Joint domain adaptation and speech bandwidth extension using time-domain GANs for speaker verification

Saurabh Kataria1,2, Jesús Villalba1,2, Laureano Moro-Velázquez1, Najim Dehak1,2

1Center for Language and Speech Processing, 2Human Language Technology Center of Excellence
Johns Hopkins University, Baltimore, MD, USA

{skatari1,jvillal7,laureano,ndehak3}@jhu.edu

Abstract

Speech systems developed for a particular choice of acoustic domain and sampling frequency do not translate easily to others. The usual practice is to learn domain adaptation and bandwidth extension models independently. Contrary to this, we propose to learn both tasks together. Particularly, we learn to map narrowband conversational telephone speech to wideband microphone speech. We developed parallel and non-parallel learning solutions which utilize both paired and unpaired data. We first discuss joint and disjoint training of multiple generative models for our tasks. Then, we propose a two-stage learning solution using a pre-trained domain adaptation system for pre-processing in bandwidth extension training. We evaluated our schemes on a Speaker Verification downstream task. We used the JHU-MIT experimental setup for NIST SRE21, which comprises SRE16, SRE-CTS Superset, and SRE21. Our results prove that learning both tasks is better than learning just one. On SRE16, our best system achieves 22% relative improvement in Equal Error Rate w.r.t. a direct learning baseline and 8% w.r.t. a strong bandwidth expansion system.

Index Terms: domain adaptation, speech bandwidth extension, time-domain GAN, non-parallel learning, joint learning

1. Introduction

Deep learning-powered speech systems are becoming ubiquitous. We usually design them for a particular choice of sampling frequency and acoustic domain. To deploy them in a different condition, we typically re-train them with relevant data or develop pre-processing solutions [1]. Commonly, two fundamental speech tasks, namely bandwidth extension (BWE) and domain adaptation (DA) are pursued for this. Bandwidth extension refers to predicting missing higher-frequency (upper-band) information, and domain adaptation refer to the set of techniques aimed at improving the robustness of the model to the choice of testing domain. Both tasks are greatly beneficial [2–7] but usually only one of both is employed.

Supervised and unsupervised learning can both address bandwidth extension. For supervised learning, mapping [2] (deep regression) is typically used. Common choices of generative models include Conditional Generative Adversarial Network (CGAN) [8] and Cycle-consistent GAN (CycleGAN) [9, 10] respectively. Domain adaptation is an extensively studied problem as well. [11] studies Multi-Source Domain Adaptation (MSDA) using Domain Adversarial Training (DAT) [12]. In [13], the authors perform an unsupervised training of a Probabilistic Linear Discriminant Analysis (PLDA) classifier. CycleGANs are also popular for domain adaptation [5–7, 11–16].

However, there are limited studies for joint learning of bandwidth extension or domain adaptation with other tasks. [17] and [18] first proposed joint training of speech denoising and bandwidth extension. In [19], authors learn domain adaptation and discriminative feature learning. In [20], authors learn class alignment in addition to domain alignment. To our knowledge, there is no study for joint learning of bandwidth extension and domain adaptation. We hypothesize that learning them together brings further gains in performance. We propose to use them as pre-processing solutions to improve the downstream task of Automatic Speaker Verification (ASV) –which aims at determining if the speakers in two recordings are identical or not. We are interested in Telephony Speaker Verification, i.e., our test sets (except SRE21) are composed of only telephone signals with a sampling frequency of 8kHz. However, the training data for the front-end (x-vector network [21]) and back-end (PLDA [22]) of ASV uses microphone speech, which has a sampling frequency of 16 kHz and has a mismatch w.r.t. telephone domain. We propose to project all signals to a richer space (corresponding to 16kHz wideband microphone speech) and follow a test-time extension and adaptation.

Our main contributions are: (1) We provide the first study for joint learning of domain adaptation and bandwidth extension using time-domain GANs and establish strong baselines on real test sets; (2) We provide the proof-of-concept that learning of these tasks can be synergistic while leveraging synthetic narrowband microphone data; (3) We provide formulations of joint training of supervised and unsupervised GANs and report significant improvements on SRE16 test set with our best scheme.

