SHORT-SEGMENT SPEAKER VERIFICATION WITH PRE-TRAINED MODELS AND MULTI-RESOLUTION ENCODER

Jisoo Myoung, Sangwook Han, Kihyuk Kim, Jong Won Shin

Gwangju Institute of Science and Technology, Gwangju, Korea

ABSTRACT

Speaker verification (SV) utilizing features obtained from models pre-trained via self-supervised learning has recently demonstrated impressive performances. However, these pre-trained models (PTMs) usually have a temporal resolution of 20 ms, which is lower than typical filterbank features. It may be problematic especially for short-segment SV with an input segment shorter than 2 s, in which we need to extract as much information as possible from the input with a limited length. Although there have been approaches to utilize multi-resolution features from the HuBERT models, the window shifts were 320, 640, and 1600 samples when the sampling rate was 16 kHz and thus only lower resolution features were considered. In this study, we propose an SV system which utilizes PTM features along with filterbank features and those from the multi-resolution time domain encoder with window shifts of 25, 50, 100, and 200 samples. Experimental results on the VoxCeleb dataset with various input lengths showed consistent improvements over systems with various combinations of input features.

Index Terms— self-supervised learning features, multiple temporal resolutions, speaker verification, short segment

1. INTRODUCTION

Speaker verification (SV) is the process of verifying if the speaker for a given input utterance matches the claimed identity [1]–[15]. With the advancement of deep learning, speaker representations can be utilized in terms of auxiliary information for various speech processing tasks such as target speaker extraction [16], and speech separation [17]. The performance of the SV heavily depends on the length of the input and enrollment utterances, which makes short-segment SV quite challenging. Various approaches have been proposed to extract as much speaker-related information as possible from limited speech signal by handling multi-scale features in temporal or channel directions [3]–[7]. In [3], multi-scale feature-channel attention (MFA) module is proposed as a front-end module for time delay neural networks (TDNNs) to extract multi-scale information effectively by feeding high-resolution information to lower resolution and emphasizing relevant frequency bins and channels. RawNeXt [4] produces speaker embeddings enriched with time-spectral information by iteratively and hierarchically aggregating multi-resolution features. In [5], the ECAPA-TDNN [2] taking log mel-filterbank (FBank) features is modified to incorporate the features extracted by a multi-resolution encoder (MRE) through

adapter modules to utilize information obtained with various sizes of windows. MR-RawNet [6] improves RawNet3 by replacing its first layer with multi-resolution feature extractor (MRFE), enabling simultaneous consideration of temporal and spectral resolutions. ERes2NetV2 [7] expands and compresses channels to strengthen short-duration feature extraction through a bottleneck-like local feature fusion (BLFF), and then a bottom-up dual-stage feature fusion (BDFF) integrates multiscale feature maps with different receptive fields to capture global information.

Recently, there has been growing interest in SV systems that utilize speech representations extracted from models pre-trained via self-supervised learning (SSL) on large-scale speech data [8]–[15]. In particular, various fusion methods to combine layer-wise outputs from pre-trained models (PTMs) to produce the input for the SV models have been proposed [9]–[13]. While they exhibited potential of the PTM-based features, the temporal resolutions for these features were fixed to be the same as those for the corresponding PTMs, which is typically 20 ms. In [18], multiple temporal resolutions in HuBERT [19] have been proposed by pre-training three HuBERT models at different temporal resolutions and fusing their outputs using hierarchical or parallel approaches. MR-HuBERT [20] further extended the original HuBERT with two resolutions through a hierarchical Transformer, where predefined hidden representations were downsampled to incorporate both high- and low-resolution masked unit predictions, respectively. Although exploiting multiple temporal resolutions for HuBERT was shown to be beneficial for a broad range of speech processing tasks in these papers, the temporal resolutions they utilized corresponded to 20, 40, and 100 ms of frame shifts, which may be too low for the short-segment SV. In [11], the features from the PTMs fused with a fine-grained fusion module (FGFM) were combined with the FBank features with a 10 ms temporal resolution to produce excellent performances.

