Detecting Unknown Speech Spoofing Algorithms with Nearest Neighbors

Jingze Lu[1,2], Yuxiang Zhang[1,2], Zhuo Li[1,2], Zengqiang Shang[1,2], WenChao Wang[1] and Pengyuan Zhang[1,2,*]

[1] *Key Laboratory of Speech Acoustics and Content Understanding, Institute of Acoustics, CAS, No.21 North 4th Ring West Road, Haidian District, Beijing, China*

[2] *University of Chinese Academy of Sciences, No.1 Yanqihu East Road, Huairou District, Beijing, China*

Abstract

The development of deep speech generation technology has increased the risk of people being exposed to malicious or misleading information. From a defensive perspective, merely distinguishing between genuine and fake utterances is not enough. At the vocoder level, the artifacts in different frequency bands make it possible to distinguish between different synthesis methods. A reliable model should not only classify synthesis algorithms correctly, but also be able to identify samples that have not been seen. The second Audio Deepfake Detection Challenge (ADD2023) set up Track3 (Deepfake Algorithms Recognition) to simulate such a scenario. The challenge motivates researchers to construct systems that are robust enough for In-Distribution (ID) and Out-Of-Distribution (OOD) utterances. Cosine similarity based kNN distance is introduced in this work to separate unknown samples from known ones. Together with data augmentation methods and logits based model fusion, our system wins first place in ADD2023 Track3.

Keywords

Deepfake Detection, Algorithms Recognition, Out-Of-Distribution Detection, ADD Challenge

1. Introduction

With the massive popularity of the Internet, audio and video applications have a broad market. A large amount of short-form while low-cost audio and videos are rapidly occupying people's attention. However, these audio and videos, which flood the Internet, have an impact on network security. Various AI-supported advanced algorithms are making voice and image generation easy to implement, even for people without expertise. With the development of deep learning, text-to-speech (TTS) [1] and voice conversion (VC) [2] techniques can generate speech indistinguishable from the human voice. Frauds based on these kinds of techniques have occurred from time to time in recent years. Therefore, there is an urgent need to avoid people being misled by fake speech generated by such techniques. Meanwhile, these algorithms are also effective in deceiving Automatic speaker verification (ASV) [3] systems, which play an important role in data security and passing certification.

The research community has conducted extensive research on how to distinguish the speech generated by methods such as TTS and VC, from natural speech. These research includes exploration of feature extraction front-end, such as Short-Term Fourier Transform (STFT) [4], Constant Q Cepstral Coefficients (CQCC) [5] and Linear Frequency Cepstral Coefficients (LFCC) [6], and design of classification back-end, such as RawNet [7] and AASIST [8]. In addition to the basic paradigms, various deep-learning strategies have also been studied, such as pre-training and fine-tuning [9][10], and active learning [11].

Unlike the speech spoofing detection task, research on the attribution of speech synthesis methods is still in its infancy. [12] explores the ability of different features to distinguish synthesis methods. The differences between different generation methods are mainly reflected in the artifacts of the vocoders [13].

However, in the task of attributing algorithms, a reliable model should not only have the ability to classify, but also be able to identify Out-Of-Distribution (OOD) correctly. In real-world scenario, Deep Neural Network (DNN) based classifiers often struggle with OOD samples, which are far from the data distribution of the training set. The reason for this issue could be attributed to the overconfidence of DNN-based models. The second Audio Deepfake Detection Challenge (ADD2023) [14] set up Track3 (Deepfake Algorithms Recognition) to attract researchers to solve this problem. Our proposed work is mainly based on this challenge.

