KU LEUVEN

ARENBERG DOCTORAL SCHOOL
Faculty of Engineering Technology

Deep learning for sound source localisation and speech emotion recognition

A perspective on representation learning and sequence modelling

Duowei Tang

Supervisor:
Prof. dr. ir. Toon van Waterschoot
Co-supervisors:
Prof. dr. ir. Luc Geurts
Prof. dr. Peter Kuppens

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Technology (PhD)

October 2022

Deep learning for sound source localisation and speech emotion recognition

A perspective on representation learning and sequence modelling

Duowei TANG

Examination committee:
Prof. dr. Greet Langie, chair
Prof. dr. ir. Toon van Waterschoot, supervisor
Prof. dr. ir. Luc Geurts, co-supervisor
Prof. dr. Peter Kuppens, co-supervisor
Prof. dr. ir. Hugo Van hamme
Prof. dr. ir. Bart Vanrumste
Dr. Merijn Mestdagh
Dr. Vincent Spruyt
(Reprise Digital)
Prof. dr. ir. Dorothea Kolossa
(Ruhr-Universität Bochum, Germany)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Technology (PhD)

October 2022

Uitgegeven in eigen beheer, Duowei Tang, Kasteelpark Arenberg 10, bus 2446, B-3001 Leuven (Belgium)

Preface

择高处立，就平处坐，向宽处行。[1] [2] ——左宗棠

My father often tells me this phrase. The pursuit of a PhD is a climb towards the summit of my life, a process that is a mixture of flavours; it is challenging, yet bland and enduring; it is aspirational, yet fragile and unbearable. It has been an intertwining of personal struggle, life lessons, sharing friendships, experiencing love and more, and it has allowed me to grow and climb higher.

However, the height of the climb itself may not be the most important thing, but more importantly what I experience in the process of climbing. When I stand on the heights, everything returns to the ordinary, and all I want to do is to find the next high place. I would like to thank from the bottom of my heart everyone who has helped me, inspired me and forgiven me during this time.

First of all, I would like to thank my supervisor Toon, who infected me with his rigour, his approach to life, his pursuit of art and some cold humour. He was not only inspiring in his research, but also enormously helped me to correct my papers and thesis again and again, and always supported me, even when I made mistakes that seem silly.

[1] Literally translation: Choose the high ground, sit on the level ground, and walk towards the wide ground. – Zuo Zongtang

[2] Interpretation: If you stand high, you can be far-sighted; if you lower your posture, you can eliminate disasters and distress; if you adjust your direction at the right time, you can travel unhindered.

I would also like to thank my co-supervisors, Luc and Peter, for Luc's recognition and trust in me at the beginning to undertake the m-Sense project, which led me to start my PhD research. I would like to thank Peter for his enormous help, encouragement and inspiration, and for doing everything he could to help me without letting me worry.

I am grateful to the members of the supervisory and examination committee for carefully reading my dissertation and for the interesting discussions. Thank Hugo, Bart, Merijn, Vincent, Dorothea, as well as the chair Greet.

Thanks and greetings also go to former and current colleagues and friends who have encouraged and supported me, thanks Ahmed, Aldona, Alam, Ally, Amin, Arina, Ann, Benjamin, Bruno, Bram, Bilgesu, Celien, Chetanya, Elena, Elisa, Elise, Enzo, Esmael, Evelien, Fanghui, Feiteng, Filip, Gabriel, Georgios, Gertjan, Giacomo, Giuliano, Hannes, Hannelore, Heereen, Huizi, Ine, Jesper, Jeroen, Jelimo, Jenny, Jonas, Julian, Jujian, Julia, Jingwei, Kathleen, Karsten, Kenny, Kristoff, Kymeng, Lucas, Lichen, Maja, Mathias, Martin, Mina, Min, Mohit, Naveen, Niccolò, Paul, Peng, Pourya, Qi, Qinghua, Randy, Richard, Robbe, Rob, Santiago, Stefano, Shu, Sung-kyo, Siya, Site, Taewoogn, Tao, Teng, Thomas, Wouter B., Wouter T., Xueru, Xueying, Yanyue, Yangyi, Ying, Yingyi, Yiyuan, Xue, Xuemeng, Xingyue, Xin, Zixuan, Zihan, Zhenxiang, Zhaohong, and all those that I keep deeply in my heart.

Finally, I would like to thank my family. I would like to thank my parents for their guidance and support in the trajectory of my life, allowing me to pursue my goals without worry and also giving me a shelter when I encountered difficulties. I am grateful to my brother Duolang, who has a great personality and we exchanged many ideas during my PhD. I am even more grateful to my girlfriend Melissa who at her wonderful age chooses to take care of me, bear me, give me understanding, and support me mentally during this most stressful period. Thanks to my girlfriend's father Gino for always helping us unconditionally to build our horse farm, to my girlfriend's mother Agna for always quietly helping us cleaning our house, and to my girlfriend's grandmother Moeke for always being an elder to guide the trajectory of my life and her delicious bunny stew. Thank my girlfriend's brother Sammy for the inspiring talks and his many helps with the maintenance of my car, and thank his girlfriend

Indiana for her exquisitely prepared gifts in my birthdays. [3]

I value every person I meet in my life, and my life is spectacular because of all of you.

September, 2022

Duowei Tang

[3] 译文：最后，我要感谢我的家人。我要感谢我的父母，感谢他们对我人生的指引和支持，让我可以无忧无虑地追求我的人生目标，并且在我遇到困难时给予我一个避风的港湾。我要感谢我的弟弟铎朗，他有着美好的品质和性格，也在我博士期间与我交流新鲜的想法。我更要感谢我的女朋友Melissa，是她在她最青春年少的年纪和我压力最大的这段时间选择照顾我，忍耐我，给予我理解，并在精神上支持我。我要感谢我女朋友的父亲Gino，是他总是无条件地帮助我们修缮我们的马场，我女朋友的母亲Agna，她总是默默地帮助我们打扫我们的房子，还有我女朋友的外祖母Moeke，是她作为长者指引着我人生的轨迹，还有她美味的炖兔肉。感谢我女朋友的哥哥Sammy，与我无数次迸发灵感的对话，并且帮助我保养我的汽车，并且感谢他的女朋友Indiana在我生日时精心准备的礼物。

Abstract

Speech contains a large amount of useful information, as not only it constitutes one of the main mechanisms of human-to-human communication, but it also provides one of the indispensable modalities in human-computer interaction. In order to accommodate processing and information retrieval from speech, computational speech processing systems convert speech sound waves into one-dimensional discrete time series, i.e., the digital speech recording. However, the quality of these speech recordings is hampered by various undesirable artefacts such as reverberation, background noise, distortions due to the non-ideal response and limited numerical precision of the recording device, etc. Therefore, an effective speech information retrieval system needs to simultaneously identify and interpret meaningful temporal content in the speech recording while also resisting against interference of artefacts and irrelevant components.

Present-day deep-neural-network-based data-driven models have surpassed the human average performance in a variety of perceptual tasks, and provide powerful and applicable tools for modern speech/audio processing, including speech information retrieval. In this thesis, we propose to use deep neural network models to first retrieve features that capture high-level speech representations reflecting the intrinsic structure of the data, and then explore the temporal relationships among these features through a sequence model. We apply this modelling paradigm to two speech/audio processing tasks, namely binaural sound source localisation and speech emotion recognition.

For these two tasks, binaural sound source localisation and cross-language/cross-corpus speech emotion recognition, we design distinct models to learn representations which reflect the intrinsic structure of the acquired data that is relevant to the envisaged task. In the binaural sound source localisation task, we propose a parametric embedding by defining a similarity metric in a latent space using a deep neural network architecture known as the "siamese" network. This model can be optimised to map points that are close to each other in the latent space (the space of source azimuths and elevations) to nearby points in the embedding space, thus the Euclidean distances between the embeddings reflect their source proximities, and the structure of the embeddings forms a manifold, which provides interpretability to the embeddings. We show that the proposed embedding generalises well in various acoustic conditions (with reverberation) different from those encountered during training, and provides better performance than unsupervised embeddings previously used for binaural sound source localisation. We also extend this embedding to use both supervised learning and weakly supervised learning, and show that in both conditions, the resulting embeddings perform similarly well, whereas the weakly supervised embedding allows to estimate source azimuth and elevation simultaneously.

In the cross-language speech emotion recognition task, we aim to mitigate the model performance degradation problem in cross-language and cross-corpus conditions, and propose a transfer learning method that uses a pre-trained wav2vec 2.0 model. This model can transfer the time-domain audio waveforms into a shared embedding space across different languages (i.e. 53 different languages), and it is trained in a way that contextual information is kept thus marginalising out the influence of language variability. Then, we propose a Deep-Within Class Covariance Normalisation (Deep-WCCN) layer that can be inserted into the artificial neural network model for further reducing its susceptibility to other variabilities such as speaker variability and channel variability. Experimental results show that the proposed method outperforms a baseline method that is based on common acoustic feature sets for speech emotion recognition in the within-language setting, as well as the baseline model and the state-of-the-art models for the cross-language setting. In addition, we experimentally validate the effectiveness of the Deep-WCCN, which can further improve the model performance. Finally, we show that the proposed transfer learning method exhibits good data efficiency when merging target language data into the fine-tuning process.

We also address the problem of modelling the temporal dependencies in long speech/audio sequences (especially for end-to-end learning), and propose a novel end-to-end learning deep neural network model for speech emotion recognition. This model is based on the concept of dilated causal convolution with context stacking, is parallelisable and has a receptive field as large as the input sequence length while maintaining a reasonably low computational cost. We evaluate the proposed model in speech emotion recognition regression and classification tasks, and show that it improves the recognition performance over the state-of-the-art end-to-end model. Moreover, we also study the impact of using various input representations such as the raw audio samples versus log mel-spectrograms and illustrate the benefits of an end-to-end approach over the use of hand-crafted audio features.

Beknopte samenvatting

Spraak bevat een grote hoeveelheid nuttige informatie; het vormt niet alleen een van de belangrijkste mechanismen voor communicatie tussen mensen, maar het levert ook een van de onmisbare modaliteiten in de mens-computerinteractie. Om het mogelijk te maken nuttige informatie uit spraak te verwerken en extraheren, zetten computationele spraakverwerkingssystemen spraakgeluidsgolven om in eendimensionale discrete-tijdreeksen, d.i. de digitale spraakopname. De kwaliteit van deze spraakopnamen wordt echter aangetast door verschillende ongewenste artefacten, zoals nagalm, achtergrondruis, vervormingen ten gevolge van de niet-lineaire respons en de beperkte numerieke precisie van het opnameapparaat, enz. Daarom moet een effectief systeem voor het analyseren van spraakinformatie niet alleen de relevante temporele inhoud in de spraakopname identificeren maar tegelijk ook de interferentie van artefacten en irrelevante componenten beperken.

Hedendaagse diepe-neurale-netwerk-gebaseerde data-gedreven modellen hebben de menselijke gemiddelde prestatie in een verscheidenheid van perceptuele taken overtroffen, en bieden krachtige en toepasbare hulpmiddelen voor moderne spraak/audio-verwerking, waaronder het analyseren van spraakinformatie. In dit proefschrift stellen we voor om diepe neurale-netwerkmodellen te gebruiken om eerst kenmerken te bekomen die spraakrepresentaties op hoog niveau vastleggen en zodoende de intrinsieke structuur van de data kunnen weergeven, en vervolgens de temporele relaties tussen deze kenmerken te onderzoeken via een sequentiemodel. We passen dit modelleringsprincipe toe op twee spraak/audio processing gerelateerde taken, namelijk binaurale geluidsbronlokalisatie en spraakemotieherkenning.

Voor deze twee taken, binaurale lokalisatie van geluidsbronnen en herkenning van spraakemoties over meerdere talen en corpora, ontwerpen we specifieke modellen om representaties te leren die de intrinsieke structuur van de verzamelde data weerspiegelen die relevant is voor de beoogde taak. Voor de binaurale lokalisatie van geluidsbronnen stellen we een parametrische inbedding voor door een gelijksoortigheidsmetriek te definieëren in een latente ruimte met behulp van een diepe neurale-netwerkarchitectuur die gekend is als het “siamese” netwerk. Dit model kan worden geoptimaliseerd om punten die dicht bij elkaar liggen in de latente ruimte (de ruimte van azimut- en elevatiecoördinaten van de bron) af te beelden op nabijgelegen punten in de inbeddingsruimte, zodat de Euclidische afstanden tussen de inbeddingen de afstanden in de coördinatenruimte van de bron weerspiegelen. De structuur van de inbeddingen vormt bovendien een variëteit, die interpreteerbaarheid biedt aan de inbeddingen. We tonen aan dat de voorgestelde inbedding goed veralgemeenbaar is in verschillende akoestische omstandigheden (met nagalm) die verschillen van de omstandigheden tijdens de training, en betere prestaties levert dan niet gesuperviseerde inbeddingen die eerder zijn gebruikt voor binaurale geluidslokalisatie. We breiden deze inbedding ook uit om zowel gesuperviseerd leren als zwak gesuperviseerd leren te gebruiken, en tonen aan dat in beide omstandigheden de resulterende inbeddingen even goed presteren, terwijl de zwak gesuperviseerde inbedding het mogelijk maakt om de azimut- en elevatiecoördinaten van de bron tegelijkertijd te schatten.

In onze aanpak van de spraakemotieherkenningstaak willen we de prestatievermindering van het model in scenario’s met meerdere talen en corpora tegengaan, en stellen we een transfer-leermethode voor die gebruik maakt van een voorgetraind wav2vec 2.0 model. Dit model kan de tijdsdomein audiogolfvormen overbrengen naar een inbeddingsruimte die gedeeld wordt over verschillende talen (d.w.z. 53 verschillende talen), en het is getraind op een manier dat de contextuele informatie wordt behouden waardoor de invloed van de taalvariabiliteit wordt gemarginaliseerd. Vervolgens stellen we een Deep-Within Class Covariance Normalisation (Deep-WCCN) laag voor die kan worden ingevoegd in het artificiële-neurale-netwerkmodel om de gevoeligheid van het model aan andere variabiliteiten, zoals sprekervariabiliteit en kanaalvariabiliteit, verder te reduceren. Experimentele resultaten

tonen aan dat de voorgestelde methode beter presteert dan de referentiemethode die gebaseerd is op gemeenschappelijke akoestische kenmerkverzamelingen voor spraak-emotieherkenning met een enkele taal, evenals het referentiemodel en de state-of-the-art-modellen voor de setting met meerdere talen. Bovendien valideren we experimenteel de effectiviteit van de Deep- WCCN, die de prestaties van het model verder kan verbeteren. Tenslotte tonen we aan dat de voorgestelde transferleermethode een goede data-efficieëntie vertoont bij het invoegen van data met de beoogte taal in het fine-tuningproces.

We behandelen ook het probleem om de temporele afhankelijkheden in lange spraak/audio-sequenties te modelleren (in het bijzonder voor end-to-end leren), en stellen een nieuw end-to-end diep neuraal-netwerkmodel voor spraakemotieherkenning voor. Dit model, gebaseerd op het concept van gedilateerde causale convolutie met context stacking, is paralleliseerbaar en heeft een receptief veld zo groot als de lengte van de inputsequentie, terwijl de computationele kost redelijk laag blijft. We evalueren het voorgestelde model in regressie- en classificatietaken voor spraakemotieherkenning, en tonen aan dat het de herkenningsprestaties verbetert t.o.v. het state-of-the-art end-to-end model. Bovendien bestuderen we ook de impact van het gebruik van verschillende inputrepresentaties zoals de ruwe audiobemonsteringen versus log melspectrogrammen en illustreren we de voordelen van een end-to-end aanpak ten opzichte van het gebruik van specifiek ontworpen audiokenmerken.

List of Abbreviations

ADC Analog-to-digital converter.

ANN Artificial Neural Network.

ASR Automatic Speech Recognition.

AVEC Audio/Visual Emotion Challenge and Workshop.

CCC Concordance Correlation Coefficient.

CNN Convolutional Neural Network.

DCT Discrete Cosine Transform.

DNN Deep Neural Network.

e.g. *exempli gratia*, for example.

ECG Electrocardiogram.

EDA Electro-dermal activity.

eGeMAPS Extended Geneva Minimalistic Acoustic Parameter Set.

Emo-DB Berlin Database of Emotional Speech.

ESD Emotional Speech Dataset.

FIR Finite Impulse Response.

GBDT Gradient Boosted Decision Trees.

GeMAPS Geneva Minimalistic Acoustic Parameter Set.

GMM Gaussian Mixture Model.

GRU Gated Recurrent Unit.

HCI Human Computer Interaction.

HMM Hidden Markov Model.

HRTF Head-related Transfer Function.

i.e. *id est*, that is.

i.i.d independent and identically distributed.

IEMOCAP Interactive Emotional Dyadic Motion Capture.

ILD Interaural Level Difference.

IPD Interaural Phase Difference.

LEM Laplacian Eigenmaps.

LPC Linear Predictive Coding.

LSTM Long Short-term Memory.

MAE Mean Absolute Error.

MFCC Mel-Frequency Cepstral Coefficients.

MLE Maximum likelihood estimation.

MSE Mean Squared Error.

NLP Natural Language Processing.

NN Neural Network.

PCA Principal Component Analysis.

PCC Pearson Correlation Coefficient.

RAVDESS Ryerson Audio-Visual Database of Emotional Speech and Song.

RECOLA REmote COLlaborative and Affective.

ReLU Rectified Linear Unit.

RF Random Forest.

RNN Recurrent Neural Network.

SCE Supervised Contrastive Embedding.

SER Speech Emotion Recognition.

SGD Stochastic Gradient Descent.

SNR Signal to Noise Ratio.

SSL Sound Source Localisation.

STFT Short-time Fourier transform.

SVM Support Vector Machine.

TDNN Time-Delay Neural Network.

TTS Text-to-speech.

UA Unweighted Accuracy.

VAD Voice Activity Detection.

VR Virtual Reality.

WA Weighted Accuracy.

WCCN Within-Class Covariance Normalisation.

WSCE Weakly supervised Contrastive Embedding.

List of Symbols

∞	infinity
$\mathbb{E}$	expectation
$\mathbb{R}$	the set of real numbers
$\mathcal{I}\{a\}$	the imaginary part of a
$\mathcal{R}\{a\}$	the real part of a
∂	partial derivative
$\sum$	summation
$^\circ$	degree
bits/s	bits per second
dB	decibel
kHz	killo Hertz
m	meter
ms	millisecond
s	second

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Speech is one of the most basic forms of human communication through the syntactic and phonetic vocalisation of language. Complex, diverse, syntactic language as the basic content of speech is an important factor that has contributed to the evolution of human intelligence and distinguishes humans from other animals [40]. Speech has more ways of expression than written language, such as different ways of pronouncing phonemes, varying intonations, high and low loudness, and paralinguistic elements that can describe the entire universe concretely, briefly, abstractly, vividly, emotionally and bluntly as well as expressing personal emotions, feelings, thinkings, memories, revealing one's gender, age, physical conditions, origins, etc.

Speech encodes a vast amount of information from one's brain into propagating sound waves that will be received by other listeners. From the information theory point of view, even in the narrowband case (below 7000 Hz), with additive Gaussian noise at a Signal to Noise Ratio (SNR) of 24 dB, human speech has a capacity of 3680 bits/s [54]. Some bits can be used to encode a word in a certain language containing, for example, a yes/no answer, a country name, the weather condition, whereas other bits to determine the speaker's more stationary characteristics, such as emotion state (i.e. happy, sad, angry, neutral), health condition, gender, identity, etc. At the receiver's side, the listener decodes the information using his/her own "codebook" (i.e. the definitions of the words and articulations). This natural innovation, evolving over millions of years, has created the most basic channels of communication from human to human.

In this information age, humans are not satisfied with human-to-human communication only, and in turn pursue interaction with digital devices and create human-like robots or virtual humans that exist in a virtual metaverse [46]. In addition to being a basic and expressive human-to-human communication channel, speech also has the advantage of being more accessible than images/video and physiological signals (e.g., recording speech does not require a camera oriented towards the face, nor a wearable biosensor that has to be mounted closely to the user), which makes it suitable as a complementary and essential modality in human-machine interfaces.

The demand for computational speech processing has given rise to many speech processing research areas, which subsequently make up the wide range of applications (e.g., voice assistant, conversational robot, speech dictation, etc.) that facilitate our modern lives. Those successful applications benefit from the fast developing technologies in this digital world. Physically, an exponentially increasing trend of the number of transistors in an integrated circuit is still observed (i.e. the Moore's law). Algorithmically, mathematical models tend to contain more and more parameters and to deal with massive amounts of data [28, 23, 171]. In Automatic Speech Recognition (ASR) systems, the first speech recogniser (called AUDREY) was created almost 70 years ago [117]. This giant machine was built with vacuum tubes and only performed well for a designated speaker in single-digit recognition. In contrast, nowadays speech recognition systems utilise Deep Neural Networks (DNNs), which contain millions of parameters and have been trained on thousands of hours or even half-million-hours of speech data [38, 10, 12], can be accessed in a small smartphone and can perform robustly well towards any speaker and in varying noisy environments [7].

In this thesis, we address two problems related to speech/audio information retrieval: *binaural Sound Source Localisation (SSL)* and *Speech Emotion Recognition (SER)*. By solving these problems in a deep learning framework, we develop general perspectives of learning robust representations and modelling long temporal sequences.