To utilize the representation capability of the features from the PTMs while exploiting multiple fine temporal resolutions for short-segment SV, we propose to utilize not only the features from PTMs and FBank features but also the features from the MRE. The weighted summation of the features from the PTMs and the FBank features are used as the input of the ECAPA-TDNN [2] as in [11], while the MRE features are incorporated through adapter modules as in [5]. To focus more on the combination of these three features, we simply combine the layer-wise outputs of the PTMs by weighted summations. Experimental results on the VoxCeleb dataset showed that the adoption of the FBank features and MRE features were effective to improve the performance of the SV utilizing PTMs for various lengths of the input segments and outperformed previously proposed methods in short-segment SV. Moreover, an analysis of the learned weights to combine layer-wise outputs of the PTMs for different combinations of the features demonstrated that the MRE and FBank features encouraged PTM features to focus more on the high-level information.

This work was partly supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) - ITRC (Information Technology Research Center) grant funded by the Korea government (MSIT) (IITP-2025-RS-2021-II211835, 50%) and IITP grant funded by the Korea government (MSIT) (No.IRIS-2025-25443882, S.A.M.A.N.T.H.A: Sentient Audio Machine for Alive Natural Talking & Human Affection, 50%).
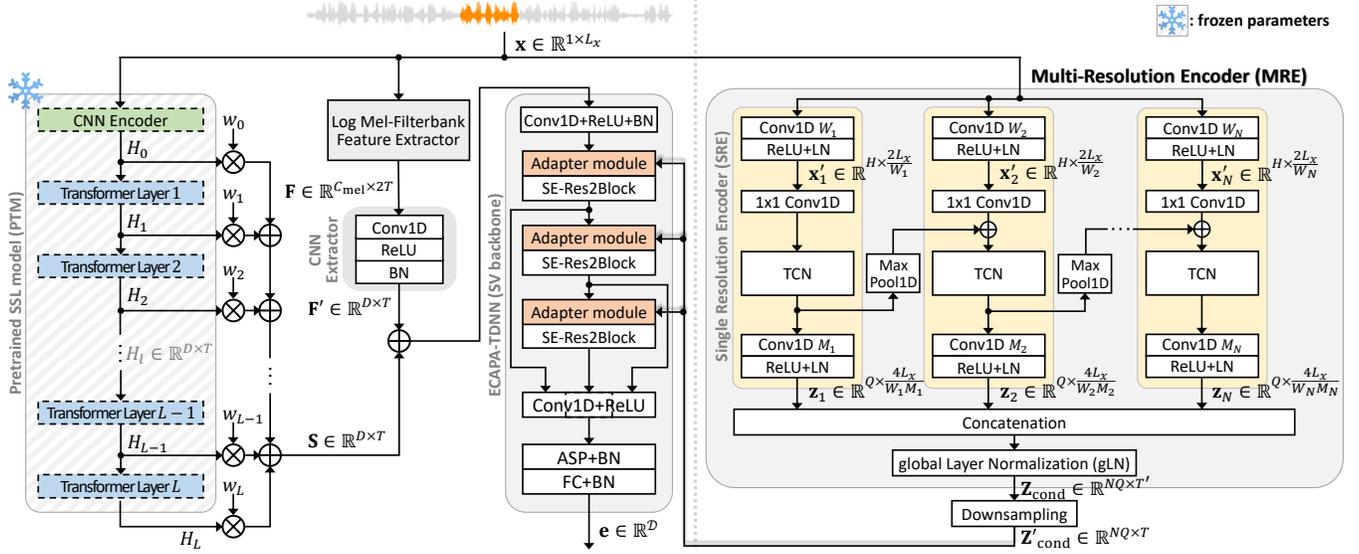
Fig. 1: Block diagram of the proposed short-segment speaker verification system with a pre-trained model and a multi-resolution encoder.

2. RELATED WORKS

2.1. Multi-Resolution Encoder (MRE)

In the short-time Fourier transform (STFT) analysis, a longer window yields a higher spectral resolution but a lower temporal resolution, while a smaller window leads to the opposite trade-off when the overlap ratio between adjacent windows is fixed. In analogy to the multi-resolution STFT analysis utilizing multiple windows with various sizes, the MRE [5] processes the input speech using N single resolution encoders (SREs), each of which analyzes the input signal using a convolutional layer with a different kernel and hop sizes as illustrated in the right part of Fig. 1. In the n-th SRE, the input speech signal $\mathbf{x} \in \mathbb{R}^{1 \times L_x}$ is first encoded with H kernels of size W_n and stride $W_n/2$ to produce $\mathbf{x}'_n \in \mathbb{R}^{H \times \frac{2L_x}{W_n}}$. The encoded features are converted to P channels through a 1×1 convolution and subsequently processed by temporal convolutional networks (TCNs) [21]. It is then convolved with Q kernels of size M_n and stride $M_n/2$ to align the temporal dimension by configuring M_n to satisfy $\frac{W_n}{2} \cdot \frac{M_n}{2} = \mathcal{S}$ where $\mathcal{S}$ is the frame shift and $T' = L_x/\mathcal{S}$ is the number of frames. The outputs $\mathbf{z}_n \in \mathbb{R}^{Q \times T'}$ of the N SREs are concatenated along the channel dimension and then normalized using global layer normalization (gLN) [22] to form a multi-resolution feature $\mathbf{Z}_{\text{cond}} \in \mathbb{R}^{NQ \times T'}$.