To eliminate the impact of OOD samples, various methods has been proposed [15][16]. Among them, distance-based methods have been demonstrated effective. k-th

IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis (DADA 2023), August 19, 2023, Macao, S.A.R

[*] Corresponding author.

lujingze@hccl.ioa.ac.cn (J. Lu); zhangyuxiang@hccl.ioa.ac.cn (Y. Zhang); li_zhuo@foxmail.com (Z. Li); shangzengqiang@hccl.ioa.ac.cn (Z. Shang); wangwenchao@hccl.ioa.ac.cn (W. Wang); zhangpengyuan@hccl.ioa.ac.cn (P. Zhang)

Nearest Neighbors (kNN) distance is adopted to detect OOD data in [17]. We find that for the task of detecting spoofing algorithms, the kNN distance based on cosine similarity can effectively detect samples from OOD algorithms. Therefore, kNN distance is introduced in this work to construct a class calibration module, which improves the performance of basic models significantly. In addition, we investigate different data augmentation and model fusion methods. All these methods help us achieve first place in ADD2023 Track3.

2. Method

The proposed work is based on Track 3 (Deepfake Algorithms Recognition) of ADD2023 Challenge. In this section, we investigate the basis of deepfake algorithms recognition, which is the artifacts introduced by vocoders located in different frequency bands. In addition, in Track3, OOD samples exist in the test set. A kNN-based OOD detection method is also proposed to identify samples from unknown counterfeit class.

2.1. Vocoder Artifacts

Before recognizing deepfake algorithms, what needs to be demonstrated is whether the utterances generated by different synthesis methods are distinguishable. In other words, on what level are they distinguishable.

Vocoder is a key component in the process of generating fake utterances, which converts features to sampling points. The quality of the vocoder determines the quality of the generated utterance. Vocoder residual artifacts located in different frequency bands could serve as markers for deepfake algorithms. For instance, non-ideal upsampling filters will leave aliasing artifacts in the high frequency part [18]. Figure 1 shows the impact of different vocoders on utterances at the frequency level. We reconstruct the same batch of natural speech using different vocoders, and calculate the average energy of the frames at each frequency point. From Figure 1, it can be analyzed that the artifacts carried by different vocoders are located in different frequency bands. Therefore, features that encode time-frequency information could be utilized to recognize deepfake algorithms.

2.2. KNN-based OOD Detection

The proposed KNN-based OOD detection method is a distance-based method, which leverages the distance between embeddings extracted by trained DNN-based models. The basis of the proposed method is an intuitive assumption that samples from the same class are closer in distance, while samples from different classes are farther apart.

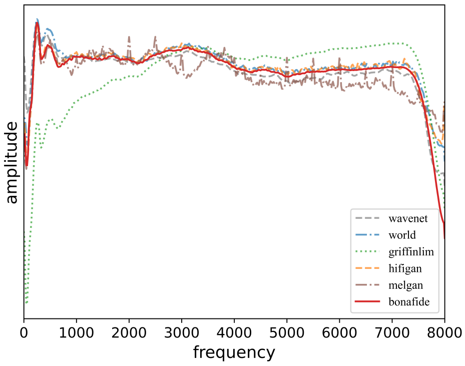

Figure 1: The average amplitude of different vocoders at each frequency bands.

We denote the sets of In-distribution (ID) data and Out-of-distribution (OOD) data as X_{in} and X_{ood}, respectively. The purpose of the algorithm is to distinguish a sample $x \in X$, where X donates the input space, is from X_{id} or X_{ood}. For this binary classification task, a direct solution is to set a mapping function $f(x)$ and a threshold λ.

$$x \in \begin{cases} X_{in}, f(x) \geq \lambda \\ X_{ood}, f(x) < \lambda \end{cases}$$

Based on the assumption that samples from different classes are farther apart, in this work, we leverage the k-th nearest neighbor (kNN) distance as the output of the mapping function $f(x)$, inspired by [17]. Compared to the 1st nearest neighbor (1NN) distance, under an appropriate k-value, kNN distance is less susceptible to noise samples. Cosine similarity is adopted to calculated the distance between the feature embeddings. Cosine similarity is defined as:

$$cos_{a,b} = \frac{z_a \cdot z_b}{\|z_a\|\|z_b\|}$$

where z_a and z_b are the embeddings of utterances extracted by models. Figure 2 shows the density of kNN distance of embeddings between ID data and OOD data. The ID data is from a known class of training set of ADD2023 Track3, and OOD data is from the other classes. kNN cosine distance of ID data is smaller than that of OOD data. Therefore, we could use a threshold-based criterion to determine whether the input utterance is OOD or not.

