf{w}_s^{real}|$, $-|\zeta_2||\mathbf{w}_s^{fake}| \leq \zeta_2 \cdot \mathbf{w}_s^{fake} \leq |\zeta_2||\mathbf{w}_s^{fake}|$, *and* $-|\zeta_1||\zeta_2| \leq \zeta_1 \cdot \zeta_2 \leq |\zeta_1||\zeta_2|$. *Thus,*

$$\begin{aligned} -2(|\zeta_1||\mathbf{w}_s^{real}| + |\zeta_2||\mathbf{w}_s^{fake}| + |\zeta_1||\zeta_2|) \leq |\mathbf{w}_s^{fake} - \mathbf{w}_s^{real}|^2 - |\mathbf{z}_s^{fake} - \mathbf{z}_s^{real}|^2 \\ \leq 2(|\zeta_1||\mathbf{w}_s^{real}| + |\zeta_2||\mathbf{w}_s^{fake}| + |\zeta_1||\zeta_2|) \end{aligned} \tag{S-31}$$

According to Equations (S-12a) *and* (S-12b)*, when* $\gamma_1 \to 0$ *and* $\gamma_2 \to 0$, $|\zeta_1| \to 0$ *and* $|\zeta_2| \to 0$. *From Equation* (S-31), $|\mathbf{z}_s^{fake} - \mathbf{z}_s^{real}| \to |\mathbf{w}_s^{fake} - \mathbf{w}_s^{real}|$, *as* $\gamma_1 \to 0$ *and* $\gamma_2 \to 0$. *Therefore,* $|\nabla_{\mathbf{z}_t} \mathcal{L}'_{\mathrm{SCL}}(\mathbf{w}_s)|$ *is equivalent to* $|\nabla_{\mathbf{z}_t} \mathcal{L}_{\mathrm{SCL}}(\mathbf{z}_s)|$.

From i)–iii), we conclude that the weight vector of the pre-trained classifier is the equivalent representation of the feature embeddings for the supervised contrastive loss.

B. Source Model Architecture

Our SDA-FAS leverages the vision transformer based architecture as the pre-trained source model. Specifically, the vision transformer encoder is employed for feature encoding. Based on the output visual tokens, a convolution layer and a linear layer are cascaded for feature embedding, and subsequently, a single-layer linear classifier is used for classification. Vision transformer (ViT) [3] flattens and tokenizes 2D images into a sequence of embeddings. A trainable linear projection $\mathbf{E}$ projects the flattened patches into patch embeddings, which then concatenate with learnable 1-D position embeddings $\mathbf{E}_{pos}$. For an input image $\mathbf{I} \in \mathbb{R}^{H \times W \times C}$, the tokenized sequence of embeddings is

$$\mathbf{z}_0 = \left[\mathbf{I}_{\text{class}}; \mathbf{I}^1\mathbf{E}; \mathbf{I}^2\mathbf{E}; \cdots; \mathbf{I}^N\mathbf{E}\right] + \mathbf{E}_{pos}. \tag{S-32}$$

$\mathbf{I}_{\text{class}}$ is a learnable class embedding and $\mathbf{I}^i$ represents the i-th patch of $\mathbf{I}$. Given the patch size $P \times P$, the number of patches is $N = HW/P^2$. The transformer encoder consists of L blocks of one multi-head self-attention (MSA) and one multi-layer perceptron (MLP) following one layer normalization (LN) respectively. For $\ell = 1 \ldots L$, a Transformer encoder block processes as:

$$\begin{aligned} \mathbf{z}'_\ell &= \mathrm{MSA}\left(\mathrm{LN}\left(\mathbf{z}_{\ell-1}\right)\right) + \mathbf{z}_{\ell-1}, \\ \mathbf{z}_\ell &= \mathrm{MLP}\left(\mathrm{LN}\left(\mathbf{z}'_\ell\right)\right) + \mathbf{z}'_\ell, \end{aligned} \tag{S-33}$$

Table S-1. HTER(%) and AUC(%) for multi-source domains cross-dataset test with four testing scenarios for different pre-trained source models, i.e., our SDA-FAS, vanilla vision transformer architecture and CNNs based ResNet-50 architecture. ↓ indicates the performance gain, i.e., HTER reduction, after adaptation.