2.2. Integrated Filterbank and PTM Features

In [11], layer-wise speech representations from the PTMs are fused using the FGFM to form a PTM feature $\mathbf{S}$, which is then combined with a 80-dimensional FBank feature $\mathbf{F}$. $\mathbf{F}$ is extracted with a 25 ms window and a hop size of 10 ms, which makes the number of frames twice compared with those for PTM features. To match the temporal and channel dimensions, $\mathbf{F}$ is processed by a CNN extractor, which consists of a 1D convolutional layer, a ReLU activation function, and batch normalization (BN). The aligned feature $\mathbf{F}'$ and $\mathbf{S}$ are added together with two learnable parameters α and β to construct the integrated features, i.e.,

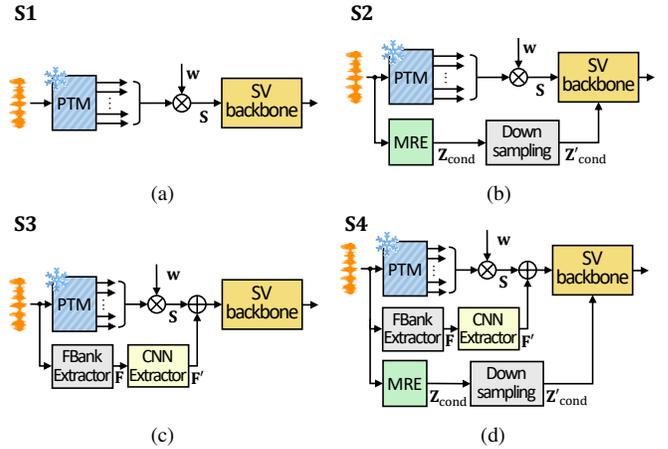$$\text{Integrated Features} = \alpha\mathbf{S} + \beta\mathbf{F}'. \tag{1}$$

Fig. 2: Four combinations of input features for speaker verification.

3. METHODS

Utilization of the MRE along with the FBank features was shown to be effective for short-segment SV [5], but the performance may be further improved if a large scale speech data is exploited through PTMs. The integrated features combining FBank and PTM features in [11] exhibited wonderful performance, but the temporal resolution was limited to 10 ms and 20 ms which may not be ideal for short-segment SV. In this paper, we propose to utilize PTM features and FBank features along with an MRE to extract as much information as possible from short utterances for SV. As a backbone SV system, we have used ECAPA-TDNN, although any other SV system can be tested. The block diagram of the proposed system is illustrated in Fig. 1, which can be simplified to Fig. 2d emphasizing combination of input features. Fig. 2 also shows other input feature combinations to utilize PTM features.

The input speech signal $\mathbf{x}$ is first fed into a log mel-filterbank feature extractor, an MRE module, and a PTM to extract FBank feature $\mathbf{F}$, multi-resolution features $\mathbf{Z}_{\text{cond}}$, and layer-wise speech repre-

Table 1: Speaker verification performance of the systems with different PTMs and input feature combinations with AS-norm.