The pipeline of the method could be summarized as: (1) Train a multi-class DNN-based classifier with training dataset $\mathbb{D}_{train}$; (2) Use the trained model to pre-classify the test set $\mathbb{D}_{test}$; (3) Extract the feature embeddings of each samples from $\mathbb{D}_{train}$ and $\mathbb{D}_{test}$. (4) Select an appropriate k-value, and calculate the kNN cosine distance of

$\mathbb{D}_{train}$ of each class, and estimate a threshold; (5) Calculate the kNN cosine distance between $\mathbb{D}_{test}$ and $\mathbb{D}_{train}$ of each class, and attribute the OOD samples to a new unknown class based on a threshold-based criterion.

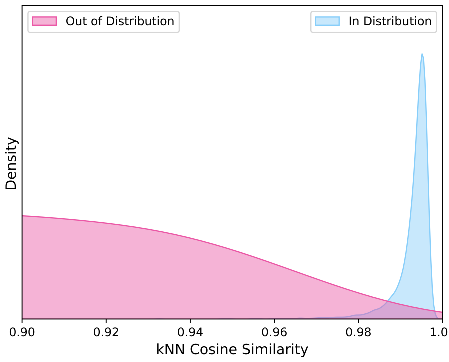

Figure 2: Density of kNN cosine distance of In-Distribution data and Out-of-Distribution data.

3. Experimental Setup

3.1. Dataset and Metrics

We used the training, development and test datasets of ADD2023 Challenge track 3 (Deepfake algorithm recognition) [14] to validate our work. The training and development sets include 6 types of counterfeit speech generated with different deepfake algorithms and 1 type of genuine speech. The test set includes the 7 classes from the training and development sets, and an unknown counterfeit speech class. The detailed information about the dataset is shown in Table 1.

Table 1
Detailed information of ADD2023 track 3 datasets.

Class idx	training set	dev set	test set
0	3200	1200	-
1	3200	1200	-
2	3197	1200	-
3	3200	1200	-
4	3200	1200	-
5	3200	1200	-
6	3200	1200	-
7	0	0	-
sum	22397	8400	79490

The F1-score is used as the evaluation metric in this work, which is defined as:

$$F_1 = \frac{2 \times P \times R}{P + R}$$

where P and R represent precision and recall, respectively. P and R are defined as:

$$P = \frac{1}{C} \sum_{i=1}^{C} \frac{TP_i}{TP_i + FP_i}$$

$$R = \frac{1}{C} \sum_{i=1}^{C} \frac{TP_i}{TP_i + FN_i}$$

where TP_i, TN_i, FP_i and FN_i denote the true positive, true negative, false positive, and false negative samples of class i.

3.2. Data Augmention

To augment training data, we utilized a common data augmentation method in the speech spoofing detection tasks, which is to add noise and reverberation to the original speech from MUSAN [19] and RIRs [20] datasets. In addition, we add some acoustic scenes as additive noise to improve the robustness of methods under various noisy scenarios. The acoustic scenes are randomly selected from the TAU Urban Acoustic Scenes database [21].

Since ADD2023 Track 3 includes OOD data, it is necessary to mitigate the common issue of overconfidence in deep neural networks. Therefore, we also introduce CutMix [22] as a data augmentation method. The operation of CutMix could be described as

$$\begin{cases} \widetilde{x} = M \odot x_A + (1^{T \times F} - M) \odot x_B \\ \widetilde{y} = \lambda y_A + (1 - \lambda) y_B \end{cases}$$

where x_A and $x_B \in \mathbb{R}^{T \times F}$ are two-dimensional time-frequency feature extracted by utterances randomly selected from the training set. y_A and y_B are the labels of the selected samples. $M \in \{0, 1\}^{T \times F}$ denotes a binary mask indicating where to drop out and fill in from two features, $1^{T \times F}$ is a binary mask filled with ones. $\odot$ is element-wise multiplication. $(\widetilde{x}, \widetilde{y})$ denotes the newly generated training sample. CutMix cuts and pastes two speeches from different classes at the two-dimensional time-frequency feature level, allowing the DNN model to learn a better decision boundary. In addition, CutMix can improve the model's ability to distinguish OOD data [22].