2. Supervised and unsupervised GANs

Here, we describe the Generative Adversarial Network (GAN) [23] based models used in this work. GANs learn to sample from true data distribution without access to class labels – in our case, speaker labels. Typically, they consist of two networks: generator and discriminator. The discriminator learns to distinguish between real and fake/generated samples while the generator learns to “fool” the discriminator [23].

2.1. Conditional GAN

CGAN is a supervised/parallel learning GAN which learns a mapping from one domain (say A) to another (say B) using paired data, i.e., corresponding samples from two domains. \( G_{A\rightarrow B} \) and \( D_B \) denotes the generator and the discriminator respectively. In addition to the typical adversarial loss of GAN (\( L_{adv} \)), CGAN minimizes regression loss (\( L_{sup} \)) between ground truth and predicted output, which is weighted by a hyperparameter \( \lambda_{sup} \). The minimax formulation is:

\[
\min_{G_{A\rightarrow B}} \max_{D_B} \mathbb{E}_{\hat{x_A}\sim D_{A\rightarrow B}} \mathbb{E}_{y\sim D_Y} \mathbb{E}_{x\sim D_X} D_B(D_Y(G_{A\rightarrow B}((x, y)))) + \lambda_{sup} L_{sup},
\]

\[
\mathbb{E}_{x\sim D_X} L_{sup}(G_{A\rightarrow B}((x, y)), y).
\]

\[\text{CGAN}_{A,B} = L_{adv} + \lambda_{sup} L_{sup}.\] (2)
2.2. CycleGAN

Cycle-consistent GAN is an unsupervised/non-parallel learning GAN that learns mappings between two domains (say A and B) without pairing information. It can be seen as two CGANs with additional cycle loss \( L_{cy} \) and identity loss \( L_{id} \), which are weighted by hyper-parameters \( \lambda_{cy} \) and \( \lambda_{id} \) respectively. Cycle loss enforces semantic consistency in mapping and the identity loss is a regularizer. CycleGAN optimizes

\[
\min_{\varphi_{A \rightarrow B}, \varphi_{B \rightarrow A}} \max_{\Psi_A, \Psi_B} L_{cy}(\varphi_{A \rightarrow B}, \varphi_{B \rightarrow A}, \Psi_A, \Psi_B),
\]

The expression for \( L_{adv,A\rightarrow B} \) is identical to Eq. 3, while \( L_{adv,B\rightarrow A} \) is defined similarly as \( L_{adv,A\rightarrow B} \) with A and B swapped. Cycle and identity losses are given by

\[
L_{cy} = E_{a \sim p_A}[\|a - \varphi_{B \rightarrow A}(\varphi_{A \rightarrow B}(a))\|_1] + E_{b \sim p_B}[\|b - \varphi_{A \rightarrow B}(\varphi_{B \rightarrow A}(b))\|_1]
\]

\[
L_{id} = E_{a \sim p_A}[\|a - \varphi_{B \rightarrow A}(a)\|_1] + E_{b \sim p_B}[\|b - \varphi_{A \rightarrow B}(b)\|_1]
\]

2.3. Joint training of CycleGAN and conditional GAN

We propose to jointly train CycleGAN and CGAN to learn \( A \leftrightarrow B \) and \( B \rightarrow C \) mappings, respectively, where C is the third domain of interest. For this model (CycleGAN\(_{CGAN}\)), we optimize:

\[
\min_{\varphi_{A \rightarrow B}, \varphi_{B \rightarrow C}, \psi_A, \psi_B} \max_{\varphi_{C \rightarrow A}, \varphi_{A \rightarrow B}} L_{cy}(\varphi_{A \rightarrow B}, \varphi_{B \rightarrow C}, \psi_A, \psi_B) + \lambda_{cy}(L_{cy,C}) + \lambda_{id}(L_{id}) + \lambda_{adv}(L_{adv,C}).
\]