Evaluation Set	PTM	System	EER (%)							MinDCF						
			full	5s	4s	3s	2s	1.5s	1s	full	5s	4s	3s	2s	1.5s	1s
VoxCeleb-O	wav2vec 2.0	S1	1.37	1.30	1.38	1.46	1.83	2.20	3.33	0.1009	0.1037	0.1043	0.1097	0.1309	0.1529	0.2277
		S2	1.20	1.23	1.24	1.39	**1.61**	1.95	2.93	0.0863	0.0857	0.0915	0.0913	0.1150	0.1338	**0.1942**
		S3	1.31	1.35	1.38	1.48	1.84	2.23	3.28	0.0941	0.0915	0.0950	0.1066	0.1232	0.1405	0.2184
		S4	**1.10**	**1.14**	**1.14**	**1.27**	1.61	**1.91**	**2.75**	**0.0840**	**0.0846**	**0.0820**	**0.0900**	**0.1034**	**0.1323**	0.1974
	HuBERT	S1	1.34	1.34	1.35	1.42	1.70	2.74	7.35	0.1007	0.0890	0.0871	0.0950	0.1261	0.1830	0.4476
		S2	1.32	1.25	1.22	1.36	1.65	2.57	5.89	**0.0875**	**0.0833**	**0.0820**	**0.0923**	0.1245	0.1930	0.3943
		S3	1.36	1.32	1.33	1.44	1.74	2.65	6.50	0.1017	0.0960	0.0970	0.1021	0.1276	0.1908	0.4102
		S4	**1.18**	**1.14**	**1.20**	**1.27**	**1.55**	**2.23**	**4.60**	0.0892	0.0903	0.0927	0.0988	**0.1105**	**0.1580**	**0.2978**
	WavLM	S1	0.97	0.96	0.98	1.05	1.36	1.60	2.49	0.0639	0.0659	0.0677	0.0829	0.0980	0.1219	0.1891
		S2	0.81	0.85	0.87	0.97	1.27	1.55	2.30	0.0614	0.0655	0.0687	0.0792	0.0954	0.1122	0.1692
		S3	0.85	0.89	0.93	1.01	1.28	1.51	2.31	0.0606	0.0651	0.0703	0.0756	0.0962	0.1185	0.1761
		S4	**0.77**	**0.76**	**0.80**	**0.89**	1.21	**1.40**	**2.24**	**0.0570**	**0.0556**	**0.0601**	**0.0644**	**0.0843**	**0.1093**	**0.1590**
VoxCeleb-E	wav2vec 2.0	S1	1.39	1.39	1.42	1.53	1.81	2.15	3.15	0.0881	0.0881	0.0907	0.0976	0.1194	0.1428	0.2003
		S2	1.30	1.30	**1.33**	**1.43**	**1.66**	**1.96**	2.82	0.0819	**0.0810**	0.0842	**0.0909**	0.1092	0.1285	0.1842
		S3	1.46	1.47	1.50	1.63	1.92	2.29	3.29	0.0900	0.0899	0.0925	0.1008	0.1213	0.1448	0.2059
		S4	**1.28**	**1.29**	**1.33**	1.44	1.68	1.97	**2.80**	**0.0814**	0.0836	**0.0841**	0.0923	**0.1080**	**0.1276**	**0.1777**
	HuBERT	S1	1.76	1.68	1.67	1.77	2.04	3.22	9.11	0.1160	0.1045	0.1043	0.1138	0.1350	0.2073	0.4872
		S2	1.28	1.23	1.25	1.33	1.58	2.26	5.35	0.0839	0.0803	0.0813	0.0874	0.1041	0.1503	0.3254
		S3	1.34	1.30	1.32	1.41	1.67	2.47	6.35	0.0870	0.0843	0.0856	0.0922	0.1098	0.1596	0.3646
		S4	**1.19**	**1.18**	**1.20**	**1.29**	**1.52**	**2.12**	**4.30**	**0.0748**	**0.0743**	**0.0767**	**0.0831**	**0.0983**	**0.1410**	**0.2736**
	WavLM	S1	1.05	1.05	1.07	1.16	1.37	1.65	2.44	0.0672	0.0664	0.0678	0.0735	0.0906	0.1103	0.1612
		S2	1.03	1.02	1.05	1.12	1.33	1.58	2.33	0.0640	0.0664	0.0661	0.0727	0.0872	0.1060	0.1548
		S3	0.98	0.99	1.01	1.09	1.29	1.56	2.37	0.0632	0.0628	0.0640	0.0692	0.0859	0.1044	0.1568
		S4	**0.96**	**0.96**	**0.98**	**1.05**	**1.24**	**1.50**	**2.18**	**0.0599**	**0.0606**	**0.0622**	**0.0672**	**0.0819**	**0.0973**	**0.1450**
VoxCeleb-H	wav2vec 2.0	S1	2.55	2.56	2.62	2.82	3.30	3.89	5.47	0.1445	0.1486	0.1517	0.1641	0.1965	0.2287	0.3137
		S2	2.37	2.37	2.41	2.59	3.04	3.53	4.89	0.1384	**0.1381**	0.1416	0.1554	0.1835	0.2129	0.2840
		S3	2.62	2.66	2.72	2.93	3.43	4.01	5.50	0.1460	0.1485	0.1544	0.1655	0.1975	0.2303	0.3112
		S4	**2.31**	**2.33**	**2.39**	**2.58**	**3.02**	**3.49**	**4.79**	**0.1341**	0.1396	**0.1409**	**0.1523**	**0.1776**	**0.2064**	**0.2768**
	HuBERT	S1	3.17	3.00	2.98	3.16	3.67	5.45	12.55	0.1817	0.1737	0.1747	0.1873	0.2182	0.3125	0.6222
		S2	2.33	2.25	2.26	2.40	2.87	3.99	8.00	0.1387	0.1340	0.1348	0.1458	0.1740	0.2407	0.4382
		S3	2.41	2.35	2.37	2.51	2.96	4.31	9.28	0.1415	0.1387	0.1412	0.1506	0.1785	0.2530	0.4896
		S4	**2.18**	**2.16**	**2.21**	**2.36**	**2.77**	**3.80**	**6.79**	**0.1263**	**0.1268**	**0.1297**	**0.1393**	**0.1663**	**0.2274**	**0.3884**
	WavLM	S1	2.10	2.09	2.12	2.28	2.68	3.17	4.46	0.1271	0.1280	0.1309	0.1430	0.1688	0.1994	0.2718
		S2	2.01	2.01	2.05	2.22	2.60	3.02	4.30	0.1175	0.1174	0.1207	0.1311	0.1609	0.1867	0.2570
		S3	1.97	2.00	2.05	2.20	2.57	3.03	4.35	0.1171	0.1198	0.1215	0.1317	0.1579	0.1866	0.2590
		S4	**1.89**	**1.89**	**1.92**	**2.08**	**2.43**	**2.86**	**4.02**	**0.1115**	**0.1129**	**0.1151**	**0.1253**	**0.1496**	**0.1736**	**0.2417**