3.3. Model Architecture

Since the method introduced in this work, detecting OOD data based on kNN, is model-agnostic, we attempt to train various model architectures. By doing so, the complementarity between different models could be utilized through fusing model in order to enhance performance.

Similar to the traditional pipeline of speech spoofing detection tasks, in this deepfake algorithm recognition task, we divide the model into a front-end for feature extraction and a back-end for classification. For the front-end, we choose a hand-crafted feature, STFT, and an unsupervised pre-trained feature extractor, Wav2Vec2 [23]. For the back-end, three kinds of model architectures are adopted, which are SENet [4], LCNN-LSTM [24] and TDNN [25]. The SENet is an integration of the ResNet with the squeeze-and-excitation (SE) [26] block. The SENet18 and SENet34 are adopted in our work, the number of blocks of which are different.

The STFT feature is a two-dimensional time-frequency feature, so convolution-based models can learn the patterns that exist in both dimensions. While, although Wav2Vec2 still extracts two-dimensional features from an utterance, the features at each time frame are context representations rather than patterns that could be learned by convolutional kernels. Therefore, SENet-based backends are cascaded to STFT front-ends. And LCNN-LSTM and TDNN, which have RNN-based, which have the ability to extract temporal information, are cascaded to the Wav2Vec2-based front-end.

3.4. Training Strategy

All DNN-based models are trained with Adam optimizer [27], which is adopted with $\beta 1 = 0.9$, $\beta 2 = 0.9$, $\epsilon = 10^{-8}$ and weigth decay 10^{-4}. Angular margin based softmax loss (A-softmax) [28] is adopted as the loss function to be optimized. For the models with STFT-based front-end, the learning rate is initialized as 3×10^{-4}. As a scheduler, StepLR is used with step size of 10 epochs and coefficient 0.5. For the Wav2Vec2-based feature extractor, the learning rate is fixed at 10^{-6}. All models are trained with 100 epochs, in which the model with the the lowest loss on the dev set is selected as the final model.

3.5. Model Fusion

Since the proposed OOD detection method is model-agnostic, to leverage the complementarity between different models, we introduce a logits-based model fusion method. Logits output by different models are weighted and then added. For the samples that are identified as OOD data by kNN-based detector, the original maximum logit value is assigned to the new unknown class, and the logit of the original max class index is set to zero.

4. Result and Analysis

4.1. Results of Data Augmentation

Two data augmentation methods are introduced in this work, namely additive noise and cutmix. Under the same DNN model (STFT+SENet34), the results of data augmentation are shown in Table 2. It should be noted that all results are obtained by directly classifying the test set into 7 classes, without considering the unknown counterfeit class. The results show that the performance of the model has been significantly improved after the addition of additive noise. which is consistent with the traditional speech spoofing detection task. While after adding Cutmix, the performance of the model is not significantly changed.

Table 2
Data augmentation experimental results.

Augment Method	F1-score(%) ↑
no augment	64.89
+additive noise	76.75
+additive noise + CutMix	76.79

4.2. Results of kNN-based OOD detection

Table 3 shows the result of our proposed kNN-based OOD detection method. The experimental results demonstrate that the cosine similarity based kNN distance can effectively distinguish between ID and OOD data. The k-value of the kNN is set to 200. The threshold for distinguishing between ID and OOD data is determined based on the kNN distance between the training data of each class. The kNN-based OOD detection method achieves improvement on five different single models, which demonstrates that it is model-agnostic. which is the basis for utilizing the complementarity between models.

