Subtraction-Positive Similarity Learning

Liang He, Xianhong Chen, Can Xu and Jia Liu
Department of Electronic Engineering, Tsinghua University, Beijing, China
E-mail: heliang@mail.tsinghua.edu.cn, Tel/Fax: +86-10-62781680

***Abstract*—Many methods evaluate the similarity between two vectors x and y by norm or metric learning. They need to get a subtraction vector $x - y$ and then evaluate its length. However, only considering the length of subtraction vector and ignoring its position may lost a lot of information. In this paper, we propose to utilize the position information of subtraction vector to evaluate the similarity. As the subtraction vector between x and y can be expressed either by $x-y$ or by $y-x$, its distribution is centrosymmetric and redundancy. Thus, only half of the subtraction vectors are chosen and named as subtraction positive vectors. The subtraction positive vectors from different classes or from the same class are then modeled by Gaussian mixture models or deep neural network. Experiments were carried out on speaker verification databases including NIST SRE08, SRE10 and NIST i-vector challenge 2014. Results demonstrate the effectiveness of the proposed method.**

I. Introduction

The task of speaker recognition is challenging, because speaker, language and content information are highly correlated and easily influenced by communication channel and background noise. A basic text-independent speaker verification system includes four parts: front-end processing, feature extraction, statistical modeling and score calibration [1].

After front-end processing and feature extraction, an utterance is transformed into a cepstral feature sequence to build statistical model. Gaussian mixture model (GMM) [2], deep neural network (DNN) [3], [4] and their variants are often adopted. Recently, DNN related algorithms become mainstream. Roughly speaking, there are three ways of using DNN.

- Yun Lei has used an ASR DNN acoustic model to extract phonetic Baum-Welch statistics [4]. This method allows comparison of speakers in the same pronunciation unit. Reported experiments on the NIST SRE English database also demonstrate good results. However, this method has a significantly increased computational burden and fails in a multi-language setting.
- Another approach is end-to-end speaker verification [5]. Following this pioneer work, various end-to-end methods are emerging., including attention model [6], ResCNN [7], LSTM[8], GRU [7] and triplet loss and [7], [9]. Compared with previous methods, the end-to-end approach simplifies system design and is very effective for short utterances. Yet, for the long duration text-independent speaker verification task, the traditional i-vector-PLDA is more competive.
- A third approach is the Xvector [10]. This can be seen as a tradeoff between the traditional Ivec-PLDA and the end-to-end method. The DNN is solely designed to extract embeddings and the decision task is left to the backend classifier. Because of its excellent performance in recent NIST SRE 2016, 2018 and its simple implementation using the Kaldi toolkit [11], Xvector has become more and more popular.

After building the statistical model, a similarity function is implemented. There are two dominant approaches:

- Likelihood ratio function. In this case, the estimated statistical model from the training utterance is used as the model to score the test utterance. Examples of this approach include GMM-UBM, latent factor analysis (LFA) [12] and joint factor analysis (JFA) [13], [14].
- Comparison of vectorized parameters. Here, the statistical models are characterized by their estimated parameters, which are often arranged in a vector form. The backend classifier often does not consider the physical meaning of these parameters and only take them as raw input vectors. Because of this, general machine learning methods can be directly applied , including such techniques as principle component analysis (PCA), linear discriminant analysis (LDA) [15] and probabilistic linear discriminant analysis (PLDA).

The task of distinguishing easily confusable speakers is still challenging [16] in this field. Speaker verification errors often occur in the case that the target speaker and mis-identified speaker are similar in phonetic pronunciation, speaking style, and word usage. Motivated by this problem, we propose a subtraction positive similarity learning (SPSL) to boost the system performance.

The remainder of the paper is as follows. Section II formulates the mathematical problem of text-independent speaker recognition. In section III, the subtraction positive similarity learning is proposed and discussed. Experimental work on the NIST SRE08, SRE10 and SRE14 is presented in section IV. Finally, a summary is given in section V.