Before introducing the SSL and SER problems in Sec 1.3 and 1.4 and elaborating on the challenges and contributions and outline of this thesis in Sec 1.5, 1.6 and 1.7, we will first provide a concise introduction to digital speech and audio processing in Sec 1.1 and 1.2.

1.1 Speech tasks

In computational speech/audio processing, sound waves are picked by microphones that transform the oscillation of the propagation medium into analog electrical signals, which can be digitalised to be further processed. In the digitalisation process, which is illustrated in Figure 1.1, if we only consider one microphone, the Analog-to-digital converter (ADC) samples the analog microphone signal at a constant interval called *sampling period*, and its reciprocal is the *sampling frequency*. Conventionally, a sampling frequency of maximumly 48 kHz is used for speech recordings according to the Nyquist criterion. Then, due to the limited precision of the floating-point number representations in digital devices, the samples are rounded to the closest number that the device can represent, which in turn causes a quantisation error. The resulting 1-dimensional sequence is referred to as the *discrete-time waveform* of the speech recording. From a causality point of view, the observed waveform depends on many factors (i.e. the sources of influence). A non-exhausting list of deterministic factors (i.e. factors that are considered stationary for a relatively long period or potentially known) is as follows:

1. The vocal cord shape and condition determines the pitch of the voice which results in a woman's voice usually sounding "higher" than a man's voice. The vocal tract shape and condition, on the other hand, determine the unique timbre of a person's voice.

2. The interior arrangement of the recording room determines its acoustic characteristic of a room, since the propagating sound wave can be reflected, diffracted, and absorbed by the furniture, the wall, the ceiling and floor, etc., and resulting a persistent sound that is added to the original sound (known as *reverberation*).

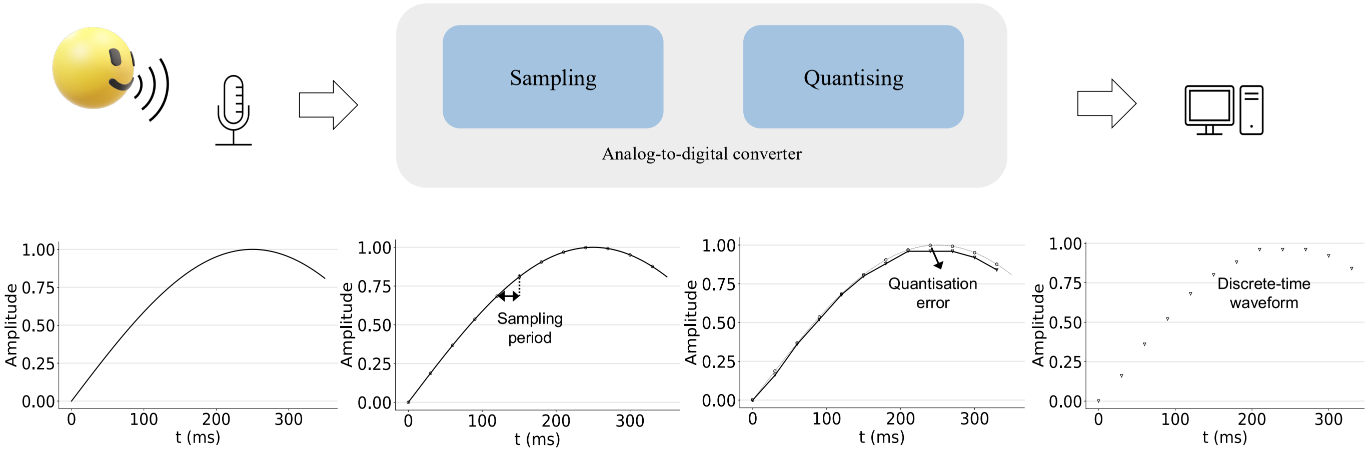


Figure 1.1: The illustration of speech signal digitalisation.

3. The speech content (e.g., the arrangement of the spoken words and the articulations) results different waveforms.
4. The location and orientation of the speaker and the recording device, especially in the presence of objects around the speaker or microphone may contribute to changes in the observed waveform.

Next to this, there are also non-deterministic factors (e.g., factors changing fast through time or not being consistent) that influence the observed speech waveforms, such as

1. The speaker's health condition (e.g., heavy breath, blocked nose, swollen throat, depressed mind, etc.) influences the speech in terms of loudness, speed, timbre, style and so on.
2. The speaker's identity may give rise to variations to pronunciation, accent, etc. Although the phonemes in one language (i.e. the basic unit speech of spoken language is built) sound similar (i.e. have similar frequency spectra), they are varying across speakers and conditions.
3. The electroacoustic characteristics of the recording device also lead to some differences between the acoustic waveform and the digitalised electric waveform. An example is that each microphone has its frequency-dependent polar pattern (i.e. the frequency respond towards sound coming from different angles), which is not only determined by the manufacturing process, but also by temperature, electrical noise, pressure, etc. This pattern variation causes the digitalised signals to have non-identical waveforms even when recordings are made with the same or similar microphones.
4. The presence of background noise may interfere with the original sound thus deteriorating or masking some components of the original sound. Unfortunately, real-life background noise is commonly non-deterministic and non-stationary.
5. The emotion of the speaker also results in variations in speech pronunciation, pitch, timbre, speed, etc.

The aforementioned factors are ingredients that are mixed, combined, and interwoven into the 1-dimensional temporal sequence that we can observe. However, most real-life speech applications require an inverse problem to be solved, i.e. extracting and separating these factors from the observed temporal sequence. Here, we briefly introduce 5 major directions in speech processing.

1. **Speech production modelling:** is done through mathematical modelling of the physical processes by which humans turn thoughts into speech. One of the most basic models is the source-filter speech model, which divides the process of speech production into two parts, the first being an excitation generator (corresponding to the vocal cord) and the second being a time-varying linear system (corresponding to the vocal tract) [119]. A typical choice for the excitation generator is an impulse train (which mimics the vocal cord vibration when pronouncing voiced speech) or white noise (which mimics the process when pronouncing non-voiced speech). The linear system is typically represented by an all-pole filter. For voiced sounds, each sound will be associated with a different pole distribution and pole bandwidth. These poles represent the resonant frequencies also known as *formants*.

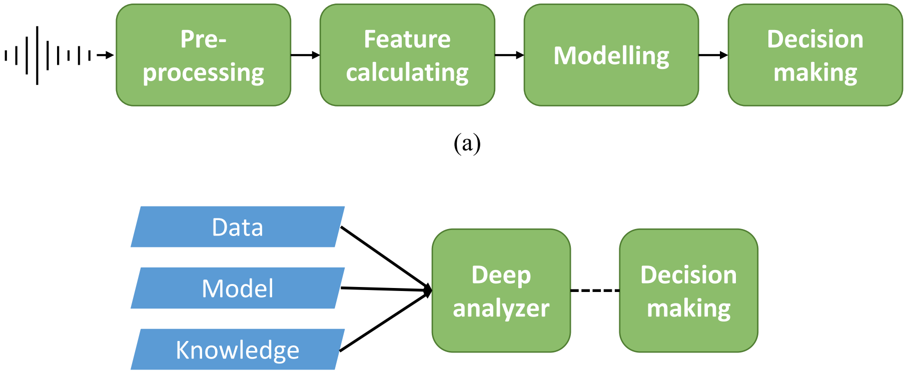
2. **Speech representation learning:** is achieved by representing a segment of speech with a low-dimensional feature vector that ideally needs to be high-fidelity, disentangled, robust to noise and reverberation, etc. A speech representation based on the source-filter model is called Linear Predictive Coding (LPC), which obtains an all-zero filter configuration (i.e. a Finite Impulse Response (FIR) filter) by finding the inverse filter of the underlying all-pole filter. LPC stores only the source-filter model parameters, thus significantly reducing the storage space required for a segment of speech (e.g., a raw audio waveform, sampled at 16 kHz and quantised at 8 bits/sample would have a bitrate equal to about 128,000 bits/s, whereas, LPC requires a bitrate equals to 7800 bits/s) [5]. However, LPC is only designed to retain formant information, thus losing a great deal of time and frequency domain details. Another widely used speech representation are the Mel-Frequency Cepstral Coefficients (MFCC), which preserve more of the most

important speech information by considering a cosine transform of the mel-frequency spectrum. More advanced speech representations through DNN and self-supervised learning [12, 38, 79, 10] will be explained in detail in later chapters.

3. **Speech information retrieval:** is the extraction of contextual semantic information from speech, such as speaker identity recognition, speech recognition, speech emotion recognition, sound source localisation, gender recognition, primary screening for respiratory diseases (e.g., for Covid-19 screening [67]), etc. As it is hard to have an accurate physical model to describe the relationship between this semantic information and the speech signal, these speech information retrieval systems are usually based on data-driven models, and therefore it is particularly important to achieve generalisability of the models to unseen data.

4. **Language modelling:** is the modelling of the linguistic information embedded in speech. In general, the language model and the speech acoustic model are separate, i.e. the speech model first processes a phonological representation to identify its corresponding phoneme sequence, then the language model combines and arranges the phonemes into words according to the content relations of the phoneme sequence (e.g., the probability distribution of the phoneme sequence), and then forms utterances through the contextual relations between words (e.g., the probability distribution of the word sequence). Thus, the language model can learn grammatical information, synonym information and other high-level features, such as in the recent DNN-based language models [28, 110].

5. **Speech synthesis:** different from the other tasks above, synthesis is the process of converting a text into speech, so that speech is no longer used as an input but as an output in this task. The simplest speech synthesis systems take pre-recorded phonemes or diphones (i.e. sound-to-sound transitions) and recombine them to produce speech for different words and sentences. Today, DNN based synthesisers can transform text into high-quality, natural, human-like speech from end-to-end [142, 157, 139].



(a)

(b)

Figure 1.2: Speech/audio processing conventional (a) versus modern (b) system architecture.

1.2 Speech/Audio processing 2.0

The above brief overview of speech processing tasks shows that the trend in computational speech processing is moving towards deep learning, which is based on DNN models. Taking a speech information retrieval system as an example, a conventional computational speech processing pipeline is shown in Figure 1.2 (a) (adapted from [138]), in which a speech recording needs to go through a few steps as follows,

- **Step 1:** *Pre-processing*, such as signal normalisation, pre-emphasis, noise reduction, reverberation reduction, echo cancellation, Voice Activity Detection (VAD), etc.

- **Step 2:** *Feature calculation*, where acoustic and speech features such as signal-level features (e.g., loudness, zero-crossing rate), frequency-related features (e.g., pitch, formant), and audio features (e.g., MFCC, LPC, i-vector features) are extracted. In some systems, some post-processing operations are applied to the features, such as feature selection and vector quantisation.

- **Step 3:** *Modelling*, where mathematical models are designed for specific tasks. However, there are practical problems in the conventional modelling, for example it may be difficult to apply the model to big data because many calculations use large matrices and in many cases the size of these matrices depends on the amount of data. In addition, most conventional signal processing models need to be redesigned to suit different input features or to add supplementary information, e.g., additional reference microphone signals for noise reduction, or varying forms of constraints such as speech content for speech recognition applications, and gender information for speaker recognition applications.

- **Step 4:** *Decision making*, for classification or regression problems. With conventional signal processing models, the decision making does not allow for much flexibility. Firstly, the decision output is generally a continuous number, which for classification problems requires further processing, e.g., thresholding. Secondly, the type of objective function is limited by meeting the convex property.

A new wave of thinking has recently emerged and laid the foundation for the speech/audio processing 2.0 era (illustrated in Figure 1.2 (b)), in which a deep analyzer can take charge of the entire conventional speech/audio processing chain (from pre-processing to decision making). The deep analyzer learns audio features, predictive models and even decision making in a purely data-driven manner, thus significantly reducing the labour cost and professional threshold for feature design, model design, etc. In addition, such deep analyzer models allow for end-to-end learning and inference. Not only does this allow the model to learn appropriate and effective features from lossless raw data or shallow features to improve the system's performance, but it also allows the use of the same model, or parts of the model, for similar tasks (e.g., speech recognition and speech emotion recognition), thus eliminating the need to redesign an entirely new model for each task, define new features, or learn from scratch for each dataset. A good example is transfer learning where multiple tasks can share the same data-driven learned features, and by fine-tuning the transfer learning models to each task, it can greatly increase the generalisation capability of the model [38, 10, 141, 91, 172].

A good candidate for the deep analyzer is the Artificial Neural Network (ANN), which has the following advantages for this new era of speech/audio signal processing 2.0. The list is non-exhaustive,

- It consists of a simple operation which is based on a linear transformation plus a non-linear activation function, does not require complex mathematical transformations (e.g., matrix inversion) and allows the use of general-purpose tensor computing hardware (e.g., tensor core [107]), which can also be adapted to different computing infrastructure (e.g., parallel processing and distributed computing [32]).

- It has a flexible structure which is inspired by the human brain. It can be arranged and combined in a variety of graphical topologies to create various models that are suitable for different inputs, different outputs, different tasks, in including additional information, and etc.

- It utilizes a simple and effective optimisation method (i.e. the back-propagation algorithm) which is suitable for diverse objective functions that are designed for supervised learning, semi-supervised learning, unsupervised learning, self-supervised learning, reinforcement learning, classification tasks, regression tasks, and so on. This optimisation method can also be applied to extremely large model learning, thus opening up the possibility of training huge models with billions of parameters to increase the model accuracy [28, 23, 171].

- It is suitable for big data, and its training complexity only increases linearly with the amount of training data. Data augmentation is also easy to implement in this training scheme, which adequately uses the annotated data, and helps to marginalise out the uncertainties from disturbing factors (e.g., noise, reverberation), thus it enhances the generalisation capability of the model.

- It is a parametric model, and the trained model can perform inferences directly on new inputs. One can also perform operations such as retention, freezing, deletion, quantisation of parameters according to task requirements (e.g., neural network pruning to reduce its size and computational complexity, or partially initialising the model with pre-trained parameters for transfer learning).

However, while entering into the era of big data and high-precision large models, many challenges still need to be considered. For example, trade-offs exist between model accuracy and computational complexity, between the amount of training data and the training time, between the interpretability and obscure, between the search for the optimal topology and the searching time, etc. There are many studies exploring one or more of these ANN problems, such as extracting information from a large model to a small model through network distillation [74], creating interpretable models [71], learning orthogonal (or disentangled) features [152], integrating knowledge graphs [81], and adding physical constraints to reduce the model's dependence on data and increase generalisation capabilities. In this context, this thesis will explore the applications of deep learning in two speech/audio processing tasks by tackling a few common problems in learning audio representations and sequence modelling.

1.3 Introduction to Sound Source Localisation (SSL)

Sound source localisation, which aims to estimate the azimuth angle, elevation angle and distance of a sound source in a 3-D space, is an intrinsic ability of human beings. This ability makes our lives more vivid, helps us understanding the surrounding environment, and facilitates us focusing on the sound in a certain direction in a noisy environment. In computational speech/audio processing, SSL can be used in many fields, such as noise suppression and speech enhancement front-ends, hearing aids, Virtual Reality (VR) devices and robots to understand and reproduce the physical world, etc.

In an indoor SSL scenario, sound waves are emitted by a source and propagate away from the source. The waves will hit the wall, the ceiling, the floor and the objects in the space, which leads to reflection, absorption, and diffraction of the waves. Using a far-field assumption, the incoming

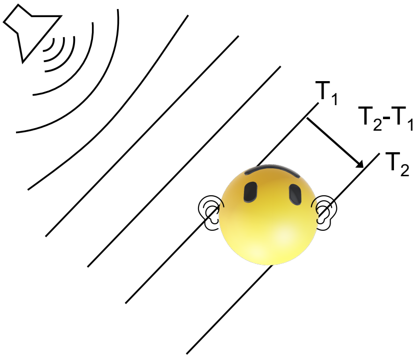


Figure 1.3: Far-field propagating sound wave resulting in a plane wavefront at the human receiver. The propagating wave results in a time difference between the two ears of a human unless the sound source is in the front/back of the human.

wavefronts to a human observer are assumed to be planar. If the source is not directly in the front or back of the receiver (i.e. the azimuth angle does not equal 0 or π), there will be a time difference between a wavefront reaching one ear of the human and then the other ear (as illustrated in Figure 1.3). Due to the shadowing effect of the head, diffraction and absorption of the sound wave occurs, and therefore the intensity of the same sound wave will also be different at both ears. This time difference and intensity difference is represented by two binaural cues, the Interaural Phase Difference (IPD) and the Interaural Level Difference (ILD) [42]. Next to these binaural cues, the human head together with the pinna and the torso act as filters that modify the incident waves, and this filtering effect is charactered by the Head-related Transfer Function (HRTF), known as the monaural cue [126]. The horizontal direction of a sound source (i.e. azimuth), can be largely determined using the binaural cues but vertical direction estimation (i.e. elevation) and depth (i.e. distance) estimation rely highly on the monaural cue and the room acoustic properties, respectively.

Computational SSL has mainly two approaches, with the first being binaural SSL, and the second being multi-microphone SSL. There is a major difference between these two approaches. In binaural SSL, the system mimics the human auditory system and relies on IPD and ILD features, which to a certain extent requires a powerful model that can learn subtle changes in the features. In multi-microphone SSL, multiple microphones can be arbitrarily arranged in a 3-D space, so that the horizontal, vertical, and depth localisation can all be treated in the same way. Also, there are three technique catalogues for multi-microphone SSL, with the first being the MUSIC method [134], which is based on the measurement covariance matrix, the second being the TDOA method [8], which is based on the measurement cross-correlation matrix, and the third being the beam forming method [8], which is based on filter-and-sum operation. A practical remark regarding multi-microphone SSL is that, more microphones will lead to more peripherals and more costs.

1.4 Introduction to Speech Emotion Recognition (SER)

People live emotional lives. Amidst all the activities, relationships, chores, meetings we undertake, it is the emotions and moods we experience that stand out, grasp our attention, and make life bright, or hard to endure: our emotions may indeed also become dysregulated, and form the core symptoms of psychopathology. *How can we understand the complexity with which our emotions are interwoven in our daily lives?* While research into the nature, antecedents, and consequences of emotions has been fruitful, how they change and fluctuate in real life remains elusive. Moreover, the emotions we experience certainly have consequences, they determine for a large part mental flourishing and suffering. On the bright side, happiness, or psychological well-being, relies greatly on how people experience positive and negative emotions in their lives [88]. On the dark side, emotions play a pivotal role in various forms of psychopathology, in particularly mood disorder (such as depression or bipolar mood disorder). Statistics show that there were approximately 253 million people (3.6% of the world's population) having a major depressive disorder in 2013 [84]. Mood disorder on the one hand lets people suffer from body and mental discomfort, on the other hand produces huge burden on the society. Understanding the behaviour of emotions in real life is therefore a crucial challenge with key scientific and societal importance.

In this information age, we can use digital tools to ponder this elusory concept “emotion”. Modern Human Computer Interaction (HCI) systems use image/video, speech, and physiological signals to determine people’s emotion [132]. SER is particularly interesting since vocal expression is a direct and affectionate way of expressing emotions, and it has the advantage of being more accessible than image/video and physiological signals. SER only needs speech signals recorded from a microphone, whereas image/video and physiological signals require either a camera that is positioned directly to the user’s face, or a well-fit wearable device. The applications of SER range from daily services, such as, an assistant robot that can provide emotional communication [31], and an in-car board system that can provide aids or resolve errors in the communication according to the driver’s emotion [136], to diagnostic tools that use the user’s emotion to provide diagnostic information to the physiotherapist for psychopathological treatments [56].

1.4.1 Emotion digitalisation

In order to quantify “emotion”, at the basic level, two types of representations are widely used. The first representation is called *categorical emotion representation* where a few basic and common emotion classes are defined, such as anger, disgust, fear, happiness, sadness, and surprise [49], and an extra neutral class. The second representation is *dimensional emotion representation* where a set of orthogonal axes determine the continuous space of emotions. The most common psychological model, named the “circumplex model”, uses two dimensions called valence (ranging from positive to negative) and arousal (ranging from low to high arousal) [128]. In another psychological model, the PAD emotional state model, a third dimension called “dominance” (ranging from dominance to submissiveness) is added complementarily to the valence and arousal dimensions [109].

1.4.2 Emotional dataset and annotation

There are two main ways of acquiring emotive speech data, the first being performing emotions and the other being spontaneous emotions. On one hand, in recording performing emotive speech, a professional actor is assigned to perform with a pre-defined script in an emotion catalogue (e.g., happy). The annotation for the recording is automatically acquired (i.e. the assigned emotion catalogue). On the other hand, in spontaneous emotions, an emotive speech utterance can be a conversation between two people, a storytelling from the past, or an actor spontaneously performing a scenario. In this case, there are two means of annotation: the first is to let the speakers annotate themselves by means of a questionnaire, e.g., with a symbolic questionnaire as shown in [125]. The second means is to let an emotion analysis expert annotate the video and/or audio recordings. The annotation here can be either an emotion category or continuous valence/arousal values [125]. However, acquiring a large amount of spontaneous emotional speech is time-consuming and expensive.

1.5 Challenges

Speech processing technology faces many challenges while painting a rosy picture of the future. In the framework of speech/audio processing 2.0, these challenges are divided into two folds. The first fold contains the challenges inherent to the audio signal, and the second fold contains the challenges involved to build good data-driven models.

1.5.1 Challenges inherent in the audio signal

Existence of reverberation and noise

In the observable signals, there are acoustic artefacts that affect speech intelligibility, such as reverberation and background noise.