sentations $H_0, \ldots, H_L$. As for PTM features, $\mathbf{x}$ is processed by a CNN encoder and L Transformer layers of a PTM to produce $L + 1$ layer-wise speech representations $H_0, \ldots, H_L \in \mathbb{R}^{D \times T}$, where D and T denote the channel and the number of frames, respectively. The speech representations are then fused by a weighted summation with learnable parameters $\mathbf{w} = [w_0, \ldots, w_L]$ to form a PTM feature matrix $\mathbf{S}$, i.e.,

$$\mathbf{S} = \sum_{l=0}^{L} w_l H_l \in \mathbb{R}^{D \times T} \qquad (2)$$

As the FBank feature $\mathbf{F} \in \mathbb{R}^{C_{\text{mel}} \times 2T}$ has C_{mel} channels and twice the number of frames compared with $\mathbf{S}$ with 10 ms frame shift, the temporal and channel dimensions are adjusted using a CNN extractor as in [11] to form $\mathbf{F}' \in \mathbb{R}^{D \times T}$. $\mathbf{S}$ and $\mathbf{F}'$ are added up to form the input of the SV system without weights expecting the CNN extractor would control the relative weights for two features.

The MRE output $\mathbf{Z}_{\text{cond}} \in \mathbb{R}^{NQ \times T'}$ is downsampled to match the temporal dimension T to form $\mathbf{Z}'_{\text{cond}} \in \mathbb{R}^{NQ \times T}$. $\mathbf{Z}'_{\text{cond}}$ is then used to modify the hidden representations before each SE-Res2Block in the ECAPA-TDNN through adapter modules [5], i.e., to modify the

hidden representation $\mathbf{h} \in \mathbb{R}^{C \times T}$ with an affine transformation

$$\bar{\mathbf{h}} = \gamma \otimes \mathbf{h} + \beta \qquad (3)$$

where $\otimes$ means the element-wise multiplication and $\gamma \in \mathbb{R}^{C \times T}$ and $\beta \in \mathbb{R}^{C \times T}$ are obtained from $\mathbf{Z}'_{\text{cond}}$. The final output of the ECAPA-TDNN, $\mathbf{e}$, is used for SV.

