BUT SYSTEMS FOR WILDSPOOF CHALLENGE: SASV IN THE WILD

Junyi Peng[1], *Jin Li*[1,3], *Johan Rohdin*[1], *Lin Zhang*[2], *Miroslav Hlaváček*[1], *Oldřich Plchot*[1]

[1]Speech@FIT, Brno University of Technology, Czechia [2]Johns Hopkins University, USA
[3]Department of EEE, The Hong Kong Polytechnic University, Hong Kong SRA

ABSTRACT

This paper presents the BUT submission to the WildSpoof Challenge, focusing on the Spoofing-robust Automatic Speaker Verification (SASV) track. We propose a SASV framework designed to bridge the gap between general audio understanding and specialized speech analysis. Our subsystem integrates diverse Self-Supervised Learning front-ends ranging from general audio models (e.g., Dasheng) to speech-specific encoders (e.g., WavLM). These representations are aggregated via a lightweight Multi-Head Factorized Attention back-end for corresponding subtasks. Furthermore, we introduce a feature domain augmentation strategy based on Distribution Uncertainty to explicitly model and mitigate the domain shift caused by unseen neural vocoders and recording environments. By fusing these robust CM scores with state-of-the-art ASV systems, our approach achieves superior minimization of the a-DCFs and EERs.

***Index Terms*—** Self-supervised learning, speaker verification, anti-spoofing, fine-tuning

1. INTRODUCTION

Although previous challenges like ASVspoof [1, 2] have advanced the field of Spoofing-robust ASV (SASV), they often rely on clean studio-recorded bona fide speech, creating a mismatch with real-world deployment scenarios where noise and reverberation are ubiquitous. The newly introduced **WildSpoof Challenge** [3] and the **SpoofCeleb** dataset [4] address this gap by deriving bona fide speech from VoxCeleb1 [5] and generating spoofing attacks using TTS systems trained on the same noisy data.

The core task of WildSpoof is SASV, which requires a system to accept only bona fide target trials while rejecting both zero-effort impostors (non-target) and spoofing attacks. This presents a core challenge: the system must be robust to the "generation-recognition trade-off" [4], effectively distinguishing forensic artifacts in noisy conditions where traditional detection cues might be masked.

In this paper, we present the **BUT** submission to the WildSpoof SASV track. Our approach focuses on maximizing the representation power of diverse Self-Supervised Learning (SSL) models for both the anti-spoofing countermeasure (CM) and automatic speaker verification (ASV) systems. We hypothesize that large-scale pre-trained models encode rich acoustic information that, when properly aggregated, is resilient to environmental noise [6]. Crucially, we propose a flexible framework designed to be compatible with both general audio SSLs (e.g., Dasheng [7]) and speech-specific SSLs (e.g., WavLM [8], W2V2-BERT [9]). This allows us to leverage the broad acoustic understanding of audio models alongside the specialized phonetic features of speech models. Moreover, we utilize a **Multi-Head Factorized Attention (MHFA)** backend [10] augmented with a **Distribution Uncertainty (DSU**) module [11] to simulate unseen domain shifts.

Our main contributions are summarized as follows:

- We demonstrate that general audio SSL models (e.g., Dasheng) provide complementary robustness to speech-specific models when applied to noisy, in-the-wild deepfake detection within our proposed framework.
- We integrate DSU-based feature augmentation into the MHFA backend, significantly improving performance on unseen attacks in the SpoofCeleb evaluation set.

2. PROPOSED METHOD

2.1. Data and Framework

The proposed system is developed within the **WeDefense** framework[1], an open-source toolkit specifically designed for defending against fake audio. We utilize the official **SpoofCeleb** dataset as the primary source for both training and developing our CM models. Furthermore, to enhance the speaker verification component, we incorporate the **VoxCeleb2-dev** dataset for training our SV systems.

2.2. Hierarchical SSL Feature Extraction

SSL models, such as WavLM [8], HuBERT [12], and Dasheng [7], encode rich acoustic information across their layers. Lower layers typically capture raw spectral details, while upper layers encode more semantic or structural information. For the task of deepfake detection, relying on the last layer is often sub-optimal, as forensic artifacts introduced by neural

[1]https://github.com/zlin0/wedefense

Table 1. ASV Performance Comparison (EER %) (spoofed trials are excluded.).