- **Reverberation**: is the persistence of sound in a room after excitation. It is created when a propagating sound wave is partially reflected and absorbed by the room interior, and the original sound wave and all the decayed reflected waves are added together at the receiver. The reverberation can be quantified by the *reverberation time*, which is the time it takes for the sound pressure level to decay by 60 dB after excitation has ended. The reverberation time is short in relatively small rooms, and rooms with more absorbent materials. Perceptually, a short reverberation time is thus less disruptive to the speech intelligibility. In contrary, in relatively large rooms equipped with highly reflective walls, the reverberation time is long and the accumulated reverberation reduces the comprehensibility of the speech signal (e.g., talking loudly in a church).

- **Background noise**: may exist in an uncontrolled recording setting. The background noise can be generated from varying sources, such as, traffic noise, fan noise, electromagnetic coupling noise, interfering speakers and so on. Background noise is often spatially distributed and generally not stationary. In addition, unlike speech we know the frequency range, the frequency spectrum of some of the background noise can cover the entire audible frequency range, and can be distributed uniformly (i.e. white noise), logarithmically linearly (i.e. pink noise), and even arbitrarily. With a low SNR, the signal of interest will be masked by the noise and will thus become less audible and interpretable.

1.5.2 Challenges in data-driven modelling

Unlike physical modelling, in data-driven modelling, there is no underlying physical law to support the training, nor is there a mathematical model of the nature of the signal. In particular with an end-to-end model, which is based on a DNN, the entire feature extractor and model needs to be trained with data. This makes model training susceptible to over-fitting to the training data and having low generalisation capabilities. It is therefore challenging to make data-driven models to learn truly relevant information. For example, using training data with background noise, the data-driven models are likely to fit irrelevant noise characteristics rather than the real signal we are interested in. We discuss a few challenges in applying data-driven models within the scope of speech/audio processing.

Modelling of long sequences

When dealing with speech signals, temporal dependencies are particularly crucial. This importance is reflected in two folds. First, if a sequence is reversed in time, it will lose its meaning. For example, one can only recognise the meaning of a word if the phonemes pronounced in the correct chronological order. Second, sequences that are time-dependent will have a certain correlation across time. This correlation can be at a low level, close to the signal, e.g., two consecutive samples will have similar amplitudes, or at a high and abstract level, e.g., the sound /l/ is more likely to be followed by a vowel rather than a consonant.

From a modelling perspective, a good speech information retrieval model thus needs to be able to represent this temporal dependency. Moreover, this temporal dependency can be long in some tasks. For example, in a speech recognition task, the model needs to learn the internal structure from a sequence of audio features (or raw waveform samples) to recognise the corresponding spoken phonemes. Since a phoneme typically has a duration of about 30 ms, the recognition model might need to learn the dependency that has a span of 30 ms, i.e. equal to the length of a phoneme. However, in the case of SER, the story will be more complex. Research shows that the time constants of emotion dynamics can range from just a few seconds to over an hour [77], thus it requires the SER model to capture correlations in the speech signal with a very long span (in tens of seconds or minutes), which is more challenging than the 30 ms span compare to speech recognition.

Distributional shift

A very important hypothesis in statistics is called the independent and identically distributed (i.i.d) hypothesis which means that the observed data used to train and test a model are independently sampled from the same underlying distribution. However, in many real-life scenarios, this assumption is difficult to meet. Take SER as an example, if we assume that all people express their emotion via speech using a same underlying mechanism (i.e. speech is generated from the same distribution), then if we can sample data independently via this mechanism, we can train a data-driven model that learns the mechanism of speech emotion. This is apparently a chicken and egg problem since we don't know the precise mechanism of the generation of emotive speech, e.g., we don't precisely know how often factors such as health condition, language, speaker identity, gender, age, influence this process. Thus the i.i.d assumption will be violated in such cases:

- **Violating the independently distributed assumption:** this implies we cannot acquire samples from the whole population that represents all the influencing factors. And the existing speech emotion datasets are a subset of the whole population, thus the data has certain correlation with each other, e.g., the speech data is all recorded from healthy people.
- **Violating the identically distributed assumption:** this occurs when we train a data-driven model with a certain dataset, and then apply it on testing data that comes from a shifted distribution (i.e. not identically distributed). This is a very common case where for example the testing speaker has not been seen during the training, the training set consists of different language speech from the testing set, the training set data is acquired with different recording devices that have different electroacoustic responses from the testing recordings, and so on.

In summary, practically, we are facing the challenge of using a small portion of data sampled from the whole underlying distribution (which including all kinds of influencing factors and their variations) for training a data-driven model that should generalise to a broad variety of unseen conditions, which we refer to as the *distributional shift* problem.

1.6 Solution strategy

To alleviate the problem of the inherent reverberation and noise in speech signals, conventional speech processing methods attempt to explore the system properties (e.g., use a mathematical room acoustic model for dereverberation [6]), the noise characteristics (e.g., estimate the noise and remove it from the signal [3]), and the intrinsic structure in the speech signal (e.g., use a sequence model such as the Hidden Markov Model (HMM) to model speech structure thus without modelling irrelevant noise [59]).

The conventional methods estimate linear models on short time-domain segments or frequency-domain bins, and therefore have limited capacity and are unable to capture a very long-term temporal structure. Statistical models also provide solutions to this and show robustness to reverberation [108, 165]. Some subspace methods transfer the measurement space to a subspace in which the signal and the noise components are separated [134, 73]. From a representation learning point of view, these subspace points can be considered as speech/audio representations or as features. These audio features can be close to signal-level (i.e. low-level features), or can represent abstract knowledge (i.e. high-level features). For example in speaker recognition, the signal-level features are the MFCC features. These features start from a mel-scale frequency spectrum in which each mel frequency band represents perceptually similar intensities based on the hearing sensitivity of the human ear, and then a Discrete Cosine Transform (DCT) is applied to extract important spectral information, thereby greatly reducing the amount of irrelevant information. Reynolds and Rose then propose to use Gaussian Mixture Model (GMM) to model the speakers' MFCC distribution [122]. Each Gaussian component in the GMM represents a model of the underlying acoustic class, and the combination of the GMM components thus characterises a speaker (i.e. how the speaker MFCCs distribute among the underlying acoustic classes), and can be considered as a higher level feature than the MFCC. In [41], the authors use the concatenation of the GMM means (referred to as the supervector in [41]) as a high-dimensional representation and then use it to extract the i-vector, which is a low-dimensional subspace representation of the total variability (i.e. speaker variability and channel variability to a universal background model). The i-vector feature brings each speaker model into a common subspace and models the speaker and channel variations, and can thus be considered as a high-level feature.

If we look at the problem from an another angle, that is, the presence of reverberation and noise changes the clean data distribution and consequently cause a distribution shift then it make sense to merge the challenge to eliminate reverberation and noise into the distributional shift challenge and solve both in a unified approach.

To this end, the solution strategy proposed in this thesis follows a representation learning perspective, by first constructing high-level representations that reflect the intrinsic structure of the data, and then using these for downstream tasks. This scheme can be flexible, that is, the high-level features can be extracted from audio features, signal-level features, or even from raw audio waveform samples, and then followed by a sequence model (to explore the temporal dependencies in the high-level features), or globally pooled (e.g., average across time) to generate an utterance-level feature.

1.7 Contributions and outline

Both the representation learning and the sequence modelling can be build upon a unified DNN model whose parameters are learned in a purely data-driven approach. In this thesis, we employ this strategy and propose novel DNN methods for robust binaural SSL, cross-language SER, and end-to-end SER as outlined below:

Introduction to DNN

Sub-objective: To give a brief introduction to DNN topics relevant to this thesis.

In Chapter 2, we first provide a brief introduction to DNNs, which includes introductions to the perceptron, the feed-forward Neural Network (NN), the Convolutional Neural Network (CNN), and the Recurrent Neural Network (RNN). Next, we present the Stochastic Gradient Descent (SGD) algorithm with back-propagation that can be used to optimise the DNN model. Finally, for the training practice, we provide an introduction to methods for model validation, hyper-parameters tuning and enhancing the generalisation capability of the DNN model.

Robust binaural SSL

Sub-objective: To propose a non-linear dimension reduction technique that can preserve sound source proximities from binaural cues.

In Chapter 3, we dive into manifold learning and propose a data-driven method for binaural SSL. We implement this method using a non-linear DNN architecture called the "*siamese*" network, which can learn a parametric mapping that transfers the binaural feature space to a low-dimensional embedding space where the sound source location is preserved. Unlike the baseline Laplacian Eigenmaps (LEM) method that estimates the source location proximity in the input space, the proposed method learns a new metric in the embedding space, in which the model is robust to mismatched training-testing conditions, additive noise, unseen room acoustics, and small training sets. Chapter 3 is based on publication:

- Tang, D., Taseska, M., and van Waterschoot, T., "Supervised Contrastive Embeddings for Binaural Source Localization", *In Proc. 2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2019, pp. 358-362, doi: 10.1109/WASPAA.2019.8937177.

Cross-language SER

Sub-objective: To improve the performance of data-driven models in cross-language/cross-corpus SER settings.

In Chapter 4, we tackle the cross-language SER problem that occurs when data-driven models trained on general acoustic features or in end-to-end approaches perform badly in case the testing data is in a different language than the training set data, as well as when they come from different datasets. To alleviate this problem, we propose an end-to-end DNN model based on transfer learning for cross-language SER. We use the wav2vec 2.0 pre-trained model to transform audio time-domain waveforms in different languages, with different speakers and in different recording conditions into a feature space shared by multiple languages, thereby reducing the language variabilities. Next, we propose a new Deep-Within-Class Covariance Normalisation (Deep-WCCN) layer that can be inserted into the DNN model and aims to reduce other variabilities including speaker variability, channel variability and so on. The whole model is fine-tuned in an end-to-end manner on a combined loss and is validated on datasets from three languages (i.e. English, German, Chinese).

End-to-end SER

Sub-objective: To design a sequence model that can learn long-term temporal dependencies relevant to end-to-end SER.

In Chapter 5, we identify the problem of modelling very long sequences (particularly in end-to-end learning for SER) and propose a novel end-to-end DNN architecture based on the concept of dilated causal convolution with context stacking for SER. Firstly, the proposed model consists only of parallelisable layers and is hence suitable for parallel processing, while avoiding the inherent lack of parallelisability occurring with the RNN layers in state-of-the-art end-to-end SER models. Secondly, the design of a dedicated dilated causal convolution block allows the model to have a receptive field as large as the input sequence length, while maintaining a reasonably low amount of parameters and computational cost. Thirdly, by introducing a context stacking structure, the proposed model is capable of exploiting short-term temporal structures through a local sub-network, and long-term temporal dependencies through a global sub-network, in which both sub-networks are related through local-conditioning. This approach hence provides a unified model structure for representation learning and sequence modelling, which is an alternative to the state-of-the-art end-to-end CNN and Long Short-term Memory (LSTM) based SER systems. Chapter 5 is based on the following publications:

- Tang, D., Kuppens, P., Geurts, L. and van Waterschoot, T., “End-to-end speech emotion recognition using a novel context-stacking dilated convolution neural network", *EURASIP Journal on Audio, Speech, and Music Processing*, 18(2021), 2021, doi: 10.1186/s13636-021-00208-5.

- Tang, D., Kuppens, P., Geurts, L. and van Waterschoot, T., "Adieu recurrence? End-to-end speech emotion recognition using a context stacking dilated convolutional network", *In Proc. 28th European Signal Processing Conference (EUSIPCO)*, January 2021, doi: 10.23919/Eusipco47968.2020.9287667.

In Chapter 6, we conclude the thesis with a summary, and then some suggestions for future research are given.

Chapter 2

Introduction to DNNs

2.1 Introduction

Deep Neural Networks (DNNs) are Artificial Neural Network (ANN) models that consist of more than one layer between the input layer and the output layer, and the ANN is a bio-inspired model which is built up by nesting simple non-linear transformations. The basic element in the ANN model is a *neuron* (i.e. a node) that has inputs and outputs, and can be connected together with other blocks to create a network. The design of the network topology is a big research area in machine learning, and may require domain-specific knowledge. There are three types of ANN layers that are often used, the feed-forward neural network, the Convolutional Neural Network (CNN), and the Recurrent Neural Network (RNN).

We will firstly give a brief introduction to the most common ANN layers, then the Stochastic Gradient Descent (SGD) algorithm and back-propagation that are used for optimising the DNNs are presented. Finally, a few topics in the practical methodology of tuning the DNNs will be discussed.

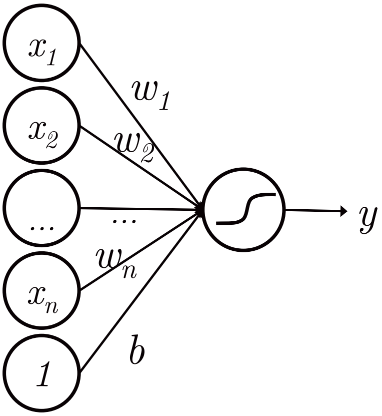


Figure 2.1: An illustration of a perceptron.

2.2 DNN architectures

2.2.1 Perceptron

The perceptron only contains one neuron and is illustrated in Figure 2.1. It is essentially representing a dot product between the vector of n weights $\mathbf{w} = [w_1, w_2, w_3, \ldots, w_n]^T$ and the n-dimensional input $\mathbf{x} = [x_1, x_2, x_3, \ldots, x_n]^T$ plus a bias b, to which a simply non-linear transformation function σ (i.e. activation function) is applied,

$$y = \sigma(\mathbf{w}^T \mathbf{x} + b) \tag{2.1}$$

There are many choices for the activation function [66], for example:

- **Rectified Linear Unit (ReLU)**: is one of the most popular activation functions because despite not being differentiable at zero, it has a gradient with favourable properties that is easy to compute. It is defined as

$$\mathrm{ReLU}(z) = \max\{0, z\}, z \in \mathbb{R} \tag{2.2}$$

- **Sigmoid**: has a long tradition as an activation function in ANNs since it has a range $(0, 1)$ of which the delimiting values correspond to the Bernoulli distribution outputs (i.e. 1 or 0). The sigmoid function is defined as:

$$\sigma(z) = \frac{1}{1+\exp(-z)} = \frac{\exp(z)}{1+\exp(z)} = 1 - \sigma(-z), z \in \mathbb{R} \tag{2.3}$$

- **Tanh**: has a range of $(-1, 1)$, which can be used for normalised outputs of a regression problem. It is defined as,

$$\tanh(z) = \frac{\exp(z) - \exp(-z)}{\exp(z) + \exp(-z)}, z \in \mathbb{R} \tag{2.4}$$

- **Softmax**: generalises the sigmoid function to Multinoulli output distributions, and is widely used in a classifier to represent the probability distribution over C classes.

$$\text{Softmax}(z_i) = \frac{\exp(z_i)}{\sum_{j=1}^{C} \exp(z_j)}, z_i, z_j \in \mathbb{R} \tag{2.5}$$

  where $\sum_{i=1}^{C} \text{Softmax}(z_i) = 1$, and i, $j \in \{1, 2, 3, \ldots, C\}$.

- **Linear activation**: is sometimes used to refer to the situation when there is no non-linear activation function applied to the output of a neuron. It can also be used when representing a regression output.

2.2.2 Feed-forward neural network

A feed-forward neural network consists of multiple layers that are stacks of neurons, as illustrated in Figure 2.2. Each neuron takes inputs from the previous layer (or from the input data in the first layer), and generates an output for the next layer (or the final output in the last layer), and can be considered as a perceptron. As an example, a feed-forward neural network with one hidden layer (as in Figure 2.2), consists of nested linear transformations with non-linear activation functions:

$$\mathbf{y} = \sigma(\mathbf{W}_2^T\ \sigma(\mathbf{W}_1^T\mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2) \tag{2.6}$$

where $\mathbf{W}_i$ is the weight matrix where the columns are the weights associated with each neuron in layer i, $\mathbf{b}_i$ is the bias associated with

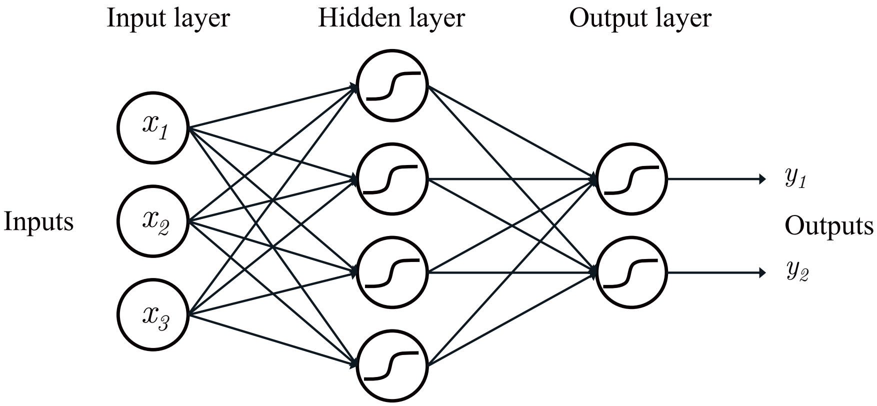


Figure 2.2: A feed-forward neural network with one hidden layer.

neurons in layer i, and the activation function σ is applied element-wise to a vector argument.

The feed-forward neural network is a very flexible model and one can add more and more layers to it. However, with large input dimension and a large number of layers, it may lead to an enormous amount of parameters, which subsequently slows down the computations for training and inference and causes the model to easily over-fit the training data. Actually, a feed-forward neural network with one hidden layer and a sigmoid activation function is a universal estimator [76], but the generalisation capability of the model, which has big practical significance in present-day machine learning tasks.

A feed-forward neural network has three limitations in practical scenarios. Firstly, it cannot be processed in parallel because each layer needs to wait for all the neurons in the previous layer to be computed before it can be processed. Secondly, it cannot be applied to inputs of variable length, in other words, the length of the input needs to be determined at the training time and cannot be changed. Thirdly, for very long inputs, the feed-forward neural network results in a large amount of parameters, it requires $(m+1)\,n$ parameters for a layer that has m inputs and n outputs, which not only reduces the speed of computation, but also makes it tend to over-fit the training data. For example, in the case of a 1 s raw audio input waveform that is sampled at 16 kHz, if the first hidden layer of a feed-forward neural network contains 1024 neurons, it will result more than 16 million of parameters only in this first layer.

2.2.3 CNN

Motivated by the need to increase computational efficiency of the ANN, the CNN is designed to leverage three ideas of *sparse interactions*, *parameter sharing* and *equivariant representations* [94, 66].

The CNN layer is based on the discrete convolution operation in (2.7),

$$y(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t-a). \tag{2.7}$$

where x is the layer input, and w is the layer filter (or kernel). Since the convolution operation is commutative, the convolution operation in (2.7) can be rewritten as

$$y(t) = (w * x)(t) = \sum_{a=-\infty}^{\infty} x(t-a)w(a). \tag{2.8}$$

and the filter then can be implemented by an ANN layer.

An example of the 1-D CNN layer is shown in Figure 2.3 where the filter size is 3. The filter is essentially a “perceptron” that takes three successive inputs, and generates one output. The leading zeros in the input x are added to maintain the same output length as the input in the example. Adding zeros in front or at the back of the input is referred as *zero padding*. To be noted, the input is not reversed in computing the convolution as in (2.8). This is because the kernel w is learnable from data, and the learning algorithm will learn the appropriate values of the kernel even the input is not reversed. The filter then will move forward, and the step size is referred as *stride*, which is equal to 1 in the example in Figure 2.3.

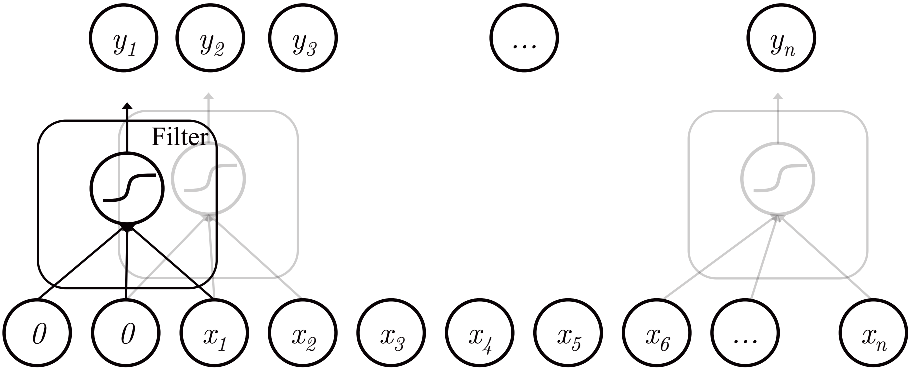


Figure 2.3: An illustration of the 1-D CNN layer with front zero-padding.

We refer to the motivations stated at the beginning of this section, and we will first elaborate on the *sparse interactions* idea. The CNN layer has a complexity of kn parameters with filter size k and n outputs, and since k is normally several orders of magnitude smaller than the input size m, it largely reduces the parameter size compared to the feed-forward neural network. Second, the same CNN filter is moving across the input resulting in the filter parameters being shared across "regions" in the input, that is *parameter sharing*, which leads to a further reduction in the parameter size. Third, due to the *parameter sharing* property, the convolution operation will generate the same outputs for an input region even if that input region is shifted (left or right). This invariance to translation is referred to as the *equivariant representations* property.

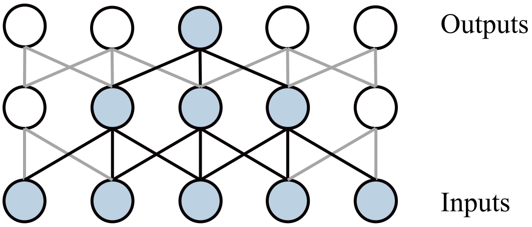


Figure 2.4: The receptive field is 5 for a two-layer CNN with filter size of 3 and stride equal to 1.