II. Mathematical problem of text-independent speaker verification

The core task of text-independent speaker verification using cepstral feature sequences can be reformulated as follows. Given the training data

$$O_{\text{train}} = \{\boldsymbol{o}_{\text{train},1}, \boldsymbol{o}_{\text{train},2}, \cdots, \boldsymbol{o}_{\text{train},T_{\text{train}}}\},$$

and the test data

$$O_{\text{test}} = \{\boldsymbol{o}_{\text{test},1}, \boldsymbol{o}_{\text{test},2}, \cdots, \boldsymbol{o}_{\text{test},T_{\text{test}}}\},$$

where O is the acoustic feature sequence and T is the time duration. Our task is to find a proper function $\mathcal{S}(\cdot,\cdot)$ which satisfies

$$\begin{cases} \mathcal{S}(O_{\text{train}}, O_{\text{test}}) > \eta, \\ \quad O_{\text{train}} \text{ and } O_{\text{test}} \text{ are from the same speaker.} \\ \mathcal{S}(O_{\text{train}}, O_{\text{test}}) \leq \eta, \\ \quad O_{\text{train}} \text{ and } O_{\text{test}} \text{ are from different speakers.} \end{cases}$$

where η is a threshold. Since each speaker can be depicted by its statistical model f, we first estimate f and then compare the estimated $\hat{f}$. We consider the property of an ideal function from the perspective of information theory [17]. Let θ denote the parameters of $\hat{f}$, $I(\cdot;\cdot)$ denote the mutual information and $H(\cdot)$ denote the entropy. The mutual information matrix is

$$\begin{bmatrix} I(\theta_{\text{train}};\theta_{\text{train}}) & I(\theta_{\text{train}};\theta_{\text{test}}) \\ I(\theta_{\text{train}};\theta_{\text{test}}) & I(\theta_{\text{test}};\theta_{\text{test}}) \end{bmatrix}.$$

Since $I(\theta_{\text{train}};\theta_{\text{train}}) = H(\theta_{\text{train}}) \geq I(\theta_{\text{train}};\theta_{\text{test}}) \geq 0$ and $I(\theta_{\text{test}};\theta_{\text{test}}) = H(\theta_{\text{test}}) \geq I(\theta_{\text{train}};\theta_{\text{test}}) \geq 0$, the above similarity matrix is a symmetric positive semidefinite matrix. From the Mercer theorem [18], we know that the ideal function can be solved in a high dimensional space called a reproducing kernel Hilbert space (RKHS) [19] and can be expressed as

$$\mathcal{S}(\theta_{\text{train}}, \theta_{\text{test}}) =< \phi(\theta_{\text{train}}), \phi(\theta_{\text{test}}) > \tag{1}$$

where $\phi(\theta)$ is a mapping and $< \cdot, \cdot >$ is an inner product in the RKHS. Prior to considering the realization of $\phi(\cdot)$, we focus on strengthening the desired information and removing nuisance information. This can be accomplished by introducing a symmetric positive semidefinite Q matrix

$$\begin{aligned} \mathcal{S}(\theta_{\text{train}}, \theta_{\text{test}}) &=< \phi(\theta_{\text{train}}), \phi(\theta_{\text{test}}) > \\ &= \phi(\theta_{\text{train}})^t Q \phi(\theta_{\text{test}}) \end{aligned} \tag{2}$$

This equation shows that there are three components (f, ϕ and Q) to be solved for the verification task. Here, f represents statistical modeling which is not the focus of this paper and we will study a kind of ϕ and Q in the next section.

III. Subtraction positive similarity learning

We begin with the classical metric learning approach which looks for a symmetric Q, $Q \succeq 0$ satisfying that the metric distance $\mathcal{L}(\boldsymbol{x}_a, \boldsymbol{x}_b) = (\boldsymbol{x}_a - \boldsymbol{x}_b)^t Q (\boldsymbol{x}_a - \boldsymbol{x}_b)$ is less than a threshold η if $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ are from the same class and is greater than η if $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ are from different classes at the same time. Considering this problem from the view of geometry, we are looking for an ellipse in a multi-dimensional variable space. In the ideal case of metric learning, the same class subtraction vectors $\boldsymbol{z}_{ab} = \boldsymbol{x}_a - \boldsymbol{x}_b$ are mapped inside the ellipse and the different classes subtraction vectors are mapped outside the ellipse, as shown in the upper figure in Fig. 1. Yet, this is not the case in many applications. For the lower figure in Fig. 1, metric learning fails to find a proper oval contour to distinguish circles and triangles. However, the pattern between circles and triangles is clear and can be recognized. Classical metric learning fails because it only concerns quadratic term while ignoring the position of subtraction vectors.