System	Vox1-O	Vox1-E	Vox1-H	SpoofCeleb-Dev (SV)	SpoofCeleb-Eval (SV)
ResNet293	0.447	0.657	1.183	3.209	3.053
WavLM Large + MHFA	0.516	0.583	1.179	3.273	3.510
W2V2+BERT + MHFA	**0.229**	**0.354**	**0.714**	**2.441**	**2.528**

vocoders (e.g., phase discontinuities or metallic buzzing) are often best preserved in intermediate representations.

Therefore, instead of using only the final-layer hidden state, we employ MHFA, a lightweight backend that learns layer-wise attention weights and computes a weighted sum of all L transformer layers. Let $X \in \mathbb{R}^{L\times T\times D}$ be the output features from all layers of the SSL encoder, where T is the number of frames and D is the feature dimension. We learn layer-specific weights to aggregate these features dynamically, allowing the backend to focus on the most discriminative level of abstraction.

2.3. Multi-Head Factorized Attention (MHFA)

To effectively aggregate the temporal frame-level features into a global utterance-level embedding, we employ the **MHFA** mechanism [10]. Unlike standard attention, which uses a single linear projection, MHFA factorizes the aggregation process into two independent streams: a *Key* stream (K) and a *Value* stream (V).

Specifically, we define two separate sets of learnable layer weights, $w^k \in \mathbb{R}^L$ and $w^v \in \mathbb{R}^L$. These weights are normalized via softmax to compute the weighted sum of the SSL layer outputs Z_l:

$$K_{feat} = \sum_{l=1}^{L} \text{softmax}(w_l^k) Z_l, \quad V_{feat} = \sum_{l=1}^{L} \text{softmax}(w_l^v) Z_l \tag{1}$$

These aggregated features are then projected into a lower dimension D_{cmp} using linear layers W_k and W_v:

$$K = K_{feat} W_k, \quad V = V_{feat} W_v \tag{2}$$

The attention weights A are computed from the Query stream, while the content to be aggregated comes from the Value stream. This factorization allows the model to learn *where* to look using K independently of *what* to extract using V. For H heads, the output is pooled as:

$$A = \text{softmax}(KW_{att}, \text{dim} = 1) \tag{3}$$

$$Embedding = \text{Pooling}(V \odot A) \tag{4}$$

Finally, a fully connected layer maps the concatenated head outputs to the final embedding e. This embedding is then processed by the corresponding classification head based on the specific task.

2.4. MHFA with DSU (Feature Domain Augmentation)

To tackle the challenge of unseen generators, we integrate a Feature Domain Augmentation strategy directly into the MHFA backend [11], termed **MHFA-DSU**. This method is based on the concept of Distribution Uncertainty (DSU), which hypothesizes that domain shifts can be simulated by perturbing the feature statistics (mean and variance) of the training data.

We apply DSU specifically to the Value stream (V_{feat}) before the linear projection. During training, with a probability p, we model the feature statistics as distributions rather than deterministic values. For an input feature map x (corresponding to V_{feat}), we compute the instance-level mean μ and standard deviation σ across the temporal dimension.

To simulate unseen domains, we assume the feature statistics follow a Gaussian distribution. We sample uncertainty perturbations ϵ_μ and ϵ_σ from a standard normal distribution $\mathcal{N}(0, 1)$:

$$\tilde{\mu} = \mu + \epsilon_\mu \cdot \Sigma_\mu, \quad \tilde{\sigma} = \sigma + \epsilon_\sigma \cdot \Sigma_\sigma \tag{5}$$

where Σ_μ and Σ_σ are variances that represent the uncertainty of the mean and variance estimates, respectively. The augmented feature $\tilde{x}$ is obtained by re-parameterization:

$$\tilde{x} = \frac{x - \mu}{\sigma} \cdot \tilde{\sigma} + \tilde{\mu} \tag{6}$$

This operation essentially "jitters" the global style and channel characteristics of the audio representation while preserving the local content, forcing the network to learn features that are invariant to global statistical shifts caused by different vocoders.

2.5. Calibration and Fusion

To fuse and calibate decisions from ASV and CM, our report discussed two methods:

- Pre-fusion individual calibration: calibrate ASV and CM scores separately with logistic regression, then fuse (optionally followed by a final calibration).
- Joint calibration/fusion: jointly learn scale and bias for ASV and CM within a single logistic fusion model (as in Eq. 9 of our previous work [2]).