Table 3
F1-scores of the proposed kNN-based OOD detection on single-models.

Single-Model	Original	After Detection
STFT+SENet18	77.93	83.08
STFT+SENet34	77.13	85.78
STFT+SENet34+CUTMix	76.79	84.81
Wav2Vec2+LCNN+LSTM	76.01	84.37
Wav2Vec2+TDNN	75.85	79.09

4.3. Results of Model Fusion

Table 4 shows the result of the proposed logits based model fusion. The five best-performing single-system models are fused together, namely: (1) STFT+SENet18; (2) STFT+SENet34; (3) STFT+SENet34+CutMix; (4) Wav2Vec2+LCNN+LSTM; (5) Wav2Vec2+TDNN. Before model fusion, the results of these models are revised by the proposed kNN-based OOD detector. The fused model achieves an absolute improvement of 3.85% in F1-score compared to our best single-system (STFT+SENet34).

Table 4
F1-score of model fusion Track 3 in ADD2023.

System	Weight	F1-score(%) ↑
STFT+SENet18	0.1	83.08
STFT+SENet34	0.3	85.78
STFT+SENet34+Mixup	0.3	84.81
Wav2Vec2+LCNN+LSTM	0.3	84.37
Wav2Vec2+TDNN	0.1	79.09
Fusion	-	**89.63**

4.4. Results of Submitted System

Table 5 presents the F1-score of the top 5 performing systems in ADD2023 Track 3. Our submitted hybrid system ultimately wins first place in this track.

Table 5
Final results of Track 3 in ADD2023.

ID	F1-score(%) ↑
D01 (our proposed)	**89.63**
D02	83.12
D03	75.41
D03	73.55
D04	73.52

5. Conclusion

This paper describes the system developed for ADD2023 Track3. Five single-models with different front-ends and back-ends are constructed as basic classifiers for the deepfake algorithms recognition task. kNN distance is effective in separating ID samples and OOD samples. Therefore, an OOD detection module based on kNN distance is introduced and improve the performance of single-models significantly. Introducing additive noise during the training process makes single-model more robust. After fusing these models at the logits level, our final system achieves first place in ADD2023 Track3.

Acknowledgments

This work is partially supported by the National Key Research and Development Program of China (No. 2021YFC3320103).

References

[1] V. Shchemelinin, K. Simonchik, Examining vulnerability of voice verification systems to spoofing attacks by means of a tts system, in: International Conference on Speech and Computer, Springer, 2013, pp. 132–137.

[2] T. Kinnunen, Z.-Z. Wu, K. A. Lee, F. Sedlak, E. S. Chng, H. Li, Vulnerability of speaker verification systems against voice conversion spoofing attacks: The case of telephone speech, in: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2012, pp. 4401–4404.

[3] D. A. Reynolds, Speaker identification and verification using gaussian mixture speaker models, Speech communication 17 (1995) 91–108.

[4] Y. Zhang, W. Wang, P. Zhang, The Effect of Silence and Dual-Band Fusion in Anti-Spoofing System, in: Proc. Interspeech 2021, 2021, pp. 4279–4283. doi:10.21437/Interspeech.2021-1281.

[5] M. Todisco, H. Delgado, N. Evans, Constant q cepstral coefficients: A spoofing countermeasure for automatic speaker verification, Computer Speech & Language 45 (2017) 516–535.

[6] H. Tak, J. Patino, A. Nautsch, N. Evans, M. Todisco, Spoofing attack detection using the non-linear fusion of sub-band classifiers, Proc. Interspeech 2020 (2020) 1106–1110.

[7] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, A. Larcher, End-to-end anti-spoofing with rawnet2, in: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2021, pp. 6369–6373.

[8] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, N. Evans, Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 6367–6371.