2.4. Joint training of two CycleGANs

We propose to train two CycleGANs that learn \( A \leftrightarrow B \) and \( B \leftrightarrow C \) mappings, i.e., we do not learn \( A \rightarrow C \) and \( C \rightarrow A \). For this model (CycleGAN\(_{CGAN}\)), we optimize:

\[
\min_{\varphi_{A \rightarrow B}, \varphi_{B \rightarrow C}, \psi_A, \psi_B} \max_{\varphi_{C \rightarrow A}, \varphi_{A \rightarrow B}} L_{cy}(\varphi_{A \rightarrow B}, \varphi_{B \rightarrow C}, \psi_A, \psi_B) + \lambda_{cy}(L_{cy,C}) + \lambda_{id}(L_{id}) + \lambda_{adv}(L_{adv,C}).
\]
Table 1: Results for direct learning schemes as EER / minDCF. Bold numbers denotes best performance among two halves of the table.

| Name of direct scheme | Model | SRE16-YUE-eval40 | SRE-CTS-superset-dev | SRE21-audio-eval |
|-----------------------|-------|------------------|----------------------|------------------|
| [narrowMic = SRE-superset-train] |       |                  |                      |                  |
| Implicit-unassisted [Baseline 1.1] | CycleGAN | 14.96 / 0.665 | 12.53 / 0.599 | 32.93 / 0.857 |
| Implicit-assisted | CycleGAN | 13.47 / 0.637 | 9.41 / 0.428 | 30.35 / 0.796 |
| Explicit-disjoint | CycleGAN + CGAN | 9.14 / 0.470 | 8.69 / 0.377 | 23.83 / 0.784 |
| Explicit-disjoint | CycleGAN + CycleGAN | 9.68 / 0.482 | 8.63 / 0.370 | 24.72 / 0.788 |
| Explicit-joint | CycleGAN + CGAN | 11.68 / 0.563 | 10.81 / 0.454 | 25.79 / 0.824 |
| Explicit-joint | CycleGAN + CycleGAN | 25.45 / 0.875 | 20.44 / 0.795 | 39.82 / 0.941 |

| [narrowMic = SRE-superset-train + SRE16-train] |       |                  |                      |                  |
| Implicit-unassisted [Baseline 1.2] | CycleGAN | 8.71 / 0.452 | 5.37 / 0.240 | 21.34 / 0.743 |
| Implicit-assisted | CycleGAN | 8.39 / 0.430 | 5.71 / 0.240 | 20.19 / 0.714 |
| Explicit-disjoint | CycleGAN + CGAN | 11.43 / 0.552 | 9.20 / 0.393 | 23.91 / 0.796 |
| Explicit-disjoint | CycleGAN + CycleGAN | 11.21 / 0.527 | 9.03 / 0.387 | 24.88 / 0.801 |
| Explicit-joint | CycleGAN + CGAN | 12.22 / 0.572 | 10.18 / 0.459 | 23.26 / 0.782 |
| Explicit-joint | CycleGAN + CycleGAN | 29.00 / 0.930 | 23.14 / 0.847 | 42.01 / 0.971 |

Table 2: Results for indirect learning scheme as EER / minDCF. M is obtained from Stage 1 (CycleGAN) training (See Fig. 2).

| Source data for Stage 2 model | Stage 2 Model | SRE16-YUE-eval40 | SRE-CTS-superset-dev | SRE21-audio-eval |
|-------------------------------|---------------|------------------|----------------------|------------------|
| narrowMic [Baseline 1.2] | CycleGAN | 8.71 / 0.452 | 5.37 / 0.240 | 21.34 / 0.743 |
| M(narrowMic) + narrowMic | CGAN | 6.90 / 0.368 | 5.21 / 0.222 | 20.88 / 0.711 |
| M(narrowMic) + narrowMic | CGAN | 6.75 / 0.358 | 5.37 / 0.221 | 20.09 / 0.688 |
| M(narrowMic) + narrowMic | CycleGAN | 7.73 / 0.407 | 5.37 / 0.227 | 21.65 / 0.706 |
| M(narrowMic) + narrowMic + narrowMic | CycleGAN | 7.51 / 0.392 | 5.00 / 0.202 | 18.77 / 0.679 |