Table 2. Performance of different CM systems trained on SpoofCeleb. *DSU indicates Distribution Uncertainty augmentation.

System	SpoofCeleb Dev		SpoofCeleb Eval		OOD (ASV5 Dev)	
	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF
Speech-Specific SSL						
WavLM Base+	0.402	0.011	0.055	0.001	11.885	0.193
+ MUSAN/RIR	0.805	0.022	0.153	0.003	7.976	0.146
+ RawBoost	**0.103**	**0.003**	**0.041**	**0.001**	6.970	0.158
General Audio SSL						
Mi-Dasheng-base (86M)	0.239	0.006	0.123	0.003	5.164	0.137
Mi-Dasheng-0.6B (600M)	0.176	0.004	0.050	0.001	3.122	0.089
+ DSU	0.213	0.005	0.078	0.002	**1.777**	**0.051**
Mi-Dasheng-1.2B (1200M)	0.265	0.006	0.090	0.002	1.625	0.047
+ DSU	0.301	0.007	0.154	0.003	**1.193**	**0.034**
Baseline						
ResNet18	0.204	0.005	0.185	0.005	12.090	0.177

3. EXPERIMENTS

3.1. Experimental Setup

Our models were implemented using PyTorch and trained on AMD Instinct MI200 GPUs. The training process was configured with a maximum of 8 epochs. We utilized the AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 1.0×10^{-4}. The batch size was set to 128.

We employed a differential learning rate strategy to fine-tune the SSL front-end and the MHFA back-end effectively. The base learning rate was initialized at 5.0×10^{-4} and decayed to a final learning rate of 1.0×10^{-5} using a Cosine Annealing scheduler. To prevent catastrophic forgetting of the pre-trained representations, the learning rate for the SSL front-end was scaled by a factor of 0.05 relative to the base learning rate. A warmup period of 2 epochs was applied at the beginning of training.

For the MHFA backend configuration, we set the number of attention heads (`head_nb`) to 32, the embedding dimension (`embed_dim`) to 256, and the compression dimension (`compression_dim`) to 128.

3.2. Results and Analysis

3.2.1. ASV

We first evaluate the performance of our ASV subsystems. Table 1 summarizes the results of three different ASV models on the standard VoxCeleb test sets (Vox1-O, Vox1-E, Vox1-H) and the SpoofCeleb Development and Evaluation sets. All models were trained on the VoxCeleb2-dev dataset with Additive Angular Margin Softmax (AAM-Softmax) as loss function. And VoxCeleb2-dev is used for cosine score normalization.

The W2V2+BERT + MHFA model demonstrates superior performance across all evaluation sets, significantly outperforming the ResNet293 baseline and the WavLM Large model. Specifically, on the challenging SpoofCeleb-Dev and SpoofCeleb-Eval sets, which contain in-the-wild noisy speech, W2V2+BERT + MHFA achieves the lowest EERs of 2.440% and 2.250%, respectively. This highlights the robustness of the MHFA backend combined with rich self-supervised representations in handling diverse acoustic conditions.

3.2.2. Anti-Spoofing

We evaluate various CM systems utilizing different SSL backbones and augmentation strategies, including WavLM Base+ and Mi-Dasheng models. For all of them, we utilize a standard Binary Cross-Entropy (BCE) loss to distinguish between *bonafide* and *spoof* classes. Table 2 presents the results on SpoofCeleb Dev/Eval sets and an out-of-domain (OOD) set (ASVSpoof5 Dev).

General Audio vs. Speech SSLs: General audio models (Dasheng) consistently outperform speech-specific models (WavLM) and the ResNet18 baseline on the OOD dataset. For instance, Mi-Dasheng-0.6B achieves 3.122% EER on ASV5 Dev compared to 11.885% for WavLM, demonstrating superior generalization. We further observe that Mi-Dasheng-0.6B slightly outperforms Mi-Dasheng-1.2B, suggesting that larger model capacity does not necessarily yield better performance, and selecting an appropriately sized backbone for the data and task can be more effective.