[9] X. Wang, J. Yamagishi, Investigating Self-Supervised Front Ends for Speech Spoofing Countermeasures, in: Proc. The Speaker and Language Recognition Workshop (Odyssey 2022), 2022, pp. 100–106. doi:10.21437/Odyssey.2022-14.

[10] H. Tak, M. Todisco, X. Wang, J. weon Jung, J. Yamagishi, N. Evans, Automatic Speaker Verification Spoofing and Deepfake Detection Using Wav2vec 2.0 and Data Augmentation, in: Proc. The Speaker and Language Recognition Workshop (Odyssey 2022), 2022, pp. 112–119. doi:10.21437/Odyssey.2022-16.

[11] X. Wang, J. Yamagishi, Investigating active-learning-based training data selection for speech spoofing countermeasure, in: 2022 IEEE Spoken Language Technology Workshop (SLT), IEEE, 2023, pp. 585–592.

[12] N. Müller, F. Diekmann, J. Williams, Attacker Attribution of Audio Deepfakes, in: Proc. Interspeech 2022, 2022, pp. 2788–2792. doi:10.21437/

Interspeech.2022-129.

[13] X. Yan, J. Yi, J. Tao, C. Wang, H. Ma, T. Wang, S. Wang, R. Fu, An initial investigation for detecting vocoder fingerprints of fake audio, in: Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia, DDAM '22, Association for Computing Machinery, New York, NY, USA, 2022, p. 61–68. URL: https://doi.org/10.1145/3552466.3556525. doi:10.1145/3552466.3556525.

[14] J. Yi, J. Tao, R. Fu, X. Yan, T. Wang, Chenglong ang Wang, C. Y. Zhang, X. Zhang, Y. Zhao, Y. Ren, L. Xu, J. Zhou, H. Gu, Z. Wen, S. Liang, Z. Lian, H. Li, Add 2023: the second audio deepfake detection challenge, in: IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis (DADA 2023), 2023.

[15] S. Liang, Y. Li, R. Srikant, Enhancing the reliability of out-of-distribution image detection in neural networks, in: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net, 2018. URL: https://openreview.net/forum?id=H1VGkIxRZ.

[16] A. Bendale, T. E. Boult, Towards open set deep networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 1563–1572.

[17] Y. Sun, Y. Ming, X. Zhu, Y. Li, Out-of-distribution detection with deep nearest neighbors, in: International Conference on Machine Learning, PMLR, 2022, pp. 20827–20840.

[18] Z. Shang, H. Zhang, P. Zhang, L. Wang, T. Li, Analysis and solution to aliasing artifacts in neural waveform generation models, Applied Acoustics 203 (2023) 109183.

[19] D. Snyder, G. Chen, D. Povey, MUSAN: A music, speech, and noise corpus, CoRR abs/1510.08484 (2015). URL: http://arxiv.org/abs/1510.08484. arXiv:1510.08484.

[20] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, S. Khudanpur, A study on data augmentation of reverberant speech for robust speech recognition, in: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2017, pp. 5220–5224.

[21] A. Mesaros, T. Heittola, T. Virtanen, A multi-device dataset for urban acoustic scene classification, in: Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018), 2018, pp. 9–13.

[22] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, Y. Yoo, Cutmix: Regularization strategy to train strong classifiers with localizable features, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 6023–6032.

[23] A. Baevski, Y. Zhou, A. Mohamed, M. Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations, Advances in Neural Information Processing Systems 33 (2020) 12449–12460.

[24] X. Wang, J. Yamagishi, A comparative study on recent neural spoofing countermeasures for synthetic speech detection, in: Proc. Interspeech 2021, 2021, pp. 4259–4263. doi:10.21437/Interspeech.2021-702.

[25] B. Desplanques, J. Thienpondt, K. Demuynck, Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification, Proc. Interspeech 2020 (2020) 3830–3834.

[26] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7132–7141.

[27] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015. URL: http://arxiv.org/abs/1412.6980.

[28] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, L. Song, Sphereface: Deep hypersphere embedding for face recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 212–220.