4. EXPERIMENTAL SETTINGS

All systems were trained on the VoxCeleb2 development set [23], comprising more than 1 million utterances from 5,994 speakers. For evaluation, we employed the VoxCeleb1-O, VoxCeleb1-E, and VoxCeleb1-H test sets, which consist of 37,611, 579,818, and 550,894 verification trials spoken by 40, 1,251, and 1,251 speakers, respectively. There is no speaker overlap between the training and test sets. The SV decision is made by thresholding the cosine similarity between the embeddings from the test and enrollment utterances after optionally applying the adaptive s-norm (AS-norm) [24], [25]. The performance on short-segment SV was evaluated by averaging the performances for the experiments with

full-length enrollment utterances and short test utterances cropped in the middle, and vice versa. The equal error rate (EER) and the minimum of the detection cost function (MinDCF) with $P_{target} = 0.05$ and $C_{FalseAlarm} = C_{Miss} = 1$ were used as performance measures.

Three large PTMs, WavLM-Large [26], HuBERT Large [19], and wav2vec 2.0 Large [27] models[1] were adopted in the experiments. The parameters for PTMs were not updated. ECAPA-TDNN with $C = 512$ channels was used as the backbone SV model. All hyperparameters for the MRE module were set identically to those for the optimal configuration reported in [5], in which 4 SREs with the window shifts of 25, 50, 100, and 200 samples were used, which correspond to the temporal resolutions of 1.56, 3.13, 6.25, and 12.5 ms for the 16 kHz sampling rate, respectively. Each utterance was randomly cropped into 2-second segments to construct training batches, i.e., $L_x = 32,000$ samples in training. For FBank features, 80-dimensional log mel-filterbank energies were extracted using a 25 ms window and a 10 ms frame shift. We applied data augmentation using simulated room impulse responses (RIRs), MUSAN noises [28], and speed perturbation with factors of 0.9 and 1.1. Voice activity detection (VAD) was not applied. The models were optimized using AdamW with a cyclical learning rate policy [29], where the maximum and minimum learning rates were set to 1×10^{-3} and 1×10^{-8}, respectively, and the maximum learning rate was reduced by half every cycle. We adopted AAM-Softmax [30] as the loss function with a margin of 0.2 and a scale of 30. Training was conducted for six cycles of 5 epochs each, totaling 30 epochs. All systems were implemented in PyTorch framework and trained on two NVIDIA RTX 3090 GPUs.

5. RESULTS AND ANALYSIS

Table 1 presents the speaker verification performances with three PTMs and four input feature combinations shown in Fig. 2 using AS-norm for various input lengths. As can be seen in the Table, the proposed system, S4, showed the best performance for most of the cases when the PTM was wav2vec 2.0 or HuBERT and all cases when the PTM was WavLM. Among three PTMs, WavLM exhibited the uniformly best performance. Introducing MRE features lead to performance improvement in most of the cases, which can be observed from the performance improvement of S2 over S1 and that of S4 over S3. On the other hand, the introduction of FBank features (S3 compared with S1, and S4 compared with S2) was not very effective for wav2vec 2.0, although it improved the SV performance for WavLM.

The WavLM-based system with four input feature combinations was also compared against previously proposed short-segment SV systems for the VoxCeleb1-O dataset, as shown in Table 2. The reported performances were obtained from the original papers [4]–[6], except for ECAPA-TDNN, where we used the re-implemented results in [5]. AS-norm was not applied in this experiment, as the compared systems did not utilize it. The "Utilized Features" columns are to show that the ECAPA-TDNN [2] and ECAPA-TDNN + MRE [5] can also be though as a different input feature combination for the same system, although the ECAPA-TDNN in [2] and [5] was with $C = 1024$ channels instead of 512 channels. These columns are not meaningful for RawNext [4] or MR-RawNet [6], but the MRE column is checked for MR-RawNet [6] to show that it uses the MRFE which is similar to the MRE. The proposed S4 system consistently outperformed all other systems for various input lengths including ECAPA-TDNN + MRE [5], which previously showed the best per-

1huggingface.co/models

Table 2: Performance comparison with previous approaches for short segment speaker verification on VoxCeleb1-O without AS-norm. [†]: re-implementation.