Pre-trained Models	Methods	O&C&I→M		O&M&I→C		O&C&M→I		I&C&M→O	
		HTER(%)	AUC(%)	HTER(%)	AUC(%)	HTER(%)	AUC(%)	HTER(%)	AUC(%)
Lv et al. [7]	SourceOnly	19.28	-	27.77	-	23.58	-	18.22	-
	After adaptation	18.17↓1.11	-	25.51↓2.26	-	20.04↓3.54	-	17.50↓0.72	-
Vanilla ViT [3]	SourceOnly	14.58	94.44	21.11	87.88	21.25	81.81	21.67	86.59
	After adaptation	7.92↓6.66	96.28	7.22↓13.89	96.54	7.75↓13.50	95.48	9.17↓12.50	97.12
ResNet-50 [4]	SourceOnly	20.42	86.13	16.67	92.05	23.75	84.39	21.60	86.55
	After adaptation	16.25↓4.17	88.77	3.52↓13.15	99.08	10.12↓13.63	96.24	12.74↓8.86	93.81
SDA-FAS	SourceOnly	12.50	93.71	20.00	90.53	16.25	90.99	17.26	91.80
	After adaptation	**5.00**↓7.50	**97.96**	**2.40**↓17.60	**99.72**	**2.62**↓13.63	**99.48**	**5.07**↓12.19	**99.01**

Table S-2. A summary of the FAS datasets used in our experiments.

Datasets	Subjects	Data	Sensors	Spoof Types
Idiap Replay-Attack (I) [2]	50	1,200 videos	2	1 Print, 2 Video-replay
OULU-NPU (O) [1]	55	4,950 videos	6	2 Print, 2 Video-replay
CASIA-MFSD (C) [12]	50	600 videos	3	2 Print, 1 Video-replay
MSU-MFSD (M) [10]	35	280 videos	2	1 Print, 2 Video-replay
CelebA-Spoof (CA) [11]	10177	625,537 images	>10	3 Print, 3 Replay, 3 Paper Cut, 1 3D Mask

The output tokens of transformer encoder are $\mathbf{z}_L = [\mathbf{z}_L^c; \mathbf{z}_L^1; \mathbf{z}_L^2; \cdots; \mathbf{z}_L^N]$, where $\mathbf{z}_L^c$ is the class token and $\mathbf{z}_L^1, \mathbf{z}_L^2, \cdots, \mathbf{z}_L^N$ represent the visual tokens. Here, we leverage visual tokens that contain meaningful representations of live/spoof features extracted from local image patches, rather than the class token. We reshape the matrix of visual tokens into a spatial feature map $\mathbf{Z} \in \mathbb{R}^{H' \times W' \times D}$ with $H'W' = N$. $\mathbf{Z}$ is then fed into a feature embedding head consisting of one convolution layer with batch normalization and ReLU activation (denoted by $\texttt{Conv-BN-ReLU}(\cdot)$) and one linear layer (denoted by $\texttt{Linear}(\cdot)$) to obtain the feature embeddings as

$$\mathbf{z} = \ell_2\texttt{-norm}(\texttt{Linear}(\texttt{Conv-BN-ReLU}(\mathbf{Z}))). \tag{S-34}$$