3. Domain adaptation and bandwidth extension schemes

There are three domains of interest: narrowband telephone (domain A or narrowMic), narrowband microphone (domain B or narrowMic), and wideband microphone (domain C or wideMic). Mapping from narrowMic to narrowMic (A → B) refers to domain adaptation, while mapping from narrowMic to wideMic (B → C) refers to bandwidth extension. We are primarily interested in A → C but may learn more mappings depending on the scheme pursued. As per our experimental setup (Sec. 4), (A, B) is unpaired data, and hence, only unsupervised learning (via CycleGAN) is possible for A → B. (B, C) is paired data, and thus, supervised learning is suitable for B → C. Consequently, (A, C) represents unpaired data. It is possible to ignore pairing information in paired data and pursue unsupervised learning instead. Our proposed schemes are categorized into direct and indirect.

3.1. Direct learning schemes

Direct learning schemes do not involve any pre-training stages and work with original (i.e., not pre-processed) data available from three domains (Fig. 1). They are further categorized into four schemes. Implicit-unassisted is the simplest (yet most demanding) learning scheme that requires implicit learning of BWE and DA from A to C, i.e., mapping to and from domain B is not modeled and must be learned inherently without any assistance. Implicit-assisted is a relaxed variant of implicit-unassisted where data from domain B is combined with domain A, thereby making learning easier. Both implicit schemes use CycleGAN since A and B have no paired data. Explicit schemes use domain B data as intermediate targets to relax implicit learning but at the cost of learning more mappings (two instead of one, see Fig. 1). Explicit-disjoint learns BWE and DA independently and uses the two learned mapping together during inference. Explicit-joint learns the two tasks in the conventional multi-target joint learning manner. While the mapping from A to B has to be learned in unsupervised fashion by CycleGAN, the mapping from B to C can be learned in supervised fashion with CGAN since we can create paired data.

3.2. Indirect (two-stage) learning scheme

The indirect scheme involves two-stage learning (Fig. 2). In the first stage, domain adaptation is learned between narrowband data (narrowMic and narrowMic) using CycleGAN, which provides us M. In the second stage, bandwidth extension is learned between wideMic and narrowMic (but projected to telephone domain via M). Since narrowMic is synthetically created (Sec. 4) data, our motivation is to improve its utility via pre-processing. Our previous work [7] showed that domain adaptation pre-processing improves the training data. Hence, in this scheme, telephone and unmodified narrowband microphone data can be combined to increase the source domain data size in stage two for further gains.

4. Experimental Setup

We use off-the-shelf TasNet architecture for generators, which is a time-domain 1-D CNN [25, 26]. We derive discriminator architectures from [27]. For CGAN, it is a single-period network with four initial channels, while for other models, it is a multi-period network with eight initial channels. X-vector architecture is Light ResNet-34 [28, 29] and its input is 80-D log Mel-Frequency Filterbank (LFMB) features. X-vector/speaker embedding dimension is 256, and the PLDA dimension is 150. We used a smaller x-vector network (but with data augmentation) due to the high computational complexity of x-vector extraction in doing multiple experiments. The training data for narrowMic is SRE superset [30] (SRE-superset-train and SRE16-dev data (SRE16-train) [31, 32] which includes Tagalog and Cantonese (YUE) languages. The training data for narrowMic and wideMic domains are VoxCeleb [33] downsam-
Table 3: Comparison of proposed techniques with stronger baselines. Supervised and unsupervised BWE is accomplished via CGAN and CycleGAN respectively (source data = narrow_mic, target data = wide_mic). Bold numbers denote overall best results.