The output value y_n in the example in Figure 2.3 is influenced by 3 inputs, and generally the size of influencing inputs is the *receptive field* of y_n. If another CNN layer with filter size 3 is added on top of y, the *receptive field* of the model will become 5, as illustrated in Figure 2.4.

To increase the *receptive field* of a CNN, one can skip a few samples in-between two successive inputs. The number of inputs to skip is a time lag for time-series data, and it is defined as the *dilation number*. A large *dilation number* results in a large receptive field as illustrated in Figure 2.5 which is a dilated causal CNN. The causal property restricts the CNN only to take past inputs and this property is widely required in time-series modelling.

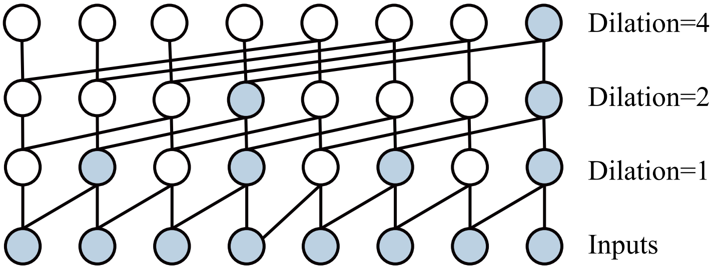


Figure 2.5: Example of a dilated causal CNN.

In addition to the excellent modelling properties of CNNs, it is also possible to parallelise and distribute the CNN computations, which provides it with better practical implications for real-world scenarios.

The CNN also facilitates 2-D inputs (e.g., images), but the 2-D CNN is beyond the scope of the thesis. More information can be found in [66].

2.2.4 Pooling

A pooling layer can be considered as a special CNN layer. Instead of performing a matrix multiplication and a non-linear transformation as in a CNN layer, the pooling layer calculates a summary statistic of the filter inputs. For example, instead of performing (2.1) as a CNN filter, a pooling filter computes the maximum value among the filter inputs in a *max pooling* layer and computes the average value among the filter inputs in a *average pooling* layer. A 1-D average pooling layer with filter size of 3 and stride equal to 1 is illustrated in Figure 2.6.

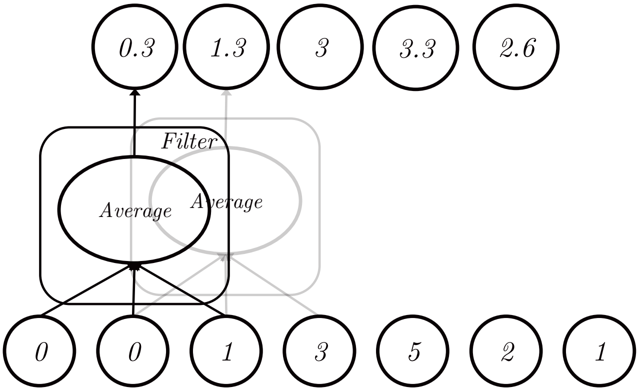


Figure 2.6: An illustration of the 1-D average pooling layer with filter size of 3 and stride equal to 1.

Pooling layers on one hand can reduce the input size, and thus allow to produce more compact representations, a property that is widely used behind CNN layers. On the other hand, a pooling layer helps to make the representation approximately invariant to small translations of the inputs. This invariance property provides robustness in for example a pattern recognition system where we aim to find if a pattern exists in the input.

2.2.5 RNN

The RNNs are a family of Neural Network (NN) layer architectures that contain recurrent connections (i.e. current inputs contain information from the past outputs) and are used for sequential data to explore the temporal dependencies.

The idea behind RNN is somewhat analogous to that of a the Kalman filter that is based on the Markov chain which represents a chain of "states". The Kalman filter models the transition between successive states with a linear state-transition matrix and allows to predict the current state from the previous state and external inputs, associated with an input affine transformation matrix, also taking into account process noise and measurement noise. The vanilla RNN somewhat resembles a non-linear state-space model where the state transition model is non-linear.

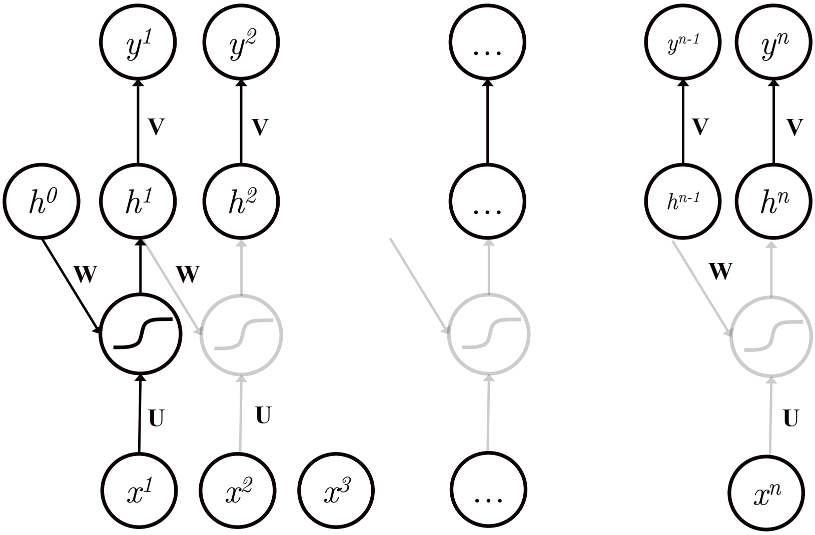


Figure 2.7: An example of a time-unfolded RNN.

An illustration of a time-unfolded RNN adapted from [66] is shown in Figure 2.7. The RNN state-transition and output equation are as follows,

$$\mathbf{h}^{(t)} = \tanh(\mathbf{b} + \mathbf{W}^T\mathbf{h}^{(t-1)} + \mathbf{U}^T\mathbf{x}^{(t)}) \tag{2.9}$$

$$\mathbf{y}^{(t)} = \mathbf{c} + \mathbf{V}^T\mathbf{h}^{(t)} \tag{2.10}$$

where $\mathbf{b}$ and $\mathbf{c}$ are bias vectors (corresponding to the process noise and measurement noise in Kalman filter, respectively), transformation matrix $\mathbf{W}$ is associated with the previous system state, $\mathbf{U}$ is the input transformation matrix, and $\mathbf{V}$ is the output transformation matrix. The $\tanh(\cdot)$ is the aforementioned non-linear activation function applied element-wise. The $\mathbf{h}^{(t)}$ is the hidden state vector at time-step t, and $\mathbf{h}^{(0)}$

is the initial system state that can be initialised randomly, as zeros, or based on prior knowledge.

A big challenge in sequential data modelling is to model the long-term dependencies, which is especially crucial for end-to-end learning on audio time-domain waveforms. For example in Speech Emotion Recognition (SER), the time constants of emotion dynamics range from just a few seconds to over an hour [162], hence a good SER model should then be able to model sufficiently long temporal dependencies in this scenario. However, the vanilla RNN suffers from the so-called gradient vanishing/exploding problem as shown below. If we simplify the hidden state adaptation as in [66],

$$\mathbf{h}^{(t)} = \mathbf{W}^T \mathbf{h}^{(t-1)} \tag{2.11}$$

and the current hidden state $\mathbf{h}^{(t)}$ is a recursive updating of the initial state $\mathbf{h}^0$,

$$\mathbf{h}^{(t)} = (\mathbf{W}^t)^T \mathbf{h}^0 \tag{2.12}$$

if $\mathbf{W}$ admits an eigendecomposition of the form

$$\mathbf{W} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T \tag{2.13}$$

then equation (2.12) can be expressed as,

$$\mathbf{h}^{(t)} = \mathbf{Q}^T \mathbf{\Lambda}^t \mathbf{Q} \mathbf{h}^0 \tag{2.14}$$

where the eigenvalues are raised to the power of t leading to the eigenvalues with magnitude less than one to decay to zero and eigenvalues with magnitude greater than one to explode [66], which is referred to as the gradient vanishing/exploding problem.

This gradient vanishing/exploding problem of the RNN has been alleviated by introducing the Long Short-term Memory (LSTM) layer where a hidden state self-updating-loop (controlled by another hidden unit) with a forgetting mechanism is designed. This structure can learn a dynamic time scale of integration of the information in the hidden state, and has been shown to be successful in many sequence modelling applications [75].

Another problem is that the RNN has a sequential type of processing and can not be parallelised. This makes it slow in the case of long inputs and limits its ability to increase the model capacity by deepening the number of RNN layers in the DNN.

2.3 DNN optimisation

2.3.1 SGD

With the construction of a DNN, we get a parametric model that can estimate a mapping between the input $\mathbf{x}$ and the target output $\mathbf{y}$. The performance of the DNN is measured by an *evaluation metric*, which is calculated on the unseen data (e.g. testing data). This metric in general can be arbitrary and task-dependent, however, it can be intractable and not necessarily has an explicit form which makes it unsuitable to be used to calculate the gradients for updating the model parameters. Instead, we can indirectly optimise the model using a loss function in which optimising the loss function can improve the *evaluation metric*. Assume we employ the Maximum likelihood estimation (MLE), which tries to maximum the conditional probability parametrised by $\boldsymbol{\theta}$,

$$\boldsymbol{\theta}_{ML} = \arg\max_{\boldsymbol{\theta}} P(\boldsymbol{Y}|\boldsymbol{X};\boldsymbol{\theta}) \tag{2.15}$$

where $\boldsymbol{X} = \{\boldsymbol{x}_i, \ldots, \boldsymbol{x}_N\}$ and $\boldsymbol{Y} = \{\boldsymbol{y}_i, \ldots, \boldsymbol{y}_N\}$ represent all the inputs and targets, respectively. If we take logarithm of the likelihood, then the maximum likelihood estimator can be re-write as follows

$$\boldsymbol{\theta}_{ML} = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{N} \log P(\boldsymbol{y}_i|\boldsymbol{x}_i;\boldsymbol{\theta}) \tag{2.16}$$

where N is the population size. Dividing (2.16) by the training set size M, we can express it as an expectation (with notation $\mathbb{E}$) with respect to the empirical data distribution $\hat{p}_{data}$ where the examples are assumed to be independent and identically distributed (i.i.d) and denoted as $\mathbf{x}, \mathbf{y} \sim \hat{p}_{data}$,

$$\boldsymbol{\theta}_{ML} = \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{\mathbf{x},\mathbf{y}\sim\hat{p}_{data}} \log P(\mathbf{y}|\mathbf{x};\boldsymbol{\theta}) \tag{2.17}$$

and the loss function to be minimised is defined as

$$\mathcal{J}(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{x},\mathbf{y}\sim\hat{p}_{data}} \mathcal{L}(\mathbf{x},\mathbf{y},\boldsymbol{\theta}) = \frac{1}{M}\sum_{i=1}^{M} \mathcal{L}(\mathbf{x}_i,\mathbf{y}_i,\boldsymbol{\theta}) \tag{2.18}$$

where $\mathcal{L}$ is the per-example negative log-likelihood $\mathcal{L}(\mathbf{x},\mathbf{y},\boldsymbol{\theta}) = -\log P(\mathbf{y}|\mathbf{x};\boldsymbol{\theta})$.

In the big data era, a training set may contain an enormous amount of data. For example, the XLS-R pre-trained speech model has been trained on nearly half a million hours of speech [10], which is practically infeasible to evaluate the loss on the whole training data set.

The idea of SGD is to estimate the expectation in equation (2.18) by a small subset of training samples. The subset of samples, which are randomly sampled without replacement, composes a *mini-batch* with size m. A large value for m may give an accurate estimation of the expectation in equation (2.18), but the value of m may be constrained by memory size and computation time. Then the parameter gradients $\mathbf{g}$ can be evaluated through the mini-batch loss,

$$\mathbf{g} = \frac{1}{m}\sum_{i=1}^{m} \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{x}_i,\boldsymbol{y}_i,\boldsymbol{\theta}) \tag{2.19}$$

and finally the SGD algorithm updates the parameters with a learning rate α as follows,

$$\boldsymbol{\theta}_k \leftarrow \boldsymbol{\theta}_{k-1} - \alpha\mathbf{g} \tag{2.20}$$

where k is the iteration index. The learning rate α is a hyper-parameter that can be tuned via a validation set, when a too large α generally leads to the optimisation algorithm oscillating across the loss function and being unable to converge, and a too small α will slow down the convergence. There are also adaptive learning rate methods such as the RMSProp [151] and the Adam [87] methods.

2.3.2 Back-propagation

The idea of back-propagation [127] is based on the construction of a graph of the forward operations through the DNN model, which is referred to as as *computation graph.* The gradients for each parameter in each operation can then be back-propagated via the chain rule of differentiation from the end of the computation graph to the beginning.

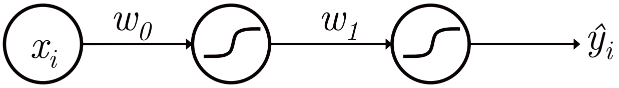


Figure 2.8: A simple NN to demonstrate back-propagation.

A simple NN consisting of two neurons, one input and one output is shown in Figure 2.8. First, we calculate the forward pass of the network *computation graph* with randomly initialised weights w_0 and w_1, where z_0, a_0 and z_1 represent intermediate variables,

Forward pass:

$$x_i \rightarrow w_0 x \rightarrow z_0 \rightarrow \sigma(z_0) \rightarrow a_0 \rightarrow w_1 a_0 \rightarrow z_1 \rightarrow \sigma(z_1) \rightarrow \hat{y}_i \tag{2.21}$$

If we employ the least-squares loss with respect to the ground truth y as,

$$\mathcal{L} = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2 \tag{2.22}$$

then by applying the chain rule, the partial derivatives of the parameters w_0 and w_1 for the ith training sample in a mini-batch can be calculated as follows,

Backward pass:

$$\frac{\partial \mathcal{L}}{\partial w_1} = \frac{1}{m}\sum_{i=1}^{m} \frac{\partial \mathcal{L}}{\partial \hat{y}_i}\frac{\partial \hat{y}_i}{\partial z_1}\frac{\partial z_1}{\partial w_1} \tag{2.23}$$

$$\frac{\partial \mathcal{L}}{\partial w_0} = \frac{1}{m}\sum_{i=1}^{m} \frac{\partial \mathcal{L}}{\partial \hat{y}_i}\frac{\partial \hat{y}_i}{\partial z_1}\frac{\partial z_1}{\partial a_0}\frac{\partial a_0}{\partial z_0}\frac{\partial z_0}{\partial w_0} \tag{2.24}$$

with each term is easy to calculate as follows,

$$\frac{\partial \mathcal{L}}{\partial \hat{y}_i} = -2(y_i - \hat{y}_i) \tag{2.25}$$

$$\frac{\partial \hat{y}_i}{\partial z_1} = \sigma'(z_1) = \sigma(z_1)(1 - \sigma(z_1)) \tag{2.26}$$

$$\frac{\partial z_1}{\partial w_1} = a_0 \tag{2.27}$$

$$\frac{\partial z_1}{\partial a_0} = w_1 \tag{2.28}$$

$$\frac{\partial a_0}{\partial z_0} = \sigma(z_0)(1 - \sigma(z_0)) \tag{2.29}$$

$$\frac{\partial z_0}{\partial w_0} = x_i \tag{2.30}$$

To be noted that the results in equations (2.26) and (2.29) are properties of the sigmoid function which do not hold for other activation functions. Finally, giving the inputs x_i and their corresponding targets y_i for $i = 1, \ldots, m$ in a mini-batch, the gradients in (2.25) to (2.30) can be evaluated using the forward pass results and the model parameters are updated using (2.23) and (2.24) as,

$$w_0^{(k)} = w_0^{(k-1)} - \alpha\frac{\partial \mathcal{L}}{\partial w_0} \tag{2.31}$$

$$w_1^{(k)} = w_1^{(k-1)} - \alpha\frac{\partial \mathcal{L}}{\partial w_1} \tag{2.32}$$

Back-propagation with the SGD algorithm enables the training of a DNN with arbitrary topology and mathematical operations as long as explicit derivatives can be calculated, and hence forms the foundation of modern deep learning.

2.4 Validation and regularisation for DNN

One of the most crucial requirements in machine learning is generalisation capability, that is, how to learn a model that will perform equally well on the training set and on unseen data. There are various methods to tackle this problem, which are generally referred to as regularisation methods. One of the most common regularisation methods is the inclusion of parameter norm penalties (i.e. L1 and L2 regularisation) in the loss function. When it comes to deep learning, a great variety of regularisation methods are available, for example the use of a dropout layer, data augmentation, weight decay etc. We will cover a few of the most common regularisation methods in this section.

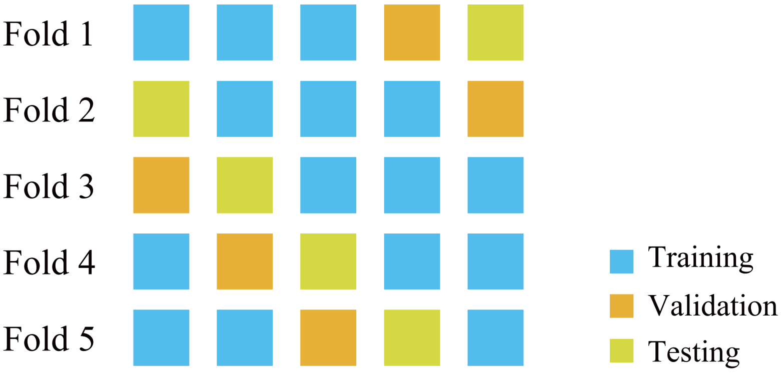
2.4.1 Model validation and hyper-parameter selection

A deep learning task may contain a number of hyper-parameters, such as the learning rate, the batch size and so on. These hyper-parameters can be set through prior knowledge, and hoping the model to perform well on unseen data, or can be tuned through some tuning strategies. Also, the training of a DNN is iterative, and as argued in section 2.3.1, the model performance is measured by some pre-defined metrics which are not necessarily the same as the loss that the model is trained on. In this case, a stopping criterion is needed to guide us when to stop training.

After having a dataset at hand, a common practise is to divide the dataset into three partitions, that are the training set, the validation set, and the testing set. Mostly, a random partitioning of ratio 7:2:1 or 6:2:2 is used, but for specific tasks, the user can design their own validation and testing sets that reflect the real target data characteristics.

With the above partitions, a DNN can be trained using only the training set, and evaluated by the pre-defined evaluation metrics at a constant interval (e.g. every 50 iterations or every epoch, which is when all training data has been used once), on the validation set. Then, the model with the best validation performance is chosen as it has the best generalisation capability. Since we can only see the iteration number that corresponds to the best validation performance after proceeding with more iterations, a trace back is required to first run the optimisation for a large number of iterations, and then choose the model at the iteration that gives the best validation performance, which is a procedure known as *early stopping*. Finally, with the selected model, a final model performance is evaluated on the testing set.

The procedure for hyper-parameter tuning is the same as described above. In the training step, a set of hyper-parameters is set manually, then the model is trained, evaluated, and finally selected based on the best validation performance. The user can search for the optimal hyper-parameters by for example linear search, grid search, random search, etc. In addition, hyper-parameter values can also be determined based on similar works.



Fold 1	🟦	🟦	🟦	🟧	🟨
Fold 2	🟨	🟦	🟦	🟦	🟧
Fold 3	🟧	🟨	🟦	🟦	🟦
Fold 4	🟦	🟧	🟨	🟦	🟦
Fold 5	🟦	🟦	🟧	🟨	🟦

🟦 Training
🟧 Validation
🟨 Testing

Figure 2.9: A 5-fold cross-validation partitioning.

In some applications, the dataset contains a limited number of samples which makes the partitioning challenging. One can then use N-fold cross-validation which allows to fully use the dataset and ensure a valid evaluation of the generalisation capability of the model. In N-fold cross-validation, we first divide the whole dataset into N parts, as shown in Figure 2.9 with N equal to 5. Then, in each fold, we use some parts for training, some other parts for validation and the rest for testing (in the example we use 3 parts for training, 1 part for validation and 1 part for testing). Next, the division of the parts is rotating, and a new part is used for validation, another part for testing, and the rest for training as illustrated in Figure 2.9. The use of the validation set in N-fold cross-validation is still intended for model selection (e.g., early stopping), and finally, after running N-folds, the testing results from each fold are collected and averaged to yield the final performance of the model.

In a more extreme scenario where only a few data samples are available, one can also use leave-one-out cross-validation where every time only 1 sample is used for validation and 1 sample for testing.

Next to manual hyper-parameter tuning using those model validation methods (e.g., with grid search, line search, random search and etc.) one can also rely on some automatic hyper-parameter tuning toolboxes, such as Hyperopt [18], which formulates the hyper-parameter search as an optimisation problem and solves it with using numerical optimisation algorithms. The drawback of the hyper-parameter optimisation algorithms is that they need to often re-run the training experiment (which can be slow with the DNN). In most cases, these hyper-parameter optimisation algorithms randomly select values from a user-defined range of hyper-parameters as initialisation. But it is very likely that these initial hyper-parameter values are not optimal solutions. In order to reduce the time used by the hyper-parameter optimisation algorithms on these trivial experiments, the user can provide a smaller search range for the hyper-parameters based on domain expertise, and design automatic stopping criterion if the training loss in one search trial is decreasing too slowly.