If we study the Fig. 1 carefully, we will find that it is centrosymmetric because $\boldsymbol{z}_{ab} = \boldsymbol{x}_a - \boldsymbol{x}_b$ and $\boldsymbol{z}_{ba} = \boldsymbol{x}_b - \boldsymbol{x}_a$ are opposite vectors. $\boldsymbol{z}_{ab}$ and $\boldsymbol{z}_{ba}$ are redundant and we use a simple rule $\mathbf{1}^t \boldsymbol{z} > 0$ to select $\boldsymbol{z}$, denoted as subtraction positive vectors (SPV) $\boldsymbol{z}_+$. There are two types of SPVs: the same class SPVs (Both subtrahend and minuend are from the same class, SCSPVs) and different classes SPVs (Subtrahend and minuend are from the different classes, DCSPVs). For SCSPVs, we use a GMM $[\kappa_{\mathcal{S}}, \mu_{\mathcal{S}}, \Sigma_{\mathcal{S}}]$ to model them

$$\begin{aligned} \mathcal{L}_{\mathcal{S}}(\boldsymbol{z}_+) =& \frac{1}{|\mathcal{S}|} \sum_{\boldsymbol{z}_+ \in \mathcal{S}} \log \sum_{m=1}^{M} \frac{\kappa_{m,\mathcal{S}}}{\sqrt{2\pi \det(\Sigma_{m,\mathcal{S}})}} \\ & \exp[-(\boldsymbol{z}_+ - \mu_{m,\mathcal{S}})^t \Sigma_{m,\mathcal{S}}^{-1} (\boldsymbol{z}_+ - \mu_{m,\mathcal{S}})] \end{aligned} \tag{3}$$

For DCSPVs, we use another GMM $[\kappa_{\mathcal{D}}, \mu_{\mathcal{D}}, \Sigma_{\mathcal{D}}]$ in a similar way. For an unknown SPV, we compute the log-likelihood $\mathcal{L}(\boldsymbol{z}_+) = \mathcal{L}_{\mathcal{S}}(\boldsymbol{z}_+) - \mathcal{L}_{\mathcal{D}}(\boldsymbol{z}_+)$. If $\mathcal{L}(\boldsymbol{z}_+) > \eta$, the related two utterances are from the same speaker and vice versa.

Usually, the number of DCSPVs is far greater than the number of SCSPVs. We do not need to use all the DCSPVs. This both reduces the amount of computation and puts more attention on easily confusable vectors. The easily confusable vectors are selected by a cosine scoring. For example, $\boldsymbol{x}_a$ is a target vector. We compute all the cosine scores with vectors from different classes, sort them in a deceding order and select vectors corresponding top-N cosine metric scores to compute easily confusable DCSPVs.

We can also use a deep neural network (DNN) instead of a GMM to perform the classification task on the extracted SPVs, as shown in Fig 2. The DNN is a traditional feed-forward network with 3 hidden fully connected layers. From bottom to top, the number of nodes in each hidden layer are 256, 256 and 64 respectively. The output layer is 2 dimension vector with $[1,0]$ for the SCSPVs and $[0,1]$ for the DCSPVs. The activation function of the hidden layers is a sigmoid and the activation function of the output layer is softmax.

The proposed SPSL has no explicit Q and requires a different interpretation approach.

IV. Experiments and Analysis

A. Databases

Experiments were carried out on two sets of data. The first set includes common condition 7 of the SRE08 core task (c7-08) and common condition 5 of the SRE10 core task (c5-10). The core task of the SRE08 is named short2-short3. There are 8 common conditions. The c7-08 task is the telephone-telephone-English condition, containing 1265 models, 1567 test segments and 17761 trials. The core task of the SRE10 is named core-core. There are 9 common conditions. The c5-10 task is telephone-telephone condition, containing 580 models, 678 test segments and 30204 trials.