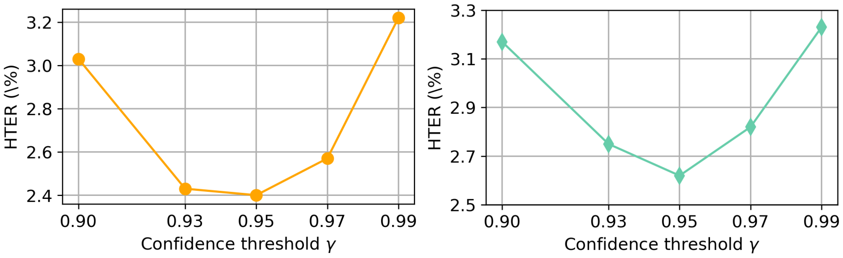
Then a one-layer linear classifier with weight normalization is leveraged for predicting real/fake as $\tilde{\mathbf{y}} = \texttt{Linear}(\mathbf{z})$. By contrast, the vanilla ViT only employs a linear classifier based on the class token as $\tilde{\mathbf{y}} = \texttt{Linear}(\mathbf{z}_L^c)$.

C. Extensive Experiments

Different Pre-trained Source Models. To further demonstrate the proposed adaptation framework is effective for various source models, we conduct experiments with two additional pre-trained source models, i.e., vanilla vision transformer architecture [3] (abbreviated as vanilla ViT) and CNN architecture of ResNet-50 [4] (abbreviated as ResNet-50). As shown in Table S-1, the proposed framework can significantly improve the performance of pre-trained models after

Table S-3. Ablation study for patch shuffle (PS) data augmentation.

Protocols	O&C&I→M		O&M&I→C		O&C&M→I		I&C&M→O	
	HTER(%)	AUC(%)	HTER(%)	AUC(%)	HTER(%)	AUC(%)	HTER(%)	AUC(%)
Ours SDA-FAS	**5.00**	**97.96**	**2.40**	**99.72**	**2.62**	**99.48**	**5.07**	**99.01**
Ours SDA-FAS w/o PS	5.42	97.96	3.33	99.26	3.75	99.36	5.56	98.40



(a) O&M&I→C (b) O&C&M→I

Fig. S-1. HTER (%) with different confidence threshold γ.

adaptation, including the vision transformer and CNN based architectures. This result demonstrates that our SDA-FAS is universally effective for different pre-trained models. Remarkably, adaptation with SDA-FAS achieves a significant HTER reduction of 12.72% on average. Implemented with the proposed framework, vanilla ViT and ResNet-50 also achieve considerable performance gains, i.e., 11.64% and 9.95% HTER reduction on average, respectively. Considering the benefits on both vanilla ViT and ResNet-50, we believe that SDA-FAS can be further boosted with a more robust source model and well-designed pre-training strategy. By contrast, Lv et.al [7] suffer from poor adaptation performance, i.e., a trivial 1.90% HTER reduction, as the adaptation is achieved with direct self-training on noisy pseudo labels without a specific design for sufficiently exploring FAS tasks.

Patch-shuffle data augmentation. We further conduct experiments on our SDA-FAS without patch shuffle data augmentation. Without the FAS specific patch shuffle data augmentation, our method suffers from the performance degradation.

The confidence threshold. We evaluate different values of γ from 0.90 to 0.99. Fig. S-1 shows the value of 0.95 achieves the lowest HTER. γ controls the trade-off between the quality and quantity of pseudo labels. A larger γ leads to higher accuracy of pseudo labels for unlabeled target data but a smaller amount of unlabeled data contributing to training. We find the best trade-off is achieved when γ is 0.95. The quality of pseudo labels is low when γ is 0.90 and the quantity is small for γ=0.99.

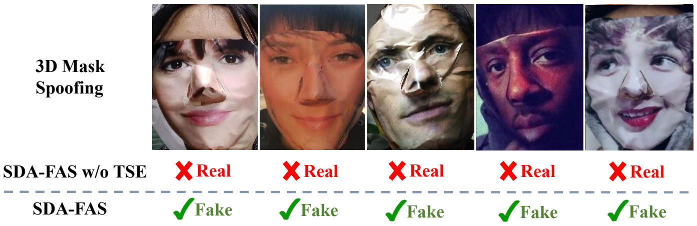


Fig. S-2. Qualitative analysis of 3D mask attack types on CelebA-Spoof [11]. 'Real' and 'Fake' denote the predicted results from the models. ✗ indicates a wrong prediction and ✓ indicates a correct prediction.