2.4.2 Bagging ensemble method

Bagging stands for bootstrap aggregating [25] and is an ensemble technique to reduce the generalisation error by combining multiple weak models that are not identically trained (e.g., with different initialisation, different model hyper-parameters, or different model configurations) on different sub-sets of the whole training data set.

While a rigorous mathematical foundation can be found in [25], a more intuitive explanation of the bagging ensemble method refers to in resemblance with the “ask the audience” practice in the “Millionaire” TV shows where a large number of people in the audience give their answers to the same question, and the ensemble answer marginalises out to some extent the uncertainties in the choices.

Formally, to apply bagging, one needs to firstly construct k different sub-sets that consist of samples randomly taken from the whole training set with replacement. Then, a model i is trained on the sub-set i. Finally, the predictions of the trained models are aggregated to generate one prediction for each input data point.

The bagging ensemble method is an extremely powerful and reliable technique for better generalisation. Some machine learning methods involve the bagging ensemble method in the algorithm, for example, the Random Forest (RF) [26] method.

A boosting technique assigning learnable weights to the ensemble models is sometimes combined with bagging ensemble methods, such as, the AdaBoost [57] and Gradient Boosted Decision Trees (GBDT) methods [58].

2.4.3 Dropout

The bagging ensemble method is good for increasing the generalisation capability of a model, however, it requires a large amount of computational power when applied to DNN.

Dropout [143] provides a way to regularise a DNN with low cost by considering many sub-networks of a large DNN. Different than the normal bagging ensemble method where each candidate model is independent from the others, dropout allows the sub-networks to share the same parameters in a large DNN.

To achieve this, dropout randomly removes non-output-layer neurons from the DNN with a pre-defined probability p, which is a hyper-parameter that can be tuned through the validation methods mentioned in section 2.4.1.

Apart from the benefits dropout brings, it does reduce the effective modelling capacity of the DNN model. One can increase the size of the DNN model to counteract this effect, but this may require more computational resources and more iterations to train.

For training with a small dataset, dropout may be less effective to increase the generalisation capability of the DNN. It may then be more interesting to consider other regularisation methods or try other machine learning methods (e.g. RF and statistical models).

2.4.4 Data augmentation for speech/audio applications

For training a data-driven model, more annotated data will always be beneficial to the model generalisation capability since it marginalises out the uncertainties in a large population space. Data augmentation builds upon this idea and aims to artificially create more annotated data.

Data augmentation generally has been applied to various machine learning tasks, such as image processing tasks [20], speech processing tasks [21] and so on. In this thesis, we introduce a few data augmentation techniques in speech/audio processing applications:

- **Adding background noise:** is to mix the audio recording with pre-recorded background noise, which can be real-life noise, such as, traffic noise, fan noise, babble noise, or it can be synthetic noise such as Gaussian noise with varying standard deviation.
- **Adding reverberation:** in most of the real-life scenarios, reverberation will be present. One can generate artificial room impulse responses or use measured room impulse responses and convolve these with the original audio recording to create multiple recordings with different room effects.

- **Filtering:** one can emphasise low frequencies, specific frequency bands, or high frequencies with a low-pass filter, band-pass filter, or high-pass filter, respectively.
- **Pitch shift:** is to shift the pitch of a speech signal without changing the tempo and the content.
- **Time stretching:** is to elongate or shorten the speech recording without changing its pitch.
- **Time inversion:** is to reverse the audio waveform along the time axis which of course does change the content of the speech or audio.
- **Sliding window:** consists in applying a sliding window on the entire audio waveform to generate multiple shorter waveforms.

The type of augmentation chosen should reflect the real data condition where the model will be employed. To obtain the most general model, all types of augmentation effects might need to be combined, yet considering all possible combination may be costly. Note that, data augmentation should not change the data ground truth, for example, pitch shifting might be beneficial for speech recognition where speaker pitch is considered as a disturbing factor which needs to be marginalised out, but it is not applicable to speaker recognition and gender recognition where the pitch is a strong feature to distinguish different speakers.

2.4.5 Weight decay

In addition to regularisation methods that penalise the parameter vector norms (e.g., the l1-norm or l2-norm), weight decay provides another way to constrain the parameters' magnitude. Different from l1 and l2 regularisation where an extra loss term is added to the loss function, the weight decay method directly adjusts the parameter values according to their magnitude multiplied with a pre-defined weight decay constant, such that for example in the case of l2 regularisation, the loss function and the update equation in (2.18) and (2.20) are modified as follows,

$$\mathcal{J}(\boldsymbol{\theta}) = \frac{\lambda}{2}\boldsymbol{\theta}^T\boldsymbol{\theta} + \frac{1}{M}\sum_{i=1}^{M}\mathcal{L}(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\theta}) \tag{2.33}$$

$$\boldsymbol{\theta}_k \leftarrow \boldsymbol{\theta}_{k-1} - \alpha\mathbf{g} - \alpha\lambda\boldsymbol{\theta}_{k-1} \tag{2.34}$$

where λ is the weight decay constant. And writing in another way,

$$\boldsymbol{\theta}_k \leftarrow (1 - \alpha\lambda)\boldsymbol{\theta}_{k-1} - \alpha\mathbf{g} \tag{2.35}$$

When applying the weight decay method, some of the parameters in the over-parametrised DNN will become very small, thus have no significant effect on the overall decision making.

2.5 Conclusion

In this section, we have briefly introduced the common layers of a DNN, the model training methods, the model validation and hyper-parameter tuning methods and a few regularisation techniques for DNN. In addition to these fundamentals, DNN is still a very active area of research nowadays in which building generalised and robust models is a promising direction. However, there are many other machine learning models and algorithms, such as Support Vector Machines (SVMs) [24, 144], decision tree based methods [118], logic programming [111], Bayesian methods [15], etc., and it is important to choose the best method for the problem at hand, rather than using DNN exclusively.

Chapter 3

Towards learning robust contrastive embeddings for binaural sound source localisation

This chapter is partially based on the following publication:
Tang, D., Taseska, M. and van Waterschoot, T., Supervised Contrastive Embeddings for Binaural Source localisation, *in Proc. 2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2019, pp. 358-362, doi: 10.1109/WASPAA.2019.8937177.

Abstract

Recent deep neural network based methods provide accurate binaural source localisation performance. These data-driven models map measured binaural cues directly to source locations hence their performance highly depend on the training data distribution. In this paper, we propose a parametric embedding that maps the binaural cues to a low-dimensional space where localisation can be done with a nearest-neighbour regression. We implement the embedding using a neural network, optimised to map points that are close to each other in the latent space (the space of source azimuths or elevations) to nearby points in the embedding space, thus the Euclidean distances between the embeddings reflect their source proximities, and the structure of the embeddings forms a manifold, which provides interpretability to the embeddings. We show that the proposed embedding generalises well in various acoustic conditions (with reverberation) different from those encountered during training, and provides better performance than unsupervised embeddings previously used for binaural localisation. In addition, the proposed method performs better than or equally well as a feed-forward neural network based model that directly estimates the source locations from the binaural cues, and it has better results than the feed-forward model when a small amount of training data is used. Moreover, we also compare the proposed embedding using both supervised and weakly supervised learning, and show that in both conditions, the resulting embeddings perform similarly well, but the weakly supervised embedding allows to estimate source azimuth and elevation simultaneously.

Keywords— Manifold learning, Non-linear dimension reduction, Siamese neural network, Binaural sound source localisation, Deep learning

3.1 Introduction

Sound source localisation is aiming to estimate a sound source position in terms of azimuth, elevation and distance. A large part of the source localisation literature focuses on the azimuth and elevation estimation only, hence this is also the scope we adopt in this paper. The human auditory system is capable of localising acoustic signals using binaural cues such as the Interaural Phase Differences (IPDs) and Interaural Level Differences (ILDs) [22]. Computational localisation algorithms in robot audition [8], hearing aid[55], virtual reality [86], etc., aim at mimicking this process and therefore estimate the binaural cues from binaural microphone signals. The binaural microphones are typically two identical microphones that are mounted at the entries of two ear canals of an artificial head. In a sound source localisation scenario, the human/artificial head together with the pinna and the torso act as filters that modify the incident sound waves. This filter effect is crucial for sound source localisation, especially vertical sound source localisation (i.e. elevation estimation), and can be characterised by the Head-related Transfer Function (HRTF) [126].

Acoustic artefacts such as noise and reverberation, introduce uncertainties in the binaural cues. Although the existence of reverberation can aid distance localisation [126], the resulting noisy and reverberant binaural cues make sound source localisation challenging. Traditionally, robustness to reverberation has been tackled with statistical model-based approaches [108, 165, 106], which outperform lookup tables and template matching methods that rely on an anechoic assumption [85, 120]. Some works propose to estimate the direct-path relative transfer function, which encodes the source azimuth information, in order to avoid the contamination of audio from reverberation noise, however, this type of methods highly rely on the onset of the source acoustic events [98].

In contrast, data-driven approaches are able to learn the non-linear functions that map binaural cues to source locations [39]. Recently, Deep Neural Networks (DNNs) has been used to learn the relationship between azimuth and binaural cues, by exploiting head movements to resolve the front-back ambiguity [105], and by combining spectral source models to robustly localise the target source in a multiple sources scenario [104]. Additionally, a few works use DNNs to enhance the binaural features so that they can eliminate reverberation and additive noise [168, 114]. In [167, 161], the authors utilize DNNs to directly map the audio spectrogram or its raw waveform to the source azimuth in an end-to-end manner, which is also applicable to reverberant and noisy environments. However, those works only consider source azimuth estimation and the localisation is done by classification (i.e. the predictions can only be in a pre-defined grid).

A different data-driven approach was used in [43, 42], where the relationship between source locations and binaural cues was modelled with a probabilistic piecewise linear function. By learning the function parameters, sources can be localised by probabilistic inversion. An implicit assumption of the piecewise linear model in [43, 42] is that similar source locations result in similar binaural cues. The same assumption is also used in non-parametric source localisation algorithms based on manifold learning in [92, 93]. In this paper, we focus on data-driven source localisation approaches, inspired by low-dimensional manifold learning [92, 93].

Manifold learning in sound source localisation is aiming to find a non-linear transformation that transforms acoustic measurements to a low-dimensional representation that preserves the source locality information. Manifold learning methods in [92, 93] rely on smoothness in the measurement space with respect to the underlying source locations, an assumption that might generalise poorly to varying acoustic conditions. The uncertainties in the binaural cue measurements introduced by reverberation, introduce variations in the measurement space neighbourhoods that might not be consistent with their source locations. To preserve neighbourhoods in term of the source location, we are inspired by the *"siamese"* neural network in the machine learning community that is optimised with a contrastive loss function [68]. This particular model learns a similarity metric defined in the latent space (i.e. written digit classes and orientation of air plane pictures in [68]). This paradigm, which doesn't rely on an explicit neighbourhoods definition in the measurement space, is suitable for problems that have a large amount of classes and in each class there are only a few training examples, such as face verification [146, 36] and signature verification [27], and can also be used in sound source localisation. We have proposed and published earlier a regression method for binaural sound source localisation based on the *"siamese"* neural network and contrastive loss in [149]. This method converts binaural cues into a low-dimensional embedding, and there is a small Euclidean distance between the embeddings obtained from binaural cues of similar source locations. A similar work using triplet loss somewhat resembles our idea [113], but in their work, a model directly maps the binaural cues to source location predictions, and pre-defined proximity for both positive and negative cases (i.e. points with similar and dissimilar source locations) have to be present at the same time for the triplet loss.

In this paper, we first propose an update on the model architecture introduced in [149], and then validate its robustness with respect to three aspects:

1. mismatched audio content between the training and testing sets,
2. the presence of unknown reverberation and noise,
3. and the availability of only a small amount of annotated training data,

through abundant experiments in fixed and varying acoustic scenarios, respectively. Afterwards, we extend our method to a weakly supervised learning scheme, where the annotation of source directions (i.e. azimuth and elevation) is no longer required for training the embeddings, but only the relative source position proximity is needed for any pair of training examples. Unlike the supervised approach proposed in [149], which treats azimuth and elevation estimation in two separate tasks, this weakly supervised embedding can be used to estimate both the source azimuth and elevation at the same time, and providing a good visualisation of the manifold.

The paper is organized as follows. In Section 3.2, we first revise the binaural cue extraction and formulate the source localisation problem. Then, in Section 3.3, we provide a brief overview of the related manifold learning work that has been applied in binaural sound source localisation. Next, the proposed method is presented in Section 3.4 and finally, experimental results are shown in Section 3.5.

3.2 Data model and problem formulation

3.2.1 Binaural cue extraction

Let $s_1(\tau)$ and $s_2(\tau)$ denote the signals captured at the left and right microphones in a binaural recording setup in a noisy and reverberant environment. In this work, we extract the binaural cues in the Short-time Fourier transform (STFT) domain, as in [120, 42].

Let $S_1(t,k)$ and $S_2(t,k)$ denote the STFT coefficients of $s_1(\tau)$ and $s_2(\tau)$, where t and k are the time frame and frequency index, respectively. At a time-frequency bin (t,k) an ILD α_{tk} and an IPD ϕ_{tk} are defined as

$$\alpha_{tk} = 20\log_{10}\frac{|S_1(t,k)|}{|S_2(t,k)|}, \quad \phi_{tk} = \angle\frac{S_1(t,k)}{S_2(t,k)}. \tag{3.1}$$

Assuming that a single sound source is active, we follow the binaural feature extraction approach from [42], and compute time-averaged ILDs and IPDs across T frames as follows

$$a_k = T^{-1}\sum_{t=1}^{T}\alpha_{tk}, \quad p_k = T^{-1}\sum_{t=1}^{T}\exp(j\phi_{tk}). \tag{3.2}$$

By concatenating the ILDs, and the real and imaginary parts of the IPDs in selected frequency ranges $[k_1,k_2]$ and $[k_3,k_4]$, the binaural information is summarised in a measurement vector $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^D$,

$$\boldsymbol{x} = [a_{k_1},\ldots,a_{k_2},\, \mathcal{R}\{p_{k_3}\},\mathcal{I}\{p_{k_3}\},\ldots,\mathcal{R}\{p_{k_4}\},\mathcal{I}\{p_{k_4}\}]^{\mathrm{T}} \tag{3.3}$$

with dimensionality $D = k_2 - k_1 + 2(k_4 - k_3)$.

It is known that IPDs carry reliable location cues below 2 kHz [22], while ILDs contribute to localisation at higher frequencies as well [42]. Hence, we used the ranges $\frac{f_s}{K}[k_1, k_2] = [200, 7000]$ Hz for ILDs and $\frac{f_s}{K}[k_3, k_4] = [200, 2500]$ Hz for IPDs, where f_s denotes the sampling frequency and K is the Discrete Fourier transform (DFT) size used in the STFT, and $k_i = \text{round}(\tilde{k}_i), i = 1, 2, 3, 4$, where the round() operation rounds $\tilde{k}_i$ to the closest integer. For a typical audio recording with sampling rate $f_s = 16\,\text{kHz}$, and the DFT size $K = 1024$, the dimensionality D is equal to 729 (i.e. a 729-dimensional feature vector $\boldsymbol{x}$).

3.2.2 Measurement to embedding transformation

From the above binaural cue extraction process, a pair of signals $s_1(\tau)$ and $s_2(\tau)$ is associated to a vector $\boldsymbol{x} \in \mathcal{X}$. We refer to $\mathcal{X}$ as the *measurement* space. Let the unknown source location be denoted by $\boldsymbol{u} \in \mathcal{U}$. We refer to $\mathcal{U}$ as the *latent space*. $\mathcal{U}$ is one-dimensional if one considers azimuth or elevation separately, or two-dimensional if the localisation angles are considered simultaneously. Given a training set of N pairs $\mathcal{T} = \{(\boldsymbol{x}_i, \boldsymbol{u}_i)\}_{i=1}^{N}$, the localisation problem consists of finding a function h

$$\hat{\boldsymbol{u}} = h(\boldsymbol{x}), \quad h : \mathcal{X} \to \mathcal{U}. \tag{3.4}$$

that accurately maps measurements to latent variables. Although, one can implement h with a powerful non-linear model (e.g., a DNN), the proposed approach of first transforming the measurement space to an embedding space and then performing the localisation in the embedding space comes with several advantages:

1. Learning the transformation from measurement space to embedding space does not necessarily require the latent space annotation information, thus enables the possibility of semi-supervised learning and weakly supervised learning.
2. The low-dimensional embedding can preserve the latent space neighbourhood relationships (in which the Euclidean distance in the embedding space roughly corresponds to the latent space “semantic” relationship) and the embedding eliminates useless information, which can be used to study or visualise the latent space structure. A vanilla example of this is the Principal Component Analysis (PCA).
3. By learning the structure of the latent space, the training of the model will be less dependent on the distribution of the training data. In contrast, if the mapping from measurement space to latent space is learned directly,

the model is more likely to over-fit to the dense part of the training data and its generalisation capability decreases when there is not enough annotated training data.

Therefore, our main objective in this work is to learn an embedding function f that maps the vectors $\boldsymbol{x}$ to a low-dimensional space which preserves latent space neighbourhoods, i.e.,

$$\boldsymbol{z} = f(\boldsymbol{x}), \quad f : \mathcal{X} \to \mathcal{Z} \subset \mathbb{R}^d, \quad d << D. \tag{3.5}$$

We propose a neural network framework to learn a parametric function f that satisfies these properties both in a supervised and weakly supervised manner. A nearest-neighbour regression function $h : \mathcal{Z} \to \mathcal{U}$ is then used for localisation.

3.3 Baseline manifold learning method

If the microphone location in a given room is fixed, the authors in [93] showed that features extracted from binaural signals can be embedded in a low-dimensional space $\mathcal{Z}$, in a way that recovers source locations. The framework in [93] is based on unsupervised manifold learning, in particular, *Laplacian eigenmaps* (LEM) [16].

The Laplacian Eigenmaps (LEM) method defines the neighbourhood relationships of the data using a similarity matrix $\boldsymbol{K} \in \mathbb{R}^{N \times N}$, with entries $\boldsymbol{K}[i, j]$ related to the Euclidean distances $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2$ between feature vectors x_i and x_j, with $i, j \in [1, N]$. One way to compute $\boldsymbol{K}$ is using nearest-neighbours, i.e., $\boldsymbol{K}[i, j] = \boldsymbol{K}[j, i] = 1$ if $\boldsymbol{x}_i$ is among the M nearest neighbours of $\boldsymbol{x}_j$, or if $\boldsymbol{x}_j$ is among the M nearest neighbours of $\boldsymbol{x}_i$ (in Euclidean distance). A second way is using an exponentially decaying kernel function, such as the Gaussian kernel

$$\boldsymbol{K}[i, j] = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2}{\varepsilon}\right), \tag{3.6}$$

where ε is the kernel bandwidth. Such kernel is used for source localisation in [93]. Given the similarity matrix $\boldsymbol{K}$, the neighbourhood-preserving optimisation problem of LEM to find the embeddings $z_1, z_2, \ldots, z_N$ is given by [16]

$$\begin{aligned} &\underset{\boldsymbol{z}_1,\ldots,\boldsymbol{z}_N}{\arg\min} \quad \sum_{i,j=1}^{N} \|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2^2 \, \boldsymbol{K}[i, j], \\ &\text{subject to} \quad \boldsymbol{Z}^T \boldsymbol{D} \boldsymbol{Z} = \boldsymbol{I} \end{aligned} \tag{3.7}$$

which enforces that points x_i, x_j with large similarity $\boldsymbol{K}[i,j]$, are to be mapped to points z_i, z_j with a small Euclidean distance $\|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2$ where $\boldsymbol{D}$ is a diagonal matrix with entries $\boldsymbol{D}[i,i] = \sum_{j=1}^{N} \boldsymbol{K}[i,j]$.

The optimisation problem (3.7) has a closed-form solution, given by the eigenvectors of $\boldsymbol{P} = \boldsymbol{D}^{-1}\boldsymbol{K}$ corresponding to the largest eigenvectors. If $\{\boldsymbol{\psi}_i\}_{i=1}^{N}$ denote the eigenvectors of $\boldsymbol{P}$, with eigenvalues $1 = \lambda_1 > \lambda_2 \geq \ldots, \geq \lambda_N$, the d-dimensional LEM embedding is given by [16]

$$\boldsymbol{z}_i = f(\boldsymbol{x}_i) = [\,\boldsymbol{\psi}_2[i],\ \boldsymbol{\psi}_3[i],\ \ldots,\ \boldsymbol{\psi}_{d+1}[i]]^{\mathrm{T}}\,, \tag{3.8}$$

where the constant eigenvector $\boldsymbol{\psi}_1$ is not included [16, 37] and $[i]$ denotes the vector element index. The LEM embedding f is non-parametric, and the low-dimensional representation $\boldsymbol{z}$ of a new measurement $\boldsymbol{x}$ is obtained as a linear combination of the training points $\{\boldsymbol{z}_i\}_{i=1}^{N}$ [17]. However, this procedure is often insufficiently accurate and represents a disadvantage of LEM and of spectral embeddings in general. One can include every new testing data and re-run the unsupervised training to get a more accurate representation for the new testing data, however, this may prolong the training time, especially for large datasets, and due to the fact that the kernel matrix $\boldsymbol{K}$ is $N \times N$, the computation of eigenvectors will dramatically increase for a large N.

Besides the promising performance of spectral embeddings for localisation [93, 92, 150], their major drawback is the assumption that the neighbourhoods in the measurement space are consistent with the source locations. Although the assumption is shown to hold when all signals are recorded in one room for fixed microphone locations [43, 93, 150], this is not the case when the signals are filtered by various acoustic channels in different enclosures.