The second set is the SRE14 ivec (SRE14) challenge database. Differently from previous SREs, i-vectors instead of speech are provided in this challenge. The purpose of NIST is to attract more scholars in the field of machine learning to participate in the challenge. The NIST SRE14 is gender

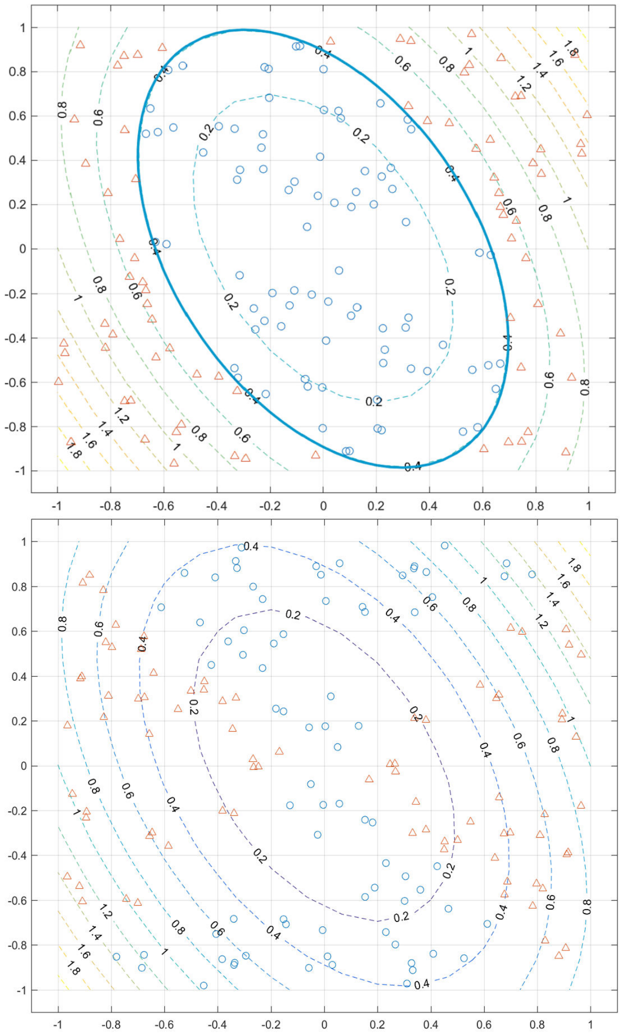

Fig. 1. *Two cases of metric learning. The circle represents subtraction vectors belonging to the same class and the triangle represents subtraction vectors belonging to the different classes. The solid line in the above figure represents the ideal classification ellipse.*

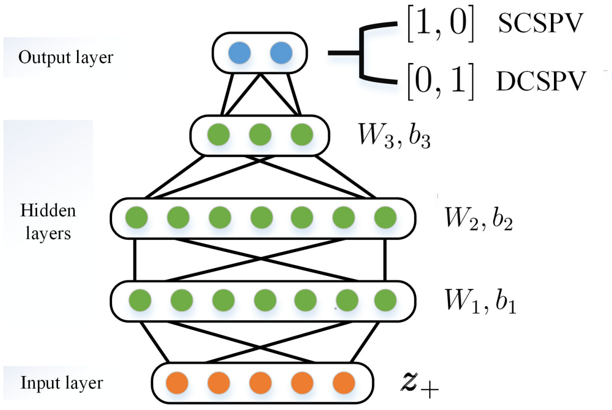

Fig. 2. *DNN for SPV classification.*

independent, contains 1306 models, 9634 test segments and 12582004 trials. The trials are randomly divided into two subsets: a progress subset (40%) and an evaluation subset (60%). Each speaker model has 5 i-vectors and there are 6530 i-vectors for speaker models. In addition, NIST provided a development set, containing 36572 i-vectors. All the i-vectors are 600-dimensional.