| BWE DA | DA and/or BWE scheme | SRE16-YUE-eval40 | SRE-CTS-superset-dev | SRE21-audio-eval |
|---------|----------------------|-------------------|----------------------|------------------|
| ✓ ✓     | Supervised BWE       | 7.46 / 0.382      | 5.42 / 0.217         | 18.52 / 0.662    |
| ✓ ✓     | Unsupervised BWE     | 7.35 / 0.376      | 4.90 / 0.205         | 17.92 / 0.648    |
| ✓ ✓     | Supervised DA        | 8.62 / 0.437      | 5.67 / 0.249         | 20.92 / 0.737    |
| ✓ ✓     | Unsupervised DA      | 11.50 / 0.532     | 9.13 / 0.387         | 24.25 / 0.800    |
| ✓ ✓     | Direct (implicit-assisted) | 8.39 / 0.430    | 5.71 / 0.240         | 20.19 / 0.714    |
| ✓ ✓     | Indirect (semi-supervised) | 6.75 / 0.358    | 5.37 / 0.221         | 20.09 / 0.688    |
| ✓ ✓     | Indirect (unsupervised w/ addn. data) | 7.51 / 0.392 | 5.00 / 0.202         | 18.77 / 0.679    |

5. Results

5.1. Direct learning schemes

To study the effect of using matching data in train and test, we conduct experiments with and without using SRE16-train. Consider the upper-half of Table 1 which does not use SRE16-train in narrow_tel. All schemes (except last) can outperform Baseline 1.1 (implicit-unassisted), which is the most straightforward scheme in terms of formulation. Explicit-disjoint gets the best results. Explicit-joint models give promising results, but their performance lags for two reasons. One, CycleGAN⊕ CGAN and CycleGAN⊕CycleGAN are computationally heavy models, and we could not tune their hyper-parameters. Two, CycleGAN⊕CycleGAN is a fully unsupervised model which is difficult to train compared to supervised models. Now, consider the upper half of Table 1 which uses SRE16-train. As expected, the baseline (1.2) is much stronger. Here, implicit-assisted achieves the best results while explicit learning schemes achieve performance to their counterparts from the upper half of the table. Hence, explicit schemes are promising and robust to narrow_tel choice. In explicit-disjoint schemes, replacing CGAN with CycleGAN retains performance, showing that paired and unpaired learning results can be close. Also, note that using time-domain GAN models brings significant challenges w.r.t. training stability and hyper-parameter choice.

5.2. Indirect (two-stage) learning scheme

Table 2 presents the results for the indirect scheme. Here, we include SRE16-train in narrow_mic. Unsupervised learning in the first stage (Fig. 2) gives us a pre-processor we can use during supervised training in the second stage. Thus, we can do supervised training while (indirectly) using unpaired (telephone) data. CGAN models benefit from supervised learning and matching train-test data and perform best on the SRE16 test set. However, CycleGAN, which can leverage any amount of unsupervised data, gets the best performance on the other two test sets using additional source data.

5.3. Comparison of all techniques

We compare proposed techniques to stronger baselines in Table 3. The first row does not involve BWE or DA. The second and third row does only BWE using CGAN and CycleGAN, respectively. We observe that BWE is essential to reducing EER/minDCF. The fourth row does only unsupervised DA using CycleGAN and degrades performance. Hence, sampling frequency mismatch is a more significant challenge in our experimental setup than domain mismatch. An unsupervised method such as direct (implicit-assisted) can beat unsupervised baselines (row 3 and 4), but not supervised method (row 2). It is important to note that supervised BWE (row 2) is a very strong baseline obtained via extensive tuning. All other models in this work follow similar hyper-parameters as row 2, thereby giving sub-optimal results. Indirect (semi-supervised) is from Table 2 which uses CycleGAN (for pre-training) and CGAN, and hence is termed semi-supervised. It gives the best SRE16 test performance (8% relative EER reduction w.r.t. row 2 and 22% EER reduction w.r.t. baseline 1.2). However, it degrades performance on SRE21, which the indirect model retains by using additional data (last row). We find the SRE21 test set to be challenging. Finally, our schemes can beat three out of six metrics (w.r.t. strongest baseline), demonstrating their potential. Additional tuning, critical for GAN models, can bring in more improvements. Since there is a mismatch in the BWE/DA and ASV training objective, speaker-identity preserving domain adaptation [7] can help.