3.4 Contrastive embedding for localisation

We propose a parametric embedding, designed to preserve neighbourhoods in terms of sound source locations. Such embeddings are robust to unseen room reverberation and small training set size (e.g. when the training set does not contain the complete latent space annotations). The proposed framework firstly includes the definition of the neighbourhoods, which can be supervised (Section 3.4.1) or weakly supervised (Section 3.4.2) depending on whether one uses the azimuth/elevation label or the source relative proximity. Secondly it includes the transformation from the measurement space to the embedding space by training a DNN which is optimised on a contrastive loss function (Section 3.4.3 and Section 3.4.4). Finally the sound source localisation will be performed in the embedding space using nearest-neighbour regression (Section 3.4.5).

3.4.1 Supervised neighbourhoods definition

Consider two labelled measurements $(\boldsymbol{x}_i, u_i)$ and $(\boldsymbol{x}_j, u_j)$ where u_i and u_j are denoted as scalars since we estimate azimuth and elevation separately. To avoid the phase wrapping ambiguity, we define $d_u(u_i, u_j) = \min(|u_i - u_j|, 360° - |u_i - u_j|)$ denote the shortest possible distance in the latent space $\mathcal{U}$, where u_i, u_j corresponds to the source azimuth or elevation angles in degree. A neighbourhood indicator $y_{ij} \in \{0, 1\}$ is defined as

$$y_{ij} = \begin{cases} 0, & \text{if} \quad d_u(u_i, u_j) > \epsilon_u \\ 1, & \text{if} \quad d_u(u_i, u_j) \leq \epsilon_u, \end{cases} \tag{3.9}$$

for a user-defined threshold angle ϵ_u.

3.4.2 Weakly supervised neighbourhoods definition

As an alternative to directly using the latent space label information to define the neighbourhoods, we can also use the relative proximity between sound sources. Here, we only consider the sound sources at the ball with radius Φ and centred at the receiver, or sources whose relative position to the receiver can be found (then the source locations can be firstly projected onto a ball with radius Φ around the receiver by distance normalisation).

In order to define the weakly supervised neighbourhoods, we can use the physical distance $d_s(S_i, S_j)$ between two sound sources S_i and S_j which corresponds to the Euclidean distance between the Cartesian coordinate vectors of S_i and S_j . Similarly,

$$y'_{ij} = \begin{cases} 0, & \text{if} \quad d_s(S_i, S_j) > \epsilon_s \\ 1, & \text{if} \quad d_s(S_i, S_j) \leq \epsilon_s, \end{cases} \tag{3.10}$$

for a user-defined threshold distance ϵ_s. The threshold ϵ_s and ϵ_u are related as ϵ_s represents the arc length of the angle ϵ_u on a circle with radius Φ and hence,

$$\epsilon_s \approx \epsilon_u \cdot \Phi \cdot \pi/180° \tag{3.11}$$

In particular, in our proposed method, one can also implicitly define the similarity indicator y'_{ij} by using it as a training data label. For example, consider a scenario when multiple recordings are acquired from excitations at each of the pre-defined sound source locations, then y'_{ij} equals to 1 for recordings acquired at the same or at close source locations, and y'_{ij} equals to 0 for recordings acquired at different or far source locations.

3.4.3 Contrastive loss

We seek to learn a parametric function $f_W : \mathcal{X} \to \mathcal{Z} \subset \mathbb{R}^d$, with parameters W, that maps $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ to their low-dimensional embeddings $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$. If $y_{ij} = 1$, the Euclidean distance $\|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2$ should be small, and if $y_{ij} = 0$, then $\|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2$ should be large. For a given embedding function f_W, we have

$$\|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2 = \|f_W(\boldsymbol{x}_i) - f_W(\boldsymbol{x}_j)\|_2. \tag{3.12}$$

A *contrastive loss function* over the parameters W, tailored for neighbourhood preservation has been proposed in [68] for non-linear dimensionality reduction, and is given by

$$\begin{aligned} L(W) = \sum_{i=1}^{N}\sum_{j=1}^{N} \Big(& y_{ij}\, \|f_W(\boldsymbol{x}_i) - f_W(\boldsymbol{x}_j)\|_2^2 \\ & + (1 - y_{ij}) \max(0, \mu_{ij} - \|f_W(\boldsymbol{x}_i) - f_W(\boldsymbol{x}_j)\|_2)^2 \Big). \end{aligned} \tag{3.13}$$

The parameter μ_{ij} is a positive real-valued margin, such that $\mu_{ij}/2$ can be interpreted as the same radius of circles centered on $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$. If the circles intersect and $y_{ij} = 0$, the two dissimilar pairs are too close in the embedding space, thus increasing the *contrastive loss* in (3.13). On the other hand, if $y_{ij} = 1$, large distances are penalised, enforcing f_W to preserve neighbourhoods.

Intuitively speaking, during the training, each example in a mini-batch is subjected to two "forces". One force is between the similar pairs, pulls them closer to each other in the embedding space. The other force between dissimilar pairs is repulsive and it pushes the dissimilar pair away from each other in the embedding space (if they are too close when $\|f_W(\boldsymbol{x}_i) - f_W(\boldsymbol{x}_j)\|_2 < \mu_{ij}$). During training, the embeddings are moving according to the forces they encounter, and thus will eventually lead to an equilibrium (i.e. convergence). Globally, the embedding space convergences to a manifold. Since the forces are subjected to latent space similarities, this will result in meaningful distances between each pair of embeddings (i.e. the distance between a pair of embeddings somewhat indicates the proximity of their corresponding sound sources).

It is important to note that in [68], where the *contrastive loss* was first proposed for classification, $\mu_{ij} \equiv \mu$ is a constant margin. In our application, the latent space of azimuths and elevations is continuous. To accurately preserve its geometry, we propose an adaptive margin as follows,

$$\mu_{ij} = \frac{\exp(d_{ij})}{\exp(d_{ij}) + 1}. \tag{3.14}$$

As d_{ij} decreases, the margin μ_{ij} decreases as well. One can compute d_{ij} either in a supervised manner using the azimuth/elevation, thus $d_{ij} = d_u(u_i, u_j)$, or in a weakly supervised manner, where $d_{ij} = d_s(S_i, S_j)$. In the case that there is no quantitative measure in the latent space, a constant margin can be used (e.g., $\mu = 1$).

3.4.4 Learning the embedding

We implement f_W with a DNN as shown in Figure 3.1(a). The DNN architecture consists of two fully-connected hidden layers with D neurons in each layer. Between the fully connected layers, we add batch-normalisation layers [80] to speed up the convergence and dropout layers to prevent the model from over-fitting [143]. The output layer has 3 neurons, corresponding to a 3-dimensional embedding space i.e., $d = 3$. The hidden neurons have *Sigmoid* non-linear activations, and the output neurons have linear activations. In order to train the DNN model to minimise the cost function in (3.13), we use the *siamese* architecture that was proposed in [27] and used for various tasks in [36, 68]. This special DNN architecture consists of two identical branches that are sharing the same model parameters. Taking a pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ as an input, the measurements $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are passed through the branches (one per branch) and hence produce their corresponding embeddings $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$. Then the cost is evaluated in (3.13) using the neighbourhood indicator y_{ij} and the outputs $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ of the branches. Finally, the gradient per model parameter is calculated and back-propagated to update the model parameters. Depending on which definition for the neighbourhood indicator is used, we call the corresponding embedding Supervised Contrastive Embedding (SCE) if the supervised neighbourhoods definition is used, or Weakly supervised Contrastive Embedding (WSCE) if the weakly supervised neighbourhoods definition is used.

A key aspect of the proposed framework is the selection of pairs $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for training. For small datasets, one could consider all pairs and proceed with training on all training data pairs. However the polynomial growth of the number of pairs results in memory problems even for moderately large datasets. To solve this problem, we use mini-batches and calculate the neighbourhood indicator y_{ij} for every pair of examples in each mini-batch. To be noted, we suggest to choose a large enough batch size so that there are both similar pairs and dissimilar pairs in one batch. Because a randomly selected mini-batch generally contains examples from sources of different locations (i.e. those examples will be defined as dissimilar pairs), if the batch size is too small, the probability of having similar pairs in a batch will be very low, so that the loss will be inaccurately evaluated and thus slow down the convergence rate. Intuitively, if there is no similar pair in a batch, the embeddings will not be

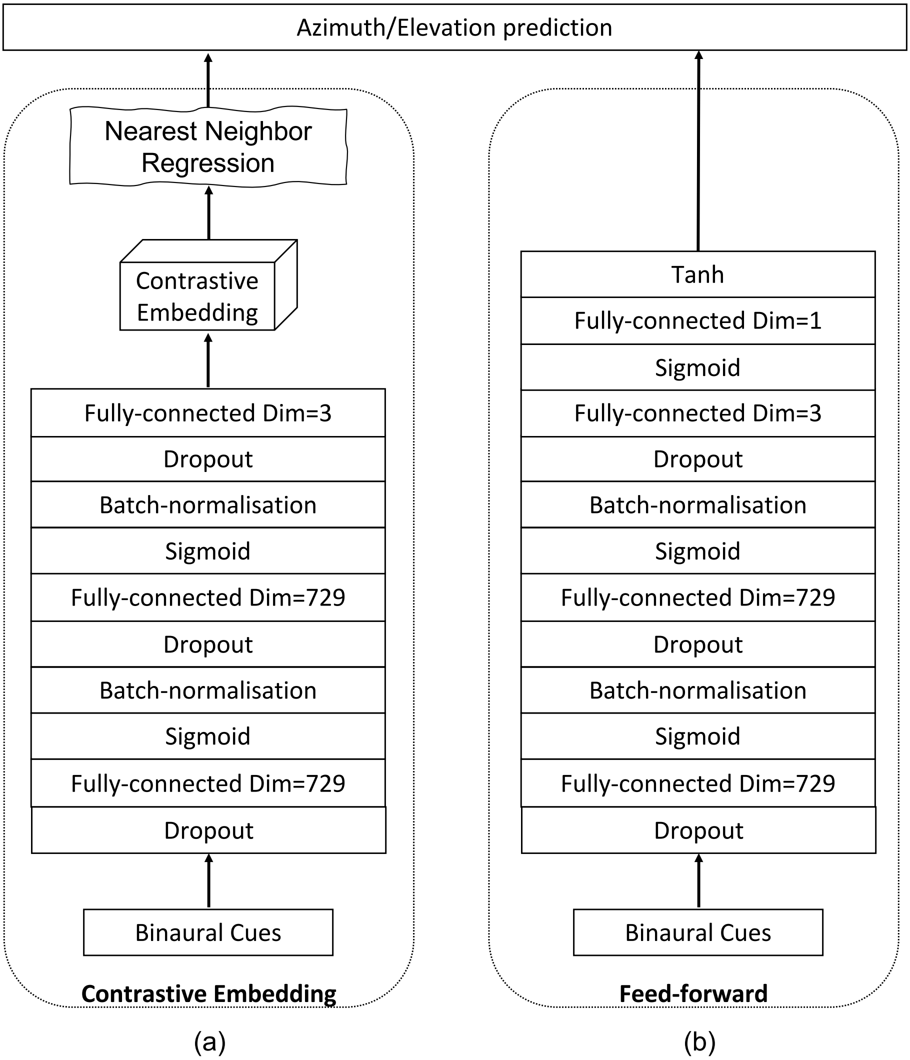


Figure 3.1: Model architecture. The proposed contrastive embedding model in (a), and the feed-forward model in (b).

subjected to a pulling force to their similar points. This would lead to the embeddings that just randomly reside in the embedding space and form local clusters.

3.4.5 Nearest-neighbour localisation

Once the weights of f_W are optimised, we compute the embedding of a new $\boldsymbol{x}$ by a forward-pass through the DNN model. Let $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_K$ denote the K nearest-neighbours of $\boldsymbol{z}$ in the training set. The latent variable (azimuth or elevation) is then estimated as

$$\hat{u} = \sum_{i=1}^{K} w_i u_i, \quad \text{with} \quad w_i = \frac{\exp\left(-\frac{\|\boldsymbol{z}-\boldsymbol{z}_i\|_2^2}{\varepsilon}\right)}{\sum_{j=1}^{K} \exp\left(-\frac{\|\boldsymbol{z}-\boldsymbol{z}_j\|_2^2}{\varepsilon}\right)}. \tag{3.15}$$

The bandwidth ε of the exponential kernel is obtained as the median of the squared distances from the K neighbours, i.e.,

$$\varepsilon = \text{median}\left(\|\boldsymbol{z}-\boldsymbol{z}_1\|_2^2, \ldots, \|\boldsymbol{z}-\boldsymbol{z}_K\|_2^2\right). \tag{3.16}$$

Note that if the embedding is accurately preserving neighbourhoods, the choice of regression weights is not critical. For instance w_i can be inversely proportional to $\|\boldsymbol{z}-\boldsymbol{z}_i\|_2^2$. However, in our experiments, the latter generally leads to less accurate location estimates than exponentially decaying weights.

3.5 Experiments

3.5.1 Experimental settings

To evaluate the proposed SCE in terms of the localisation error and robustness, we compare the SCE with two baseline methods:

1. The LEM embeddings [92, 93] with nearest neighbour localisation.
2. A feed-forward neural network which is optimised with the Mean Squared Error (MSE) loss. This feed-forward neural network has the same structure as one of the branches in the proposed *siamese* structure except for an additional output layer with *tanh* activation functions that outputs the source location predictions, shown in Figure 3.1(b). Since the *tanh* activation function has a range of (-1, 1), we normalise the training labels also to the same range by $\hat{u_i} = u_i/180°$, and $i = 1 \ldots N$. Note that, the original labels have a range $[-180°, 180°]$. During testing, the feed-forward predictions are firstly converted back to degree before calculating the localisation errors.

As the neighbourhoods for LEM are defined in the input space, a single embedding is used to estimate both azimuth and elevation. Similarly, our proposed method can be trained to estimate azimuth and elevation simultaneously as well by using the weakly supervised neighbourhoods definition introduced in Section 3.4.2. However, a system with two separately trained embeddings might provide better results for the same amount of data, which we will compare for SCE and WSCE in the later experiments.

For the nearest neighbour regression in (3.15), $K = 5$ neighbours are used in all localisation experiments. A few threshold values ϵ_u in (3.9) and (3.11) are tested for both azimuth and elevation. We choose ϵ_u in $\{5°, 15°, 30°\}$ to have a big span so that we can evaluate its impact on the localisation results. Essentially, ϵ_u is a hyper-parameter that can be tuned with a validation set. We implemented the LEM using a nearest neighbour kernel $\boldsymbol{K}$ with $M = 10$ nearest neighbours, which in our experiments, provided better results than the Gaussian kernel used in [93, 150].

For DNN training, we use the Adam optimiser [87] with a learning rate equal to 10^{-3} that is automatically halved if the validation performance does not improve after 20 epochs. The mini-batch size is set to 128, and this will result 8128 pairs of measurements per mini-batch for training. We select the model based on the best validation performance, and then the selected model is used to calculate the testing set predictions.

All audio files are sampled at 16 kHz. To extract the ILD and IPD features, we use the STFT with a cosine window of 1024 samples at 16 kHz, 75% overlapping.

3.5.2 Datasets

Fixed acoustic conditions

With the first dataset, we want to verify the effectiveness of our proposed methods for preserving the locality information of the audio source when the training and the testing set have different audio content (and different spectral distribution). We employ the CAMIL dataset which consists of binaural recordings and was gathered using a Sennheiser MKE 2002 dummy head in a real-life reverberant room (i.e. a room with a few furnitures and background noise) [42]. To generate recordings that have different azimuth and elevation angles, a loudspeaker (i.e. the source) is placed at a fixed position, 2.7 m from the dummy head (i.e. the receiver). The dummy head is mounted on a step-motor which generates 10800 pan-tilt states. This results in source azimuth and elevation angle in the range [-180 °, 180 °] and [-60 °, 60 °] respectively (with

2 °resolution). To only evaluate the methods in localising frontal sources, we select the recordings that have source azimuth and elevation angle in the range [-90 °, 90 °] and [-45 °, 45 °] respectively. The CAMIL dataset consists of a training set made using white noise (1 s per recording), and a testing set made using 1-5 s speech samples from the TIMIT corpus [61]. We further randomly divide the whole training set into a smaller training set (consisting of 70% samples from the original training set), and a validation set (consisting of the remaining 30% samples from the original training set). Finally, spatially uncorrelated white noise with a Signal to Noise Ratio (SNR) of 15 dB is added to the testing set.

Varying acoustic conditions

With the second dataset, we want to verify the robustness of the proposed methods for varying acoustic conditions. We use the VAST dataset [62] of simulated binaural room impulse responses of a KEMAR dummy head [60, 133]. The training set consists of 16 different rooms with reverberation time 0.1-0.4 s. For each room we select spherical grids of source positions with radii 1 m, 1.5 m and 2 m, centered at 9 predefined receiver positions (inside each room). Similarly to the fixed acoustic conditions in Section 3.5.2, we use 70% of randomly selected data as the training set, and the remaining 30% as the validation set. The receiver's height is fixed at 1.7 m. Then two testing sets are provided:

- *Testing-set-1*: The source and receiver are placed at random positions in the same 16 rooms as the training set.
- *Testing-set-2*: The source and receiver are placed in shoebox rooms of random width and length between 3 × 2 m and 10 × 4 m, with absorption profiles randomly picked from those of the training rooms. Those rooms have reverberation time 0.1 s-0.4 s.

All the training set's and testing sets' Head-related impulse responsess (HRIRs) are simulated using the image source method [4] and provided by the VAST dataset [62].

As in Section 3.5.2, we have only selected recordings that have frontal angles. To focus on the influence of the varying room acoustics while exciting all frequencies, 2 s white noise source signals were considered in this experiment.

3.5.3 SCE for unidimensional source localisation

Tuning the dropout rate

We first determine an optimal dropout rate for both the SCE method and the feed-forward model by line search. We test dropout rate values in $\{0.0, 0.2, 0.5, 0.8\}$, and similarity threshold values ϵ_u for SCE equals to 5° and 15° (denoted by "_sim5" and "_sim15", respectively). The azimuth/elevation localisation error of the validation sets for both the CAMIL dataset and the VAST dataset are plotted in Figure 3.2.

In Figure 3.2 (a) and (b), the azimuth and elevation estimation results for the CAMIL dataset are illustrated respectively. We can observe that the SCE has better validation performance than the feed-forward model for all testing dropout rates, and its localisation error is essentially equal to zero when using either similarity threshold value, i.e. 5° or 15°. The feed-forward model exhibits a clear concave curve in median localisation error and has the lowest localisation error at the dropout rate value of 0.2, thus indicating that a dropout rate equal to 0.2 is an optimal value for the feed-forward method.

In Figure 3.2 (c) and (d), the median azimuth and elevation localisation error for the VAST dataset are illustrated respectively. Both the SCE and the feed-forward model in this case exhibit a concave curve in median localisation error and they both exhibit an optimal dropout rate of 0.2. We also observe that, in the VAST azimuth validation performance, the SCE_sim5 performs equally well as the feed-forward model when dropout rate is 0.2, which is slightly better than SCE_sim15. In the elevation estimation, SCE_sim5 performs the best over the feed-forward model and SCE_sim15.

Based on the validation results, we choose the dropout rate equal to 0.2 for both the SCE and the feed-forward methods for the next experiments.

Comparison with the baseline

In this experiment, we compare the localisation performance of the proposed SCE with the baseline LEM embedding and the feed-forward model. For the proposed SCE, we evaluate a small threshold angle (i.e. $\epsilon_u = 5\,°$) and a large threshold angle (i.e. $\epsilon_u = 15\,°$), denoted by "_sim5" and "_sim15" respectively.

The testing set results are illustrated in Figure 3.3. It can be seen that in the fixed acoustic condition with the CAMIL dataset, the proposed SCE performs better than the LEM embedding and the feed-forward model in terms of

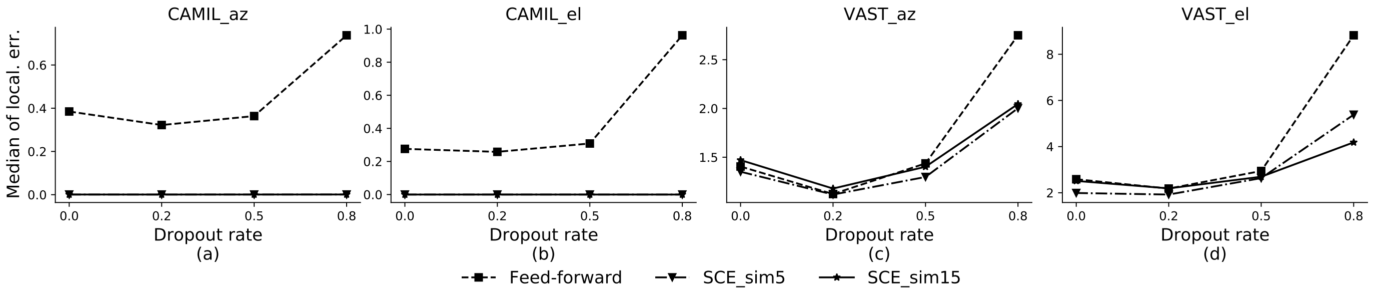


Figure 3.2: Validation performance across different dropout rates. The dropout rate is tested for both the proposed SCE with similarity threshold angle ϵ_u equal to 5°and 15°(denoted by _sim5 and _sim15, respectively), and the baseline feed-forward method. Both methods are tested for source azimuth (denoted by _az) and elevation (denoted by _el) estimation using the CAMIL and the VAST datasets. Localisation errors are in degrees.