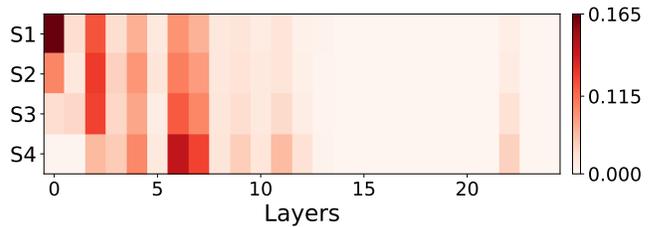| System	Utilized Features			EER (%)				
	PTM	FBank	MRE	full	5s	2s	1.5s	1s
RawNext [4]	–	–	–	1.29	1.45	2.34	–	4.47
MR-RawNet [6]	–	–	✓	0.83	0.99	1.61	–	3.47
ECAPA-TDNN[†] [2]	✗	✓	✗	1.03	1.05	1.76	2.12	3.04
+ MRE [5]	✗	✓	✓	1.01	1.03	1.32	1.60	2.33
S1	✓	✗	✗	0.98	1.00	1.40	1.65	2.52
S2	✓	✗	✓	0.88	0.91	1.30	1.56	2.36
S3	✓	✓	✗	0.89	0.92	1.32	1.54	2.38
S4	✓	✓	✓	**0.78**	**0.81**	**1.24**	**1.46**	**2.29**

Fig. 3: Learned layer-wise weights for WavLM-Large features in the systems with different combinations of input features.

formance for the input length of 2 s or shorter. With the PTM features alone, S1 did not outperform ECAPA-TDNN + MRE [5] for short-segment SV.

To investigate the effect of the additional features to the PTM-based SV system, we plot the learned layer-wise weights, $\mathbf{w} = [w_0, \dots, w_L]$, for four input combinations when the PTM was the WavLM in Fig. 3. The learned weights for S1 showed that w_0 had the biggest value, which was similar to the weights reported in the original WavLM paper [26]. With the additional MRE or FBank features, S2 and S3 emphasized higher level informations more than H_0. With both MRE and FBank features, the layer-wise weights for the proposed S4 system H_0 was barely used, which may imply that the low-level information was exploited enough through the MRE and FBank features. Instead, higher level information can be more emphasized in $\mathbf{S}$ to complement $\mathbf{F}$ and $\mathbf{Z}_{cond}$.

6. CONCLUSION

In this paper, we have introduced a short-segment SV system which utilizes the self-supervised learning models trained with large-scale databases and supplements their shortcoming for the short-segment SV, insufficient temporal resolution, by utilizing log mel-filterbank features and a multi-resolution encoder with various kernel sizes. The features from the PTMs are combined with the FBank features to form the input tensor for the backbone SV model, while the features generated by the MRE are used to modify the hidden representations within the SV model. Experimental results on the Vox-Celeb dataset with WavLM for various input lengths demonstrated that the proposed system outperformed previously proposed methods on short-segment SV. Furthermore, an analysis of the learned weights to combine layer-wise outputs of the WavLM for different combinations of the input features revealed that the inclusion of the MRE and FBank features lead PTM representations to focus more on the high-level information.

7. REFERENCES

[1] S. Han, Y. Ahn, and J. W. Shin, "Noise-robust speaker verification with attenuated speech restoration and consistency training," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 33, pp. 1987–1997, 2025.

[2] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *Proc. Interspeech*, 2020, pp. 3830–3834.

[3] T. Liu, R. K. Das, A. L. Kong, and H. Li, "MFA: TDNN with multi-scale frequency-channel attention for text-independent speaker verification with short utterances," in *Proc. ICASSP*, 2022.

[4] J.-H. Kim, H.-J. Shim, J. Heo, and H.-J. Yu, "RawNeXt: Speaker verification system for variable-duration utterances with deep layer aggregation and extended dynamic scaling policies," in *Proc. ICASSP*, 2022, pp. 7647–7651.

[5] S. Han, Y. Ahn, K. Kang, and J. W. Shin, "Short-segment speaker verification using ecapa-tdnn with multi-resolution encoder," in *Proc. ICASSP*, 2023, pp. 181–185.

[6] S. B. Kim, C. Y. Lim, J. Heo, J. H. Kim, H. S. Shin, K. W. Koo, and H. J. Yu, "MR-RawNet: Speaker verification system with multiple temporal resolutions for variable duration utterances using raw waveforms," in *Proc. Interspeech*, 2024, pp. 2125–2129.

[7] Y. Chen, S. Zheng, H. Wang, L. Cheng, Q. Chen, S. Zhang, and J. Li, "ERes2NetV2: Boosting short-duration speaker verification performance with computational efficiency," in *Proc. Interspeech*, 2024, pp. 3245–3249.