B. Configuration

Speech/silence segmentation was performed by a G.723.1 VAD detector. A 13-dimensional MFCC was extracted, with appended delta and acceleration coefficients. 39-dimensional vectors were subjected to feature warping [20]. UBMs with 1024 Gaussian components were gender-dependent. The rank of the matrix T is 800. Length normalization was applied [21].

C. Database 1: SRE08 and SRE10

We used previous NIST evaluation data and some additional corpora to estimate our system parameters. Table I summarizes the data we used. The EER, MDCF08 [1] and MDCF10 [2] were adopted as the performance measurements.

TABLE I
TRAING DATA FOR UBM, T

	SWB	SRE04	SRE05	SRE06	SRE08
UBM	×	×			
T	×	×	×	×	×

[1] The last column are only used for the SRE10 trials.

In SRE08 II, the performance of SPSL-GMM is a little worse than PLDA and LDA-PLDA. But in SRE10 III, it can compete with PLDA and LDA-PLDA.

TABLE II
SUMMARY EXPERIMENTS ON THE C7-08

case	Female		Male	
	EER(%)	MDCF08	EER(%)	MDCF08
Cosine	7.17	0.297	5.69	0.221
PLDA	2.09	0.092	1.93	0.116
LDA-PLDA	1.99	0.090	1.70	0.108
SPSL-GMM	2.12	0.106	2.01	0.125

TABLE III
SUMMARY EXPERIMENTS ON THE C5-10

case	Female		Male	
	EER(%)	MDCF10	EER(%)	MDCF10
Cosine	6.76	0.694	6.70	0.718
PLDA	2.82	0.391	2.55	0.416
LDA-PLDA	2.57	0.435	2.83	0.314
SPSL-GMM	2.71	0.406	2.63	0.332

[1] The minimal detection cost function defined by the NIST SRE08.

[2] The minimal detection cost function defined by the NIST SRE10.

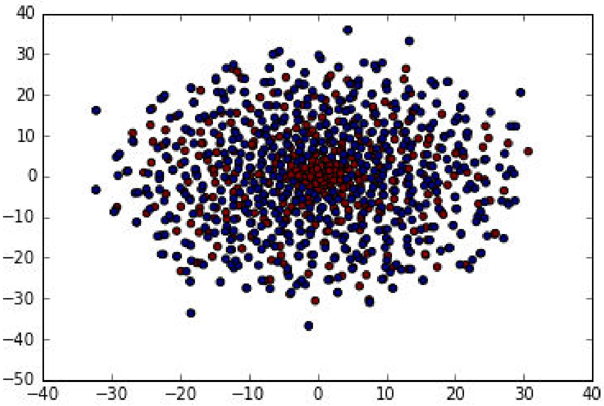

Fig. 3. *tSNE cluster results of SCSPVs (red circles) and DCSPVs (blue circles). The points from SCSPVs show stronger cohesion and the points from DCSPVs show more dispersion. This is the reason that we use more DCSPVs to train GMMs or DNN.*

D. Database 2: SRE14

For the SRE14 experiments, we used the labeled development subset to train our models. The dimension of LDA and PLDA are 250 and 200, respectively. The GMMs with full covariance matrices are trained on both SCSPVs and DCSPVs. If the number of vectors in a class is l, the number of SCSPVs is $l(l-1)/2$. The total number of DCSPVs is much larger than SCSPVs, and only 85000 easily-confused DCSPVs are used, as shown in Fig 3. The configuration of the DNN is depicted in section III.

We also use the PLDA result in Table IV as a baseline system to examine the relative performance improvement of the proposed method. The SPSL-GMM has a relative improvement of 10.32% and 4.03%, which demonstrates the effectiveness of proposed method. The SPSL-GMM in our experiment uses a single GMM mixture. We have tried a higher number of mixtures but it doesn't shown improved performance. The number of confusable DCSPVs is also an experimental number. Higher (120000, 240000) or lower number (70000, 80000) is also examined but we get worse results. As far as we known, the SPSL-GMM is one of few methods which achieves state-of-the-art result without a PLDA backend.

Similarly to [22], we also study the DNN as the backend classifier. The main difference is that the former is based on i-vectors and the latter is based on SPVs. Our experimental results are consistent with [22] in that the DNN has a certain effect.