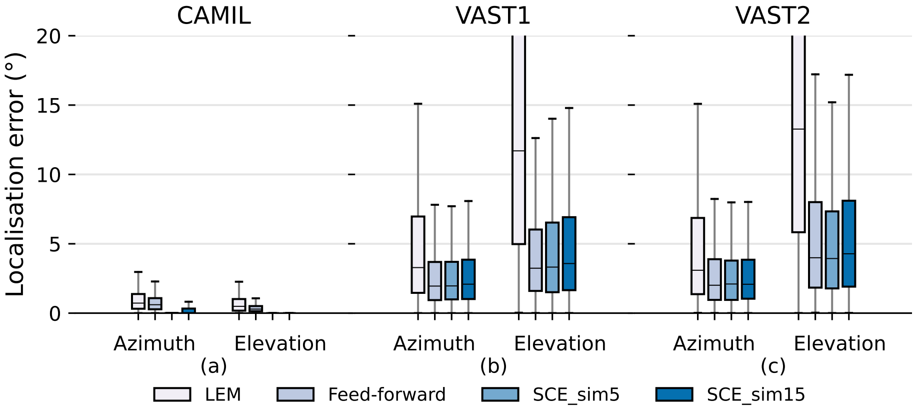


Figure 3.3: Testing set localisation performance for the baseline LEM, the baseline feed-forward and the proposed SCE methods. Localisation errors in (a) the fixed acoustic condition using the CAMIL testing set, and (b), (c) the localisation performance in the varying acoustic condition using the VAST testing sets. "_sim5" and "_sim15" denote the use of similarity threshold angles ϵ_u equal to 5° and 15° respectively.

median error and maximum error. Especially when using the small similarity threshold, the SCE performs excellent, as the SCE_sim5 has almost zero median error in azimuth and elevation estimations. It can also be noted that the feed-forward model performs slightly better than the LEM embedding, with a median error equal to 0.61° and 0.29° for azimuth and elevation respectively, whereas the LEM model has median errors equal to 0.72° and 0.49° for azimuth and elevation respectively. In summary, in the fixed acoustic condition, the proposed SCE can almost perfectly preserve the source location information even when reverberation and additive white noise are present, while the feed-forward model performs better than the LEM embedding, but both exhibit some estimation error. This could be due to the fact that the feed-forward model highly depends on the training data, and due to the presence of audio content mismatch between the training and testing sets, the feed-forward model has some difficulty to generalise to unseen audio contents, thus negatively influencing the localisation performance.

In the varying acoustic conditions with the VAST testing sets, the proposed SCE_sim5 performs slightly better than the SCE_sim15 and equally well as the feed-forward model. The SCE_sim5 and feed-forward model achieve VAST testing-set-1 azimuth median errors equal to 1.96° and 1.95°, VAST testing-set-1 elevation median errors equal to 3.32° and 3.24°, VAST testing-

H0 hypothesis	**VAST 1 AZ**	**VAST 1 EL**	**VAST 2 AZ**	**VAST 2 EL**
There is no difference in localisation error between feed-forward and SCE_sim5	0.3934	**0.0106**	0.0648	0.283
There is no difference in localisation error between SCE_sim5 and SCE_sim15	**0.0305**	**0.047**	0.2247	0.4956

Table 3.1: Wilcoxon signed-rank tests and p-values. Numbers in bold indicates the corresponding test reject H0 for confidence interval equal to 0.05 and in favour of the alternative hypothesis.

	Feed-forward vs. SCE_sim5	**SCE_sim5 vs. SCE_sim15**
VAST 1 AZ	Fail to reject H0 that there is no difference in localisation errors	Reject H0 in favour of alternative that there is difference in localisation errors
VAST 1 EL	Reject H0 in favour of alternative that there is difference in localisation errors	Reject H0 in favour of alternative that there is difference in localisation errors
VAST 2 AZ	Fail to reject H0 that there is no difference in localisation errors	Fail to reject H0 that there is no difference in localisation errors
VAST 2 EL	Fail to reject H0 that there is no difference in localisation errors	Fail to reject H0 that there is no difference in localisation errors

Table 3.2: The conclusions to Wilcoxon signed-rank tests.

set-2 azimuth median errors equal to 2.1° and 2.01°, and VAST testing-set-2 elevation median errors equal to 3.94° and 3.99°, respectively. To statistically verify the model performance, we employ the Wilcoxon signed-rank test on the differences of localisation errors among the feed-forward model, the SCE_sim5 and the SCE_sim15. The p-values are listed in Table 3.1, which are further compared with the confidence interval equal to 0.05. The conclusions are listed in Table 3.2, indicating that there is no statistical difference in localisation errors between the feed-forward model and the SCE_sim5 except in VAST1 elevation predictions, and there is no statistical difference between SCE_sim5 and SCE_sim15 in VAST1 with both azimuth and elevation predictions. In addition, the localisation errors of the SCE_sim5 and the feed-forward model in the VAST testing sets are relatively low (e.g. median error is less than 4°), and in most of the statistical tests there is no significant difference between two models, therefore we can conclude that both models can generalise well to unseen acoustic environments, and show robustness towards reverberation and noise.

The LEM embedding performs the worst in the presence of various reverberations. It achieves median errors equal to 3.3 and 11.7 for azimuth and elevation in VAST test-set-1, respectively, and 3.1 and 13.3 for azimuth and elevation in VAST test-set-2, respectively. This may indicate that the LEM, which is easily affected by geometric distortion in the measurements, is not robust to reverberation.

Reduced training-set

A common problem related to data-driven methods is the model generalisability, or in other words, how can a trained model generalise to unseen data. In the source localisation scenario, the training set may not include training recordings from every pair of azimuth/elevation angles, hence it is desirable that the model can somehow interpolate the predictions that lie in-between the training points. In this experiment, we are aiming to evaluate the robustness of the proposed SCE towards the training size. With a smaller training size, there will be more source locations that are not included in the training. We use a similarity threshold angle $\epsilon_u = 5\,^\circ$ for SCE in this experiment and all methods are conducted with 10%, 25%, 50% and 70% randomly selected training sets. The median localisation errors are illustrated in Figure 3.4.

As illustrated by these results, all methods show a decreasing trend in localisation error when a larger training set is used, however, the median localisation error of the proposed SCE does not vary much with the changing size of the training set, and shows a flatter pattern. Although in the fixed acoustic condition, SCE

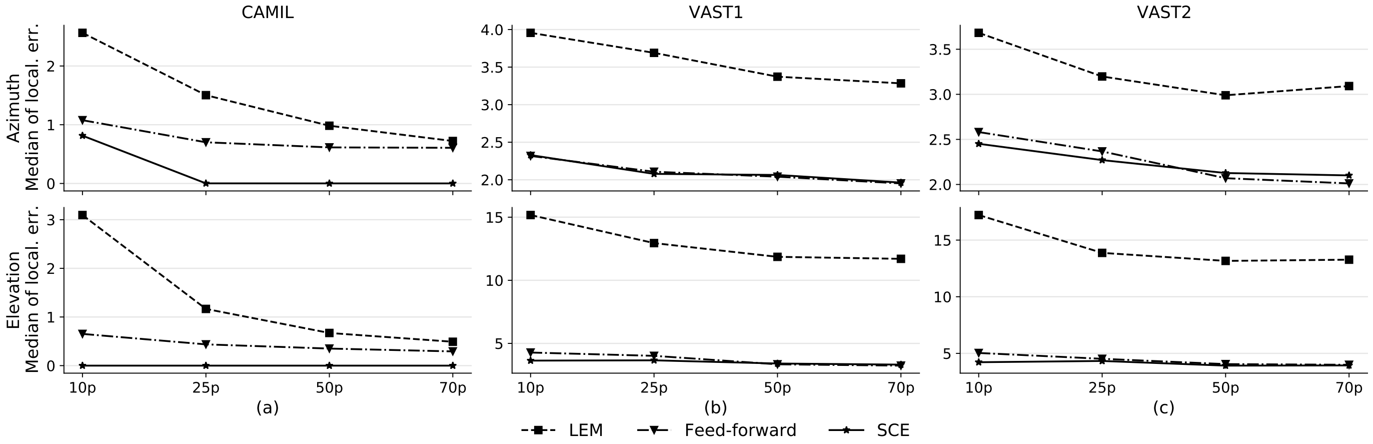


Figure 3.4: Localisation performance on the CAMIL testing set, the VAST testing-set-1 and the VAST testing-set-2. 10p, 25p, 50p, and 70p denote the cases when 10%, 25%, 50%, and 70% of the original training data is used, respectively.

results a in a higher median error when 10% of the training set is used (median azimuth error equal to 0.81°) than when a larger training set is used, the error is still lower than for the other two methods (as feed-forward and LEM achieve median azimuth errors equal to 1.08° and 2.56°respectively when 10% of the training data is used). This allows to conclude that the SCE is more robust to the use of training data that not cover the entire latent space.

The results allow us to hypothesise that the proposed SCE, leveraged by the contrastive loss and the adaptive margin (see Section 3.4.3), is aiming to learn a similarity metric between input binaural cues from the latent space. This similarity metric implies that the underlining structure in the latent space is robust to unseen source locations. In contrast, the feed-forward model tends to transform the measurement space to an abstract high-level space in which the Euclidean distance between embeddings is not necessarily a similarity metric, and thus it is difficult to infer the unseen source locations from this embedding space.

3.5.4 WSCE for multidimensional source localisation

The LEM embedding as well as the proposed WSCE are capable of estimating the sound source azimuth and elevation simultaneously. It should be noted that both the proposed WSCE and the LEM need source annotations in order to localise new examples under the nearest-neighbour localisation framework, thus the localisation phase is still a supervised learning task for both methods.

To explore the learned latent space structure, we test several similarity threshold angles $\epsilon_u \in \{5°, 15°, 30°\}$, indicated as "_sim5", "_sim15", and "_sim30" respectively. Since when calculating the similarity labels, we first normalise the relative source location coordinates to have unit norm (i.e. source coordinates are relocated to have unit distance to the receiver), chosen the similarity threshold angles yield the following similarity threshold for the physical source distance: $\epsilon_s \in \{0.09\,\mathrm{m}, 0.26\,\mathrm{m}, 0.52\,\mathrm{m}\}$. Figure 3.5 shows the training set embeddings and the testing set embeddings for the CAMIL testing set and the VAST testing-set-1. Firstly, it can be observed that the proposed WSCE method learns a manifold from the binaural cues that can reflect the sound source location without any azimuth/elevation annotations. This manifold has a clear structure and a similar structure is obtained in both the CAMIL dataset (with reverberant speech) and the VAST dataset (with varying reverberation). Secondly, when using smaller similarity threshold angles (i.e. $\epsilon_u = 5°$), the structure of the manifold tends to become irregular and folded, and when using larger threshold angles (i.e. $\epsilon_u = 15°$ and $\epsilon_u = 30°$), the structure of the manifold tends to become smooth and unfolded. Elaborating the intuition introduced in Section 3.4.3, this

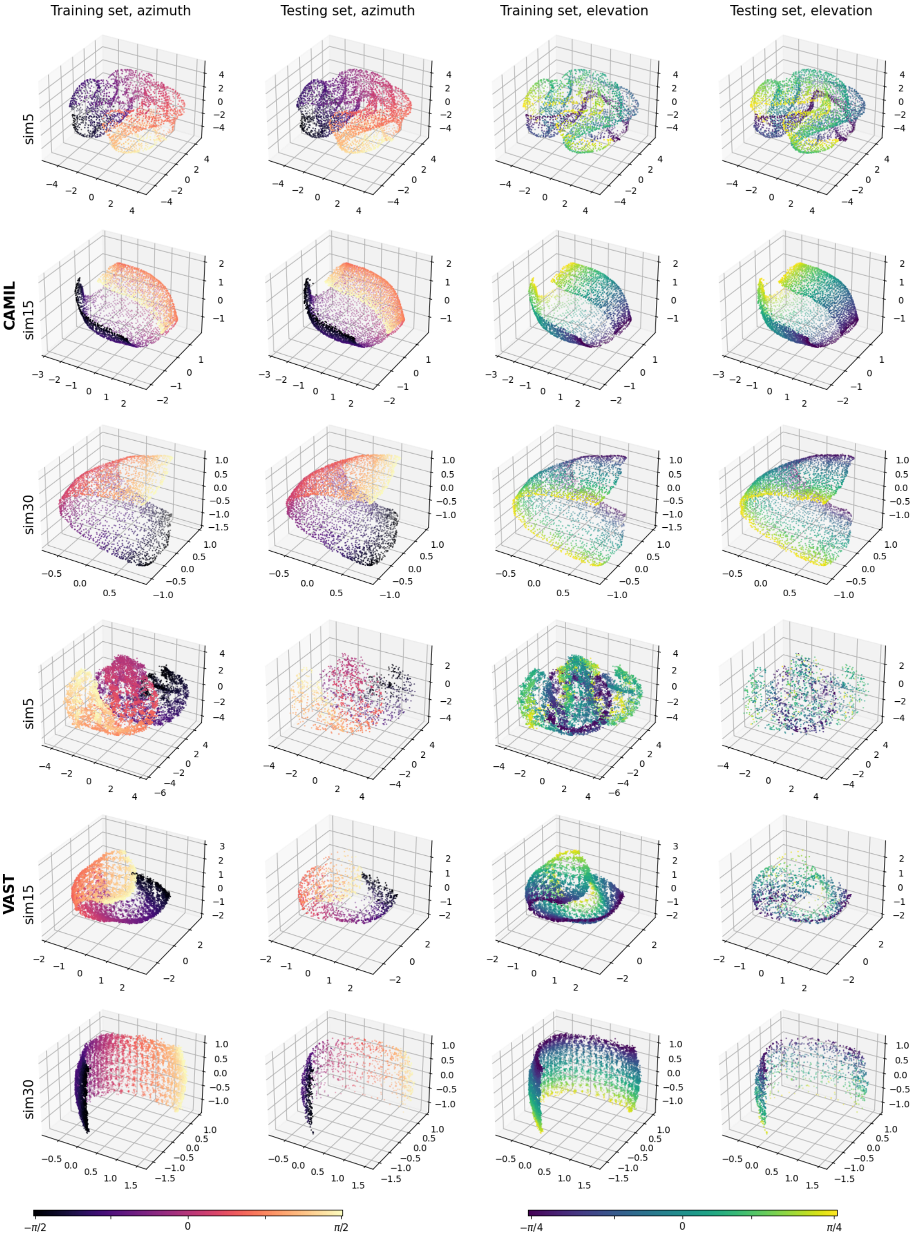


Figure 3.5: Visualisations of the WSCE embeddings. Column 1 and 2 are azimuth training and testing embeddings. Column 3 and 4 are elevation training and testing embeddings.

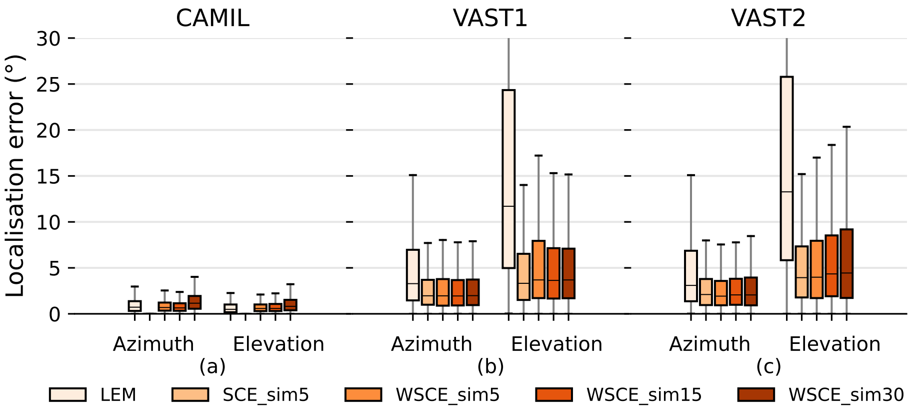


Figure 3.6: Testing set localisation performance for the LEM, the SCE, and the WSCE methods.

may be due to the fact that when the similarity threshold angle is small, the contrastive loss has a small range of action on penalising mislocated dissimilar pairs, resulting in many dissimilar pairs not being subject to repulsive forces, and instead, similar pairs are attracted and clustered in local areas. When a large similarity threshold angle is used, each embedding is subject to both attractive and repulsive forces from a large number of other embeddings, thus maintaining an overall uniformly equilibrium state in the global perspective.

In addition to the above mentioned qualitative experiments, we also conduct quantitative experiments to use the WSCE for source localisation and compare the results to the LEM embeddings and the SCE_sim5. The localisation results are shown in Figure 3.6. In the fixed acoustic condition with the CAMIL dataset, the SCE_sim5 still performs the best but it trains separate embeddings for azimuth and elevation. In contrast, both the proposed WSCE and the LEM embedding train one embedding for both azimuth and elevation estimation and shown a strong source localisation ability as well. In azimuth estimation, the WSCE_sim15 performs slightly better than the WSCE_sim5, then followed by LEM and WSCE_sim30 (achieving median errors equal to 0.64°, 0.69°, 0.72°, and 1.16°, respectively). In elevation estimation, LEM exhibits a median error equal to 0.49° and performs slightly better than the WSCE_sim15 and WSCE_sim5, which have the same median error equal to 0.58°. WSCE_sim30 performs worst in elevation estimation and achieves a median error equal to 0.82°. Nevertheless, the WSCE shows a comparable localisation ability to the LEM in the fixed acoustic condition.

In varying acoustic conditions with the VAST dataset, instead, the WSCE

H0 hypothesis	**VAST 1 AZ**	**VAST 1 EL**	**VAST 2 AZ**	**VAST 2 EL**
There is no difference in localisation error between SCE_sim5 and WSCE_sim5	0.2619	$\mathbf{8 \times 10^{-9}}$	0.0896	0.7994
There is no difference in localisation error between SCE_sim5 and WSCE_sim15	0.681	0.094	0.9992	0.9532
There is no difference in localisation error between SCE_sim5 and WSCE_sim30	0.253	**0.0158**	0.7094	0.6011

Table 3.3: Wilcoxon signed-rank tests and p-values. Numbers in bold indicates the corresponding test reject H0 for confidence interval equal to 0.05 and in favour of the alternative hypothesis.

	SCE_sim5 vs. WSCE_sim5	**SCE_sim5 vs. WSCE_sim15**	**SCE_sim5 vs. WSCE_sim30**
VAST 1 AZ	Fail to reject H0 that there is no difference in localisation errors	Fail to reject H0 that there is no difference in localisation errors	Fail to reject H0 that there is no difference in localisation errors
VAST 1 EL	Reject H0 in favour of alternative that there is difference in localisation errors	Fail to reject H0 that there is no difference in localisation errors	Reject H0 in favour of alternative that there is difference in localisation errors
VAST 2 AZ	Fail to reject H0 that there is no difference in localisation errors	Fail to reject H0 that there is no difference in localisation errors	Fail to reject H0 that there is no difference in localisation errors
VAST 2 EL	Fail to reject H0 that there is no difference in localisation errors	Fail to reject H0 that there is no difference in localisation errors	Fail to reject H0 that there is no difference in localisation errors

Table 3.4: The conclusions to Wilcoxon signed-rank tests.

shows a much lower localisation error than the LEM embeddings and it is even approaching the SCE_sim5 performance. Firstly, with the VAST testing-set-1, the WSCE_sim5, WSCE_sim15, and WSCE_sim30 perform equally well (azimuth median errors equal to 1.96°, 1.94°, and 1.98°, respectively, and elevation median errors equal to 3.69°, 3.64°, and 3.69°, respectively), and the SCE_sim5 has slightly better elevation estimation than either WSCE method (achieving azimuth and elevation median errors equal to 1.96° and 3.32°, respectively). For the VAST testing-set-2, similarly, the WSCE_sim5, WSCE_sim15, WSCE_sim30, and SCE_sim5 perform somewhat equally well (achieving azimuth median errors equal to 1.93°, 2.1°, 2.1°, and 2.1°, respectively, and elevation median errors equal to 3.99°, 4.34°, 4.44°, and 3.94°, respectively). We also conduct the Wilcoxon signed-rank statistical tests to verify if there is significant difference in localisation errors between SCE_sim5 and WSCE_sim5, SCE_sim5 and WSCE_sim15, and SCE_sim5 and WSCE_sim30. The H0 hypotheses and p-values are shown in Table 3.3, which are compared with confidence interval equal to 0.05, and the test conclusions are drawn in Table 3.4. The results indicate that there is almost no difference in localisation errors between SCE_sim5 and the WSCE models except for VAST testing-set-1 which differs between SCE_sim5 and WSCE_sim5, and SCE_sim5 and WSCE_sim30 in terms of elevation prediction. Although the unidimensional SCE_sim5 and the WSCE with a small similarity threshold show narrower interquartile range than other methods, we do suggest to use a similarity threshold angle $\epsilon_u = 15$ °for WSCE to achieve both good visualisation and localisation performance.

Secondly, the WSCE largely outperforms the LEM embeddings in varying acoustic conditions where LEM only obtains an azimuth median error of 3.28°and an elevation median error of 11.7° for VAST testing-set-1, and an azimuth median error of 3.09°and an elevation median error of 13.27° for VAST testing-set-2, respectively. Also, the WSCE has a much narrower interquartile range than the LEM, which may indicate that the proposed WSCE is more robust to reverberation than the LEM embeddings.