D. Visualization

Qualitative analysis for validating the effectiveness of the target self-supervised exploration (TSE) module is conducted by visualizing examples of hard 3D mask spoofing faces, which are misclassified by SDA-FAS w/o TSE but correctly classified by SDA-FAS. As shown in Table S-2, SDA-FAS can correctly identify the fake 3D mask faces, while SDA-FAS w/o TSE fails. This fact demonstrates the effectiveness of TSE for exploring novel attack types in the target data by itself.

E. Datasets

Experiments are conducted on five publicly available datasets: Idiap Replay-Attack [2] (denoted as I), OULU-NPU [1] (denoted as O), CASIA-MFSD [12] (denoted as C), MSU-MFSD [10] (denoted as M) and CelebA-Spoof [11] (denoted as CA). Basic information of these datasets is summarized in Table S-2.

- **Idiap Replay-Attack** (*abbr.* I) captures all live and spoof faces from 50 clients under two different lighting conditions in 1,200 videos. Five attack types consist of four kinds of replayed faces and one kind of printed face.
- **OULU-NPU** (*abbr.* O) is a high-resolution dataset with 3,960 spoof face videos and 990 live face videos, containing two kinds of printed spoof faces and two kinds of replayed spoof faces captured under six cameras and three sessions.
- **CASIA-MFSD** (*abbr.* C) consists of 50 subjects and each subject has 12 videos. Three attack types (printed photo attack, cut photo attack, and video attack) are used to create spoof faces, and each face image is recorded with three kinds of imaging qualities.
- **MSU-MFSD** (*abbr.* M) consists of totally 280 videos for 35 subjects under two different cameras. Three spoof types include two kinds of replayed faces and one kind of printed face.
- **CelebA-Spoof** (*abbr.* CA) is the current largest scale FAS dataset with rich and diverse annotations, which comprises 625,537 pictures of 10,177 subjects

Table S-4. Configuration of training hyper-parameters.

Phases	Source Pretraining	Target Adaptation
Epochs	30	60
Feature Embedding Head & Classifier		
Optimizer	SGD	SGD
Learning rate	1e-3	1e-4
Weight decay	5e-4	5e-4
Momentum	0.9	0.9
Vision Transformer Encoder		
Optimizer	AdamW	AdamW
Learning rate	1e-4	5e-5

covering four spoofing types (i.e., print, paper-cut, replay, and 3D mask) captured under eight scenes.

F. Implementation Details

For pre-training on the source domain, we employ DeiT-S [9] pre-trained on ImageNet as the transformer encoder. Since the original input size for the transformer encoder is 224×224, we conduct the bilinear interpolation of the positional embedding. We randomly specify a 0.9/0.1 train-validation split in the source dataset and generate the optimal pre-trained model based on the HTER of the validation set after 30 epochs of training.

For adapting on the target domain, the trainable vision transformer encoder is fine-tuned by the AdamW [6] optimizer. The feature embedding head and the classifier are fine-tuned by the SGD optimizer. The temperature τ for supervised contrastive loss is set to 0.1, and the temperature η for self-supervised learning is set to 0.1. The confidence threshold γ is 0.95. Network and center momentum rates are set to $l = 0.999$ and $m = 0.9$, respectively. The maximum number of the training epoch is 60. Other training hyper-parameters of the pre-training phase and target adaptation phase are listed in Table S-4.