V. Conclusions

In this paper, we propose a subtraction positive similarity learning (SPSL) for text-independent speaker recognition. SPSL can take good advantage of the position information of subtraction vector. Experiments were carried out on the NIST SRE08, SRE10 and SRE14 data corpus to demonstrate the effectiveness of proposed method.

TABLE IV
System experiments on the SRE14

case	Progress		Evaluation	
	EER(%)	MDCF14	EER(%)	MDCF14
cosine	4.78	0.386	4.46	0.378
PLDA	2.52	0.281	2.39	0.266
LDA-PLDA	2.51	0.259	2.36	0.250
DLPP-PLDA	2.33	0.245	2.17	0.231
SPSL-GMM	2.20	0.271	2.20	0.254
SPSL-DNN	2.95	0.323	2.82	0.307

Acknowledgment

This work was supported by the National Natural Science Foundation of China under Grant No. U1836219.

References

[1] T. Kinnunen and H. Li, "An overview of text-independent speaker verification: From features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12 – 40, 2000.

[2] Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19 – 41, 2000.

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

[4] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Acoustics, Speech, and Signal Processing (ICASSP), 2014IEEE International Conference on*, mar. 2014, pp. 1714 – 1718.

[5] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Acoustics, Speech, and Signal Processing (ICASSP), 2016 IEEE International Conference on*, 2016, pp. 5115 – 5119.

[6] S. Zhang, Z. Chen, Y. Zhao, J. Li, and Y. Gong, "End-to-end attention based text-dependent speaker verification," in *2016 IEEE Workshop on Spoken Language Technology*, 2016, pp. 171–178.

[7] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep speaker: an end-to-end neural speaker embedding system," in *arXiv:1705.02304v1*, 2017.

[8] F A Rezaur Rahman Chowdhury, Quan Wang, Ignacio Lopez Moreno, and Li Wan, "Attention-based models for text-dependent speaker verification," in *arXiv:1710.10470*, 2017.

[9] C. Zhang and K. Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances," in *INTERSPEECH*, 2017, pp. 1487–1492.

[10] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *INTERSPEECH*, 2017, pp. 999–1003.

[11] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. Dec. 2011, IEEE Signal Processing Society, IEEE Catalog No.: CFP11SRW-USB.

[12] R. Vogt, B. Baker, and S. Sridharan, "Modelling session variability in text independent speaker verification," in *INTERSPEECH*, 2005, pp. 3117–3120.

[13] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of interspeaker variability in speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 5, pp. 980–988, 2008.

[14] P. Kenny, G. Boulianne, and P. Dumouchel, "Eigenvoice modeling with sparse training data," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 13, no. 3, pp. 345–354, 2005.

[15] A.M. Martinez and A.C. Kak, "Pca versus lda," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 2, pp. 228 –233, feb 2001.

[16] Liang He and Jia Liu, "Prism: A statistical modeling framework for text-independent speaker verification," in *IEEE China Summit and International Conference on Signal and Information Processing*, 2015, pp. 529–533.

[17] T. M. Cover and J. A. Thomas, *Elements of information theory*, New York: Wiley, 2006.

[18] J. C. Ferreira and V. A. Menegatto, "Eigenvalues of integral operators defined by smooth positive definite kernels," *Integral Equations and Operator Theory*, vol. 64, pp. 61–81, 2009.

[19] S.K. Zhou and R. Chellappa, "From sample similarity to ensemble similarity: probabilistic distance measures in reproducing kernel hilbert space," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 6, pp. 917–929, June 2006.

[20] B. Xiang, U. V. Chaudhari, J. Navratil, G. N. Ramaswamy, and R. A. Gopinath, "Short-time gaussianization for robust speaker verification," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, 2002, pp. 681–684.

[21] D. G. Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker verification systems," in *INTERSPEECH*, 2011, pp. 249–252.

[22] Javier Hernando, Javier Hernando, Javier Hernando, and Javier Hernando, "Deep learning backend for single and multisession i-vector speaker recognition," *IEEE/ACM Transactions on Audio Speech , Language Processing*, vol. 25, no. 4, pp. 807–817, 2017.