6. Conclusion

We investigated joint learning of bandwidth extension and domain adaptation for improving telephony speaker verification. We proposed two types of joint learning (direct and indirect) and provided schematic illustrations. Under direct learning scheme, we noted how synthetically created narrowband microphone data could be used in multiple ways, especially with joint training of multiple generative models like Conditional GAN and CycleGAN. Under the indirect learning scheme, we saw how multi-stage learning could outperform strong baselines. We also saw the critical role of the choice of telephone training data. In the future, we can analyze the role of languages in the training and test sets, which seems to play a significant role.
7. References

[1] P. García, J. Villalba, H. Bredin, J. Du, D. Castan, A. Cristia, L. Bullock, L. Guo, K. Okabe, P. S. Nidadavolu et al., “Speaker detection in the wild: Lessons learned from jslt 2019,” arXiv preprint arXiv:1912.00693, 2019.

[2] K. Li and C.-H. Lee, “A deep neural network approach to speech bandwidth expansion,” in 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015, pp. 4395–4399.

[3] P. S. Nidadavolu, C.-I. Lai, J. Villalba, and N. Dekah, “Investigation on bandwidth extension for speaker recognition,” in INTERSPEECH, 2018, pp. 1111–1115.

[4] P. S. Nidadavolu, V. Iglesias, J. Villalba, and N. Dekah, “Investigation on neural bandwidth extension of telephone speech for improved speaker recognition,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 6111–6115.

[5] P. S. Nidadavolu, J. Villalba, and N. Dekah, “Cycle-gans for domain adaptation of acoustic features for speaker recognition,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 6206–6210.

[6] P. S. Nidadavolu, S. Kataria, J. Villalba, and N. Dekah, “Low-resource domain adaptation for speaker recognition using cycle-gans,” in 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). IEEE, 2019, pp. 710–717.

[7] S. Kataria, J. Villalba, P. Želasko, L. Moro-Velázquez, and N. Dekah, “Deep feature cyclengans: Speaker identity preserving non-parallel microphone-telephone domain adaptation for speaker verification,” arXiv preprint arXiv:2104.01433, 2021.

[8] J. Su, Y. Wang, A. Finkelstein, and Z. Jin, “Bandwidth extension is all you need,” in ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021, pp. 696–700.

[9] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2223–2232.

[10] D. Haws and X. Cui, “Cyclegan bandwidth extension acoustic modeling for automatic speech recognition,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 6780–6784.

[11] Z. Wang and J. H. Hansen, “Multi-source domain adaptation for text-independent forensic speaker verification,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2021.

[12] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, “Domain-adversarial training of neural networks,” The journal of machine learning research, vol. 17, no. 1, pp. 2096–2030, 2016.

[13] D. García-Romero, A. McCree, S. Shum, N. Brunner, and C. Vaqueiro, “Unsupervised domain adaptation for i-vector speaker recognition,” in Proceeding of Odyssey: The Speaker and Language Recognition Workshop, vol. 8, 2014.

[14] P. S. Nidadavolu, S. Kataria, J. Villalba, P. García-Perera, and N. Dekah, “Unsupervised feature enhancement for speaker verification,” in ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020, pp. 7599–7603.

[15] P. S. Nidadavolu, S. Kataria, P. García-Perera, J. Villalba, and N. Dekah, “Single channel far field feature enhancement for speaker verification in the wild,” arXiv preprint arXiv:2005.08331, 2020.

[16] E. Hosseini-Asl, Y. Zhou, C. Xiong, and R. Socher, “A multi-discriminator cycleGAN for unsupervised non-parallel speech domain adaptation,” arXiv preprint arXiv:1804.00322, 2018.

[17] M. L. Seltzer, A. Acero, and J. Droppo, “Robust bandwidth extension of noise-corrupted narrowband speech,” in Ninth European conference on speech communication and technology. Citeseer, 2005.