Following [5,8], we use the training/test split provided by the original dataset, i.e., 120/160 videos for M, 360/480 videos for I, 240/360 videos for C, 1800/1800 videos for O and 500,429/62,553 images for CA. For M, I, C and O, We perform adaptation using unlabeled target training set and evaluate on the test set.

G. Main Algorithms

Our SDA-FAS contains two phases: source pre-training phase on the company side and target adaptation phase on the deployment side, as elaborated in Algorithms 1 and 2 , respectively. After source pre-training, the pre-trained source model is provided for deployment and adapted with few unlabeled target data.

Algorithm 1 Source Model Pre-Training

```
Require: Source domain dataset $\mathcal{D}_S = \{\mathbf{x}_S, \mathbf{y}_S\}$, maximum number of training epochs $N_S$, feature extractor $g_s$ composed of a pre-trained transformer encoder and a randomly initialized feature embedding head with a convolutional layer and a linear layer, and a randomly initialized one-layer linear classifier $h_s$ with $\ell_2$-normalized weights.
Ensure: Pre-trained source model $f_s = h_s \circ g_s$.
Randomly split the dataset $\mathcal{D}_S$ into training set $\mathcal{D}_S^{train}$ and validation set $\mathcal{D}_S^{val}$ by the ratio of 0.9 to 0.1.
Initialize the best HTER with $\text{HTER}_{best} = 1$.
for $epoch = 1$ to $N_S$ do
    Obtain the model output $\tilde{\mathbf{y}}_S^{train} = f_s(\mathbf{x}_S^{train})$.
    Calculate the loss $\mathcal{L}_{ce} = \text{BCE}(\tilde{\mathbf{y}}_S^{train}, \mathbf{y}_S^{train})$.
    Update the parameters of $f_s(\cdot)$ via $\mathcal{L}_{ce}$.
    Evaluate $f_s(\cdot)$ on $\mathcal{D}_S^{val}$ to obtain $\text{HTER}_{current}$
    if $\text{HTER}_{current} \leq \text{HTER}_{best}$ then
        Save $f_s(\cdot)$ as the best model.
        Update $\text{HTER}_{best} = \text{HTER}_{current}$.
    end if
end for
return Pre-trained source model $f_s = h_s \circ g_s$.
```