Effectiveness of DSU: Although applying DSU augmentation doesn't show improvement on the in-domain data, it shows significantly improved robustness on OOD data. For Mi-Dasheng-0.6B, DSU reduces the OOD EER from 3.122% to 1.777%. Similarly, for the 1.2B model, DSU improves

Table 3. Results of fusion and calibration on ASV and CM systems. (Only spoofceleb data are used for calibration)

CM		ASV		SASV	Spoofceleb dev.
CM model	**calibrate on**	**ASV model**	**calibrate on**	**fusion/cali. on**	**a-DCF**
Dasheng 0.6B + MHFA + DSU	dev + eval	W2V2+BERT + MHFA	dev + eval	dev+eval	0.02747
Dasheng 0.6B + MHFA + DSU	no	W2V2+BERT + MHFA	dev + eval	dev+eval	0.02747
Dasheng 0.6B + MHFA	dev + eval	W2V2+BERT + MHFA	dev + eval	dev+eval	0.02695

OOD EER from 1.625% to 1.193%, validating its efficacy in handling unseen domains.

3.3. Calibration and Fusion

Table 3 reports results for the two calibration strategies described in Section2.5. We observe that calibrating component scores before fusion versus calibrating the score joinlty during fusion yields similar performance. This suggests that separately calibrating the CM system is unnecessary if a joint calibration is applied. However, it remains unexplored whether joint optimization can be numerically challenging without pre individual calibration before fusion for more difficult sets or for the ASV system which, due to the properties of cosine scoring, has more constrained raw scores.

4. CONCLUSION AND DISCUSSION

This report describes the BUT systems for the WildSpoof Challenge 2025 SASV track. For ASV, we compare ResNet with popular SSL backbones, and for CM, we use general-audio SSL models (Dasheng) with DSU-based feature augmentation and a lightweight MHFA backend. For fusion/calibration, we compare pre-fusion individual calibration and joint calibration/fusion. Results show that general-audio models outperform speech-specific models and the ResNet18 baseline on OOD data, DSU improves generalization, and pre- vs post-fusion calibration performs similarly.

5. REFERENCES

[1] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee *et al.*, "Asvspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, p. 101114, 2020.

[2] J. Rohdin, L. Zhang, O. Plchot, V. Staněk, D. Mihola, J. Peng, T. Stafylakis, D. Beveraki, A. Silnova, J. Brukner *et al.*, "But systems and analyses for the asvspoof 5 challenge," *arXiv preprint arXiv:2408.11152*, 2024.

[3] Y. Wu, J.-w. Jung, H.-j. Shim, X. Cheng, and X. Wang, "Wildspoof challenge evaluation plan," *arXiv preprint arXiv:2508.16858*, 2025.

[4] J.-w. Jung, Y. Wu, X. Wang, J.-H. Kim, S. Maiti, Y. Matsunaga, H.-j. Shim, J. Tian, N. Evans, J. S. Chung *et al.*, "Spoofceleb: Speech deepfake detection and sasv in the wild," *IEEE Open Journal of Signal Processing*, 2025.

[5] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," *Proc. Interspeech 2017*, pp. 2616–2620, 2017.

[6] J. Peng, L. Mošner, L. Zhang, O. Plchot, T. Stafylakis, L. Burget, and J. Černockỳ, "Ca-mhfa: A context-aware multi-head factorized attentive pooling for ssl-based speaker verification," in *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2025, pp. 1–5.

[7] H. Dinkel, Z. Yan, Y. Wang, J. Zhang, Y. Wang, and B. Wang, "Scaling up masked audio encoder learning for general audio classification," *arXiv preprint arXiv:2406.06992*, 2024.

[8] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[9] L. Barrault, Y.-A. Chung, M. C. Meglioli, D. Dale, N. Dong, M. Duppenthaler, P.-A. Duquenne, B. Ellis, H. Elsahar, J. Haaheim *et al.*, "Seamless: Multilingual expressive and streaming speech translation," *arXiv preprint arXiv:2312.05187*, 2023.

[10] J. Peng, O. Plchot, T. Stafylakis, L. Mošner, L. Burget, and J. Černockỳ, "An attention-based backend allowing efficient fine-tuning of transformer models for speaker verification," in *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2023, pp. 555–562.

[11] X. Li, Y. Dai, Y. Ge, J. Liu, Y. Shan, and L.-Y. Duan, "Uncertainty modeling for out-of-distribution generalization," *arXiv preprint arXiv:2202.03958*, 2022.

[12] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.