[18] N. Hou, C. Xu, J. T. Zhou, E. S. Chng, and H. Li, “Multi-task learning for end-to-end noise-robust bandwidth extension,” in INTERSPEECH, 2020, pp. 4069–4073.

[19] C. Chen, Z. Chen, B. Jiang, and X. Jin, “Joint domain alignment and discriminative feature learning for unsupervised deep domain adaptation,” in Proceedings of the AAAI conference on artificial intelligence, vol. 33, no. 01, 2019, pp. 3296–3303.

[20] S. Li, C. H. Liu, B. Xie, L. Su, Z. Ding, and G. Huang, “Joint adversarial domain adaptation,” in Proceedings of the 27th ACM International Conference on Multimedia, 2019, pp. 729–737.

[21] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, “X-vectors: Robust deep embeddings for speaker recognition,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 5329–5333.

[22] J. Villalba, N. Chen, D. Snyder, D. Garcia-Romero, A. McCree, G. Sell, J. Borgstrom, F. Richardson, S. Shon, F. Grondin et al., “State-of-the-art speaker recognition for telephone and video speech: The jhu-mitsub for nist sre18,” in Interspeech, 2019, pp. 1488–1492.

[23] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial networks,” arXiv preprint arXiv:1406.2661, 2014.

[24] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, “Least squares generative adversarial networks,” in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2794–2802.

[25] Y. Luo and N. Mesgarani, “Conv-tasnet: Surpassing ideal time-frequency magnitude masking for speech separation,” IEEE/ACM transactions on audio, speech, and language processing, vol. 27, no. 8, pp. 1256–1266, 2019.

[26] J. Villalba, B. J. Borgstrom, S. Kataria, M. Rysicka, C. D. Castillo, J. Cho, L. P. García-Perera, P. A. Torres-Carrasquillo, and N. Dekah, “Advances in Cross-Lingual and Cross-Source Audio-Visual Speaker Recognition: The JHU-MIT System for NIST SRE21,” in Proc. The Speaker and Language Recognition Workshop (Odyssey 2022), 2022, pp. 213–220.

[27] J. Kong, J. Kim, and J. Bae, “Hi-fi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis,” Advances in Neural Information Processing Systems, vol. 33, pp. 17 022–17 033, 2020.

[28] J. Villalba, D. Garcia-Romero, N. Chen, G. Sell, J. Borgstrom, A. McCree, L. Garcia-Perera, S. Kataria, P. S. Nidadavolu, P. A. Torres-Carrasquillo et al., “Advances in speaker recognition for telephone and audio-visual data: the jhu-mit submission for nist sre19,” in Proceedings of Odyssey, 2020.

[29] J. Villalba, D. Garcia-Romero, N. Chen, G. Sell, J. Borgstrom, A. McCree, D. Snyder, S. Kataria, P. Garcia-Perera, F. Richardson et al., “The jhu-mitsub system description for nist sre19 av,” in NIST SRE19 Workshop, 2019.

[30] S. O. Sadjadi, “Nist sre cts superset: A large-scale dataset for telephony speaker recognition,” arXiv preprint arXiv:2108.07118, 2021.

[31] D. Reynolds, E. Singer, S. O. Sadjadi, T. Khayrkhal, A. Tong, C. Greenberg, L. Mason, and J. Hernandez-Cordero, “The 2016 nist speaker recognition evaluation,” MIT Lincoln Laboratory Lexington United States, Tech. Rep., 2017.

[32] J. Villalba, J. Borgstrom, S. Kataria, J. Cho, P. A. Torres-Carrasquillo, and N. Dekah, “The jhu-mitsub system description for nist sre20 ct challenges,” 2021.

[33] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, “ Voxceleb: Large-scale speaker verification in the wild,” Computer Speech & Language, vol. 60, p. 101027, 2020.

[34] D. Snyder, G. Chen, and D. Povey, “Musan: A music, speech, and noise corpus,” arXiv preprint arXiv:1510.08484, 2015.