Algorithm 2 Target Adaptation for Face Anti-Spoofing

```
Require: Fixed pre-trained source model $f_s = h_s \circ g_s$, unlabeled target domain training dataset $\mathcal{D}_T = \{\mathbf{x}_T\}$, maximum number of training epochs $N_T$, update period of pseudo labels $n_T$, trainable target networks $\{g_t,\ g_t^{tea},\ h_{t2s}\}$, temperature $\tau$ and $\gamma$, confidence threshold $\gamma$, center $\mathbf{C}$, network momentum rate $l$, center momentum rate $m$, loss balanced parameters $\alpha$, $\lambda_1$ and $\lambda_2$.
Ensure: Target model $f_t = h_t \circ g_t$.
Initialization: Freeze the classifier $h_t = h_s$ and $h_t^{tea} = h_s$, and copy the parameters from $h_s$ to $h_{t2s}$ as initialization. Initialize the feature extractor $g_t$ and $g_t^{tea}$ with the parameters of $g_s$.
for $epoch = 1$ to $N_T$ do
    Given an original image $\mathbf{x}_T$, a patch-permuted image $\mathbf{x}'_T = \text{augment}(\mathbf{x}_T)$ is generated by randomly patch shuffle.
    Calculate the source-oriented pseudo labels $\bar{\mathbf{y}}_T^s = \text{argmax}(h_s(g_s(\mathbf{x}_T)))$ and the prediction confidence $\mathbf{c}_T^s = \max(h_s(g_s(\mathbf{x}_T)))$.
    if epoch % $n_T$ == 0 then
        Calculate the target-oriented pseudo labels $\bar{\mathbf{y}}_T^t = \text{argmax}(h_t(g_t(\mathbf{x}_T)))$ and the prediction confidence $\mathbf{c}_T^t = \max(h_t(g_t(\mathbf{x}_T)))$.
    end if
    Calculate the feature embeddings $\mathbf{z}_t = g_t(\mathbf{x}_T)$.
    Calculate the contrastive domain alignment loss $\mathcal{L}_{\text{CDA}}$ via Equation (4).
    Calculate the model output $\tilde{\mathbf{y}}_{t2s} = h_{t2s}(g_t(\mathbf{x}_T))$ and $\tilde{\mathbf{y}}_t = h_t(g_t(\mathbf{x}_T))$.
    Calculate the loss of self-training with source regularization $\mathcal{L}_{\text{SSR}}$ via Equation (3).
    Calculate the output probability distributions of the student and teacher network $P_{stu}, P'_{tea}, P'_{stu}, P_{tea}$ via Equation (5).
    Calculate the output of the teacher network as $\tilde{\mathbf{y}}^{tea} = f_t^{tea}(\mathbf{x}_T)$ and $\tilde{\mathbf{y}}'^{tea} = f_t^{tea}(\mathbf{x}'_T)$
    Calculate the target self-supervised exploration loss $\mathcal{L}_{\text{TSE}}$ via Equation (6).
    Obtain the overall loss $\mathcal{L}$ via Equation (7).
    Update the parameters of the module $g_t$, $h_{t2s}$ via $\mathcal{L}$ by gradient descent.
    Update the parameters of the teacher network via EMA: $g_t^{tea}.\text{params} = l * g_t^{tea}.\text{params} + (1 - l) * g_t.\text{params}$
    Update $\mathbf{C} = m\mathbf{C} + (1 - m)(\tilde{\mathbf{y}}^{tea} + \tilde{\mathbf{y}}'^{tea})/2$.
end for
return Target model $f_t = g_t \circ h_t$.
```

References

1. Boulkenafet, Z., Komulainen, J., Li, L., Feng, X., Hadid, A.: OULU-NPU: A mobile face presentation attack database with real-world variations. In: FG. pp. 612–618 (2017) 6, 8
2. Chingovska, I., Anjos, A., Marcel, S.: On the effectiveness of local binary patterns in face anti-spoofing. In: BIOSIG. pp. 1–7 (2012) 6, 8
3. Dosovitskiy, A., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021) 5, 6
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016) 6
5. Jia, Y., Zhang, J., Shan, S., Chen, X.: Unified unsupervised and semi-supervised domain adaptation network for cross-scenario face anti-spoofing. Pattern Recognition **115**, 107888 (2021) 9
6. Loshchilov, I., Hutter, F.: Fixing weight decay regularization in adam. arXiv preprint arXiv:1711.05101 (2018) 9
7. Lv, L., Xiang, Y., Li, X., Huang, H., Ruan, R., Xu, X., Fu, Y.: Combining dynamic image and prediction ensemble for cross-domain face anti-spoofing. In: ICASSP. pp. 2550–2554 (2021) 6, 7
8. Quan, R., Wu, Y., Yu, X., Yang, Y.: Progressive transfer learning for face anti-spoofing. IEEE Transactions on Image Processing **30**(3), 3946–3955 (2021) 9
9. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers and distillation through attention. arXiv preprint arXiv:2012.12877 (2020) 9
10. Wen, D., Han, H., Jain, A.K.: Face spoof detection with image distortion analysis. IEEE Transactions on Information Forensics and Security **10**(4), 746–761 (2015) 6, 8
11. Zhang, Y., et al.: CelebA-Spoof: Large-scale face anti-spoofing dataset with rich annotations. In: ECCV. pp. 70–85 (2020) 6, 8
12. Zhang, Z., Yan, J., Liu, S., Lei, Z., Yi, D., Li, S.Z.: A face antispoofing database with diverse attacks. In: ICB. pp. 26–31. New Delhi, India (2012) 6, 8