[8] M. Sang and J. H. L. Hansen, "Efficient adapter tuning of pretrained speech models for automatic speaker verification," in *Proc. ICASSP*, 2024, pp. 12 131–12 135.

[9] Z. Chen, S. Chen, Y. Wu, Y. Qian, C. Wang, and S. Liu, "Large-scale self-supervised speech representation learning for automatic speaker verification," in *Proc. ICASSP*, 2022, pp. 6147–6151.

[10] J. S. Kim, H. J. Park, W. Shin, and S. W. Han, "Rethinking leveraging pre-trained multi-layer representations for speaker verification," in *Proc. Interspeech*, 2025, pp. 3713–3717.

[11] Y. Li, W. Guan, H. Huang, S. Miao, Q. Su, L. Li, and Q. Hong, "Efficient integrated features based on pre-trained models for speaker verification," in *Proc. Interspeech*, 2024, pp. 2140–2144.

[12] S. Peng, W. Guo, H. Wu, Z. Li, and J. Zhang, "Fine-tune pretrained models with multi-level feature fusion for speaker verification," in *Proc. Interspeech*, 2024, pp. 2110–2114.

[13] S. Peng, W. Guo, J. Zhang, Z. Li, Y. Guan, B. Gu, and Y. Ai, "A study of multi-scale feature learning from pre-trained models on speaker verification," in *Proc. ICASSP*, 2025, pp. 1–5.

[14] J. Peng, O. Plchot, T. Stafylakis, L. Mošner, L. Burget, and J. Černocký, "An attention-based backend allowing efficient fine-tuning of transformer models for speaker verification," in *Proc. SLT*, 2023, pp. 555–562.

[15] Z. Li, M.-W. Mak, and H. M.-L. Meng, "Dual parameter-efficient fine-tuning for speaker representation via speaker prompt tuning and adapters," in *Proc. ICASSP*, 2024, pp. 10 751–10 755.

[16] F. Hao, X. Li, and C. Zheng, "X-TF-GridNet: A time–frequency domain target speaker extraction network with adaptive speaker embedding fusion," *Information Fusion*, vol. 112, p. 102550, 2024.

[17] J. Byun and J. W. Shin, "Monaural speech separation using speaker embedding from preliminary separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 2753–2763, Aug. 2021.

[18] J. Shi, Y. Tang, H. Inaguma, H. Gong, J. Pino, and S. Watanabe, "Exploration on HuBERT with multiple resolution," in *Proc. Interspeech*, 2023, pp. 3287–3291.

[19] 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," in *Proc. ACL*, 2021.

[20] J. Shi, H. Inaguma, X. Ma, I. Kulikov, and A. Sun, "Multiresolution HuBERT: Multi-resolution speech self-supervised learning with masked unit prediction," in *Proc. ICLR*, 2024.

[21] S. Han, J. Byun, and J. W. Shin, "Time-domain speaker verification using temporal convolutional networks," in *Proc. ICASSP*, 2021, pp. 6688–6692.

[22] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[23] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep speaker recognition," in *Proc. Interspeech*, 2018, pp. 1086–1090.

[24] S. Cumani, P. D. Batzu, D. Colibro, C. Vair, P. Laface, and V. Vasilakakis, "Comparison of speaker recognition approaches for real applications," in *Proc. Interspeech*, 2011, pp. 2365–2368.

[25] S. Shum, N. Dehak, R. Dehak, and J. Glass, "Unsupervised speaker adaptation based on the cosine similarity for text-independent speaker verification," in *Proc. Odyssey*, 2010.

[26] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen *et al.*, "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *IEEE J. Sel. Top. Signal Process.*, vol. 16, no. 6, pp. 1505–1518, 2022.

[27] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. NeurIPS*, vol. 33, 2020, pp. 12 449–12 460.

[28] S. David, C. Guoguo, and P. Daniel, "MUSAN: A music, speech, and noise corpus," Oct 2015, arXiv:1510.08484.

[29] L. N. Smith, "Cyclical learning rates for training neural networks," in *Proc. WACV*, 2017, pp. 464–472.

[30] D. Jiankang, G. Jia, Y. Jing, X. Niannan, K. Irene, and Z. Stefanos, "ArcFace: Additive angular margin loss for deep face recognition," 2018, arXiv:1801.07698.