3.6 WSCE with unseen HRIRs

To further verify the generalisation capability of the proposed WSCE, we test the WSCE with different HRIRs that are not seen during the training. To create simulated binaural recordings, we use the CIPIC dataset [2] which consists of 45 real-life measured HRTFs. There are in total 45 subjects (43 human subjects and 2 dummy head subjects), and for each subject, 1250 HRTFs are measured for each ear and from different azimuth and elevation angles. We select azimuth and elevation angles in range the [-90°, 90°] and [-45°, 45°] respectively, corresponding

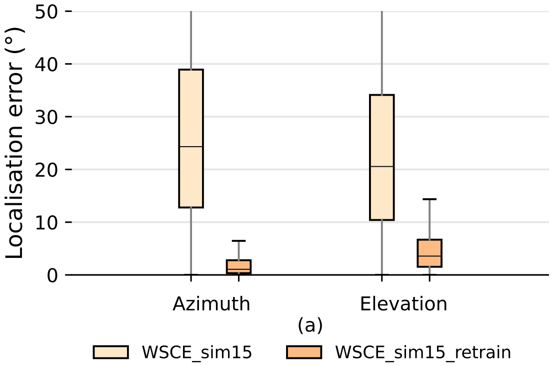


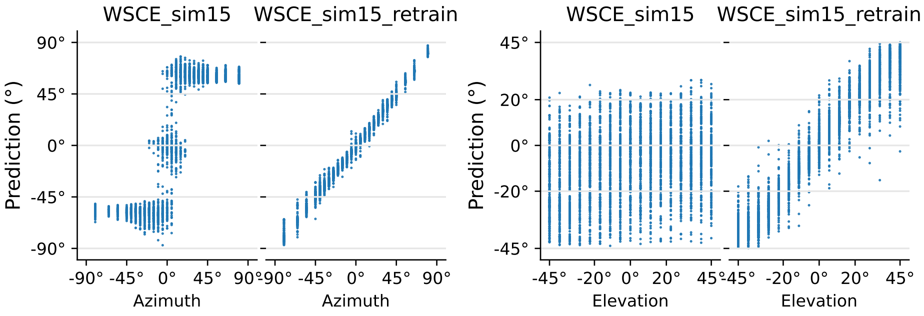
Figure 3.7: Testing set localisation performance for the WSCE_sim15 and WSCE_sim15 retrained with 33 different HRTFs other than the once used in generating test recordings.

to the other datasets mentioned in the former sections. The HRTFs are then convoluted with simulated reverberant recordings (excited by 2 s white noise). Those recordings are generated using the image method [4], in a shoebox room that has dimension 3.5 × 5 × 2.8 m, and reverberation time equal to 0.3 s.

We randomly select recordings from 10 subjects for testing, and use WSCE_sim15 for estimating their source locations. The localisation results are plotted in Figure 3.7. Since we train the WSCE_sim15 only using one HRTF, the model could not generalise well to recordings made with unseen HRTFs. Therefore, we observe a dramatic performance degradation, in which the median errors of azimuth and elevation localisation are 24.3° and 20.1° respectively.

To overcome the performance degradation, we propose two approaches:

1. Personalised training (user-dependent): this approach is especially interesting for hearing-aid applications since the hearing-aid is designed for a specific user, and it is not shared with different people. Therefore, the HRTF of the designated user can be measured and be used in the model training or fine-tuning process to create a user-dependent model.
2. Increase training data variety (user-independent): another solution consists in using more HRTFs to create the training data for training the WSCE. Then, the trained model can generalise to people with different HRTF than the ones in training data. A rule of thumb is that the higher the variety of the training data (with annotation), the better the generalisation capability of the model.



(a) (b)

Figure 3.8: True locations of sources in the testing set which uses unseen HRTFs plotted versus the location predictions.

We adopt the second approach to retrain the WSCE_sim15 and use the rest of the HRTFs from the CIPIC dataset, which are different from the data used in the testing (i.e., user-independent). This results in 33 HRTFs that are used for training, 2 for validation and 10 for testing. We also simulate random shoebox rooms that have reverberation time between 0.1 s to 0.4 s. The localisation error of the retrained model is shown in Figure 3.7 with name “WSCE_sim15_retrain”. The azimuth and elevation median errors of the retrained model have been largely reduced from 24.3° to 1.1° and 20.1° to 3.6° respectively, showing the effectiveness of this approach.

The predictions are illustrated by the scatter plot in Figure 3.8. Each point corresponds to a sample of the CIPIC testing set with its true azimuth or elevation angle on the x-axis and its source position predicted using WSCE_sim15 on the y-axis. For the original WSCE_sim15, the prediction of the source azimuth in the range [-20°, 20°] is almost distributed over the whole range [-80°, 80°], while the prediction of the source elevation angle is almost random. This suggests that the original WSCE_sim15 trained with only one HRTF cannot be generalised to unseen HRTFs. In contrast, the generalisation to unseen HRTFs is much better after retraining WSCE_sim15 with 33 real-life HRTFs. Even though there are some deviations (around 10°) in the predictions, they show a clear agreement with the real sound source locations.

However, a limitation of our simulations is that we use synthetic rooms with slightly different acoustic properties than real-life rooms. In addition, we always excite the sound source with white noise, which has a broadband spectrum, while real-life sounds may not have the same characteristics. We propose to increase the variety of training data covering real-life conditions, using more

HRTFs recorded at finer azimuth/elevation angles, and using Room Impulse Responses (RIRs) from more complex rooms, which we believe will further improve the generalisation capability of the proposed WSCE model.

3.7 Conclusions

We proposed a DNN framework for supervised dimensionality reduction of binaural cue measurements, followed by a nearest-neighbour regression method for source localisation. Our manifold-learning-based method has better binaural sound source localisation performance than the baseline manifold learning method in both know and unknown reverberant conditions and in a small training set condition. In comparison with a feed-forward learning method, our proposed method not only provides a better visualisation ability, but also achieves a similar or better performance in binaural sound source localisation. Moreover, our proposed method can capture a smooth manifold structure for low data density regions and outperforms the baseline manifold learning method and the feed-forward method in case of a small amount of training data.

In addition to the supervised dimensionality reduction method, we also proposed a weakly supervised embedding, i.e. WSCE, that only requires implicit latent space proximity labels for training. This WSCE can simultaneously estimate the azimuth and elevation of the sound source, and is also robust to unknown reverberation. Quantitative experimental results demonstrate that this WSCE has almost similar localisation performance as the supervised method, and it performs much better than the traditional unsupervised embedding in varying acoustic conditions.

To further increase the generalisation capability of the proposed model, we hope to learn the SCE and WSCE embeddings with big variety of training data covering more real-life conditions, such as using more HRTFs recorded at finer azimuth/elevation angles and using RIRs from more complex rooms. In addition, we also aim to further investigate how to apply the proposed SCE and WSCE in data synthesis. When combining these methods with a generative model, we speculate that the embeddings can be used to synthesise binaural features or even audio waveforms to aid data-driven binaural source localisation models.

The proposed methods have potential in a number of practical applications where the location of a sound source is to be identified, for example in signal processing front-ends for hearing aids and intelligent interactive dialogue systems. As such systems often have limited computational resources, reducing the model complexity and the number of model parameters is therefore a relevant direction for future research. Possible approaches to achieve this include model pruning (i.e. removing the DNN neurons that are associated with very small weights), model information distillation [74] and model parameter quantisation. Note that the proposed methods start from binaural signal features, which implies that binaural rather than bilateral hearing aids are required when using these methods for sound source localisation in hearing aid systems.

Chapter 4

End-to-end transfer learning for speaker-independent cross-language and cross-corpus speech emotion recognition

Abstract

Data-driven models achieve successful results in Speech Emotion Recognition (SER). However, these models, which are often based on general acoustic features or end-to-end approaches, show poor performance when the testing set has a different language than the training set (i.e. in a cross-language setting) or when these sets are taken from different datasets (i.e. in a cross-corpus setting). To alleviate these problems, this paper presents an end-to-end Deep Neural Network (DNN) model based on transfer learning for cross-language and cross-corpus SER. We use the wav2vec 2.0 pre-trained model to transform audio time-domain waveforms from different languages, different speakers and different recording conditions into a feature space shared by multiple languages, thereby reducing the language variabilities in the speech embeddings. Next, we propose a new Deep-Within-Class Covariance Normalisation (Deep-WCCN) layer that can be inserted into the DNN model and aims to reduce other variabilities including speaker variability, channel variability and so on. The entire model is fine-tuned in an end-to-end manner on a combined loss and is validated on datasets from three languages (i.e. English, German, Chinese). Experimental results show that our proposed method not only outperforms the baseline model that is based on common acoustic feature sets for SER in the within-language setting, but also significantly outperforms the baseline model for the cross-language setting. In addition, we also experimentally validate the effectiveness of Deep-WCCN, which can further improve the model performance. Next, we show that the proposed transfer learning method has good data efficiency when merging target language data into the fine-tuning process. The model speaker-independent SER performance increases with up to 15.6% when only 160 s of target language data is used. Finally, when comparing with the results in recent literatures which use the same testing datasets, our proposed model shows significantly better performance than other state-of-the-art models in cross-language SER.

Keywords— Cross-language, Cross-corpus, Speech Emotion Recognition, Transfer learning, Deep within-class covariance normalisation

4.1 Introduction

The emotions we daily experience determine for a large part our mental flourishing and suffering. Happiness, or psychological well-being, relies greatly on how people experience positive and negative emotions in their lives [88]. Modern Human Computer Interaction (HCI) systems use image/video, speech, and physiological signals to determine people's emotion [132]. Vocal expression is a direct and affectionate way of expressing emotions that has the advantage of being more accessible than image/video and physiological signals, for which a careful camera positioning or a well-worn wearable device is needed. As a result, Speech Emotion Recognition (SER) is widely used in many applications, such as, an in-car board system that can provide aids or resolve errors in the communication according to the driver's emotion [136], a diagnostic tool that uses the user's speech emotion to provide diagnostic information to the physiotherapist [56], and an assistant robot that can provide emotional communication [31].

SER using data-driven models has been successful in recognising emotions [137, 50, 166, 174, 173, 155, 148]. However, many data-driven models rely strongly on the mechanism underlying the generation of the data (i.e. the independent and identically distributed (i.i.d) assumption), and may hence fail when the testing data is taken from a different distribution than the training data. Alike other speech-related tasks, SER becomes highly challenging in cross-language, cross-corpus, and cross-speaker scenarios [19, 78]. To facilitate these challenges, traditional SER approaches use low-level-descriptor features that have been selected and grouped in the Geneva Minimalistic Acoustic Parameter Set (GeMAPS) feature set, including various acoustic features such as frequency-related (e.g. pitch, formant), energy-related (e.g. loudness) and spectral (e.g. spectral slope) features. In [166], the authors use low-level-descriptor features in cross-language SER, showing that people speaking different languages may express emotion in a similar way at the low-level signal level. To extract features that contain higher-level information, the i-vector proposed in [41] for speaker verification, has been further extended to SER where the "emotion i-vector" shows robustness in an English-German cross-language SER setting [44].

In recent work, the complicated feature design process is taken over by a unified Deep Neural Network (DNN) model that enables end-to-end learning from raw data or shallow features [174, 173, 155, 148]. Robust features are then learned from data to minimise an overall loss towards emotion recognition. Many works achieve good SER performance on a single dataset [174, 173, 155, 148], but the model performance degrades dramatically in cross-language and cross-corpus scenarios [78]. To alleviate this problem, several transfer learning techniques have been applied. In [112, 91], the authors pre-train a neural network with emotional datasets from one or two languages, then fine-tune the model with a small portion of the target language data, which shows a performance improvement compared to when the model is trained with one language and tested on another languages (i.e. in the cross-language setting). In [63], a similar idea to pre-train a DNN model using multiple emotional datasets is proposed, and the authors propose to fine-tune only a task-specific parallel residual adapter which achieves increased SER performance on each target task while keeping the amount of parameters to be updated low.

Other than to adapt the pre-trained model to the target language, one can also learn language-invariant or corpus-invariant features. This can be done by transforming the input space to a common subspace among each corpus. In [141], Song firstly proposed to transform the source and target input space to a common subspace where the input source-target neighbourhood relations are preserved. Zhang and Song later updated this framework to include sparsity and add discriminative power to the learning of the subspace [172]. To enable non-linear modelling with the DNN, the transformation to the common subspace is merged into an encoder neural network with a discriminator, and the learning is adversarial between the encoder and the discriminator where the encoder transforms source and target inputs to embeddings that will fool the discriminator who then needs to find the true related inputs (i.e. from the source dataset or the target dataset) [90, 99]. However, these methods require the definition of the "source" and "target" language beforehand, and their application is limited to those two languages. In [64], instead of using a binary discriminator, a Wasserstein distance [9] is used to measure the distances between target and source embeddings, and the method minimises the distances between them.

We propose an end-to-end transfer learning framework to facilitate cross-language SER. This method lies in-between the transfer learning method and the common subspace method. First, the proposed model uses a wav2vec 2.0 feature extractor which has been trained in a self-supervised manner on about 56,000 hours of raw speech waveforms from 53 languages. Since the pre-trained wav2vec 2.0 feature extractor learns contextual structures across various speech utterances, it may capture common speech factors among different languages, hence transforming the speech waveform inputs to a common speech subspace which is shared across languages [38], and in which the corresponding speech embeddings are obtained. Next, a statistical pooling layer is applied that pools a sequence of the embeddings into one feature vector per utterance. The utterance-level feature vectors are then reduced in dimensionality, and a Deep-Within-Class Covariance Normalisation (Deep-WCCN) operation is applied to compensate for other variabilities (i.e. other than language variabilities). Finally, the compensated feature is used to predict emotion class with a simple linear classifier (constructed by a dense neural network layer). The model is fine-tuned on three emotional speech datasets with English, German, and Chinese languages using a summation of the supervised cross-entropy loss and the original wav2vec 2.0 loss. We evaluate its leave-one-language-out performance on unseen speakers, and the experimental results show that the proposed framework largely increases the cross-language SER performance compared to two well-known feature sets extracted using openSMILE for SER. It also largely outperforms many recent approaches to both within-language and cross-language SER. Finally, additional experiments on the effectiveness of the Deep-WCCN operation, and the effects of merging small amounts of target language speech data into the training set will be presented.

The paper is structured as follows. First, Section 4.2 provides a brief overview about the most recent studies related to our work. Then we introduce the proposed model structure in Section 4.3, and propose some modifications to a similar approach in Deep-WCCN [48]. In Section 4.4, we describe the experiment datasets and the experimental settings. After that, we will present the simulation results and discuss these in Section 4.5. Finally, Section 4.6 presents the conclusions and suggestions for future work.

4.2 Related work

4.2.1 SER features

Traditional speech information retrieval systems use hand-crafted features such as the Mel-Frequency Cepstral Coefficients (MFCC) features and the Linear Predictive Coding (LPC) features [69], and those features are designed to largely reduce the data dimensionality and to encode temporal and spectral structures per time frame in a speech recording. A group of low-level descriptor features such as pitch, formant frequencies and bandwidths, loudness, and spectral slope have been selected and compared in [137] for SER. Reynolds and Rose later proposed to use Gaussian Mixture Models (GMMs) to model the speakers' MFCC distribution [122]. Each individual Gaussian component of a GMM encodes a general acoustic class (e.g., vowels, nasals, or fricatives), and the spectral shape of the acoustic class can be characterised by the GMM component mean vector and the GMM component covariance matrix. The concatenation of GMM component mean vectors is a representation that contains abstract information such as speaker identity. This high-dimensional representation (referred to as the supervector in [41]) is then used to extract the i-vector which encodes the speaker and channel information in a total variability subspace [41]. However, these hand-crafted features are not designed for SER specifically, and may contain dominant misleading factors (e.g., speaker identities, language factors) which leads to poor SER performance. This problem is tackled in [44] where a two-step approach is proposed. The approach first compensates for the speaker variability in the supvervector space, then an "emotion i-vector" that encodes the variabilities of each emotion supervector centred around a maximum likelihood emotion supervector is extracted.

4.2.2 End-to-end learning

With the growing popularity of deep learning, the traditional learning pipeline (i.e. pre-processing, feature extraction, modelling, inferencing) is replaced by a single DNN which learns the features and performs the modelling in a data-driven manner. In this case, raw data or shallow features are directly fed into the learning process, and the DNN model predicts the target (e.g. class labels, parameter estimates) at its output. This is referred as end-to-end learning. Due to the data-driven nature of end-to-end learning, very little domain expertise is used for learning, and brute-force feature validation is avoided, thereby significantly reducing labour costs. In addition, compared to the hand-crafted features that unavoidably leads to information loss during the extraction process, the end-to-end framework fuses feature learning and selection into one process so that dedicated features for the target problem can be learned, thus contributing to the modelling performance.

In the context of SER, Trigeorgis et al. first proposed an end-to-end DNN that consists of Convolutional Neural Network (CNN) layers for feature extraction from raw audio waveforms and Long Short-term Memory (LSTM) layers to model the temporal information in the feature space [153]. In [130], the authors replace the CNN with a Time-Delay Neural Network (TDNN) to create a larger receptive field. A similar idea has been proposed in [148] where a dilated CNN having a receptive field as large as the input sequence is used for both feature extraction and temporal modelling. However, generalisability remains a problem in supervised end-to-end learning since annotated SER datasets are general of small scale and recorded in specific environments, hence a model learned from these datasets may perform poorly when the test conditions are different from the training conditions, as well as in a cross-language setting.

4.2.3 Self-supervised learning and transfer learning

Self-supervised learning provides an end-to-end learning pathway for constructing speech features from large quantities of data without annotations. Self-supervised learning methods first transform time-domain raw audio waveforms to an embedding space (i.e. feature space), and then aim to predict randomly masked embeddings from past embeddings or adjacent embeddings [158, 135, 12, 38]. This learning scheme forces the DNN to learn intrinsic contextual structure from the data rather than modelling noise or irrelevant factors, because on a large scale, noise will not contain useful information to predict neighbouring embeddings, thus will be suppressed.

The speech features obtained from self-supervised learning can either be used directly in relevant tasks, or can be fine-tuned in a transfer learning framework. In the latter case, a few extra layers, which map the features into predictions, are added to the trained self-supervised feature extractor. Then, the predictions are evaluated with a supervised loss, and finally the entire set of DNN model parameters is updated with a supervised training dataset. An example in SER is [116] where the authors use a concatenation of several layer outputs of the wav2vec 2.0 pre-trained model for monolingual SER. However, the authors fine-tune the model on the Automatic Speech Recognition (ASR) task using a dataset that only contains neutral speech which may not be as efficient as fine-tuning the model directly on an emotive speech dataset and on a SER task.

4.3 Proposed transfer learning method with Deep-WCCN for cross-language SER

The proposed method is based on the wav2vec 2.0 self-supervised learning model [12]. This model has been pre-trained and used in cross-language speech recognition and has been shown to deliver speech representations that are shared across languages [38]. We first give the details of the wav2vec 2.0 model in Section 4.3.1, then we will present how to use the wav2vec 2.0 model in the front-stage of an end-to-end SER system, together with a modified Deep-WCCN layer inspired by similar work in [48] (in Section 4.3.2) and finally how to fine-tune the model under a supervised scheme in Section 4.3.3.

4.3.1 Self-supervised pre-training of wav2vec 2.0

The wav2vec 2.0 model learns contextual representations from speech waveforms. The model consists of a convolutional feature extractor f, a transformer-based sequence model e and a vector quantiser module [38]. The feature extractor f first transforms the raw speech waveform $\mathcal{X}$ into a sequence of d_z-dimensional latent speech representations $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, ..., \boldsymbol{z}_T]$ for T time-stamps. Then the latent speech representations on one hand are fed into the sequence modeller to build contextual representations $\boldsymbol{c}_1, \boldsymbol{c}_2, ..., \boldsymbol{c}_T$, and on the other hand are fed to the vector quantiser that contains G codebooks, and for each codebook, there are V entries. The vector quantiser discretises the latent representations and linearly transforms them into a sequence $\boldsymbol{q}_1, \boldsymbol{q}_2, ..., \boldsymbol{q}_T$. The feature extractor f, the sequence model e and the vector quantiser module are trainable and their parameters are optimised through back-propagation with two loss functions:

1. *The contrastive loss:* For a $\boldsymbol{c}_t$ obtained from masked representations centered at time-stamp t, the sequence model needs to identify the true quantised representation $\boldsymbol{q}_t$ from $K + 1$ candidate quantised representations $\tilde{\boldsymbol{q}} \in \boldsymbol{Q}_t$ which include K distractors uniformly sampled from other masked time-stamps for the same sequence. The loss is defined as:
$$\mathcal{L}_m = -\log \frac{\exp(\text{sim}(\boldsymbol{c}_t, \boldsymbol{q}_t)/\kappa)}{\sum_{\tilde{\boldsymbol{q}} \in \boldsymbol{Q}_t} \exp(\text{sim}(\boldsymbol{c}_t, \tilde{\boldsymbol{q}}_t)/\kappa)} \tag{4.1}$$
where $\text{sim}(\boldsymbol{a}, \boldsymbol{b})$ is the cosine similarity between $\boldsymbol{a}$ and $\boldsymbol{b}$, and κ is the temperature parameter that controls the kurtosis of the distribution.

2. *The diversity loss:* To increase the use of the codebooks, the diversity loss encourages the equal usage among the entries in every codebook by maximising the entropy of the averaged softmax distribution over codebook entries for each codebook $\bar{p}_g$ across a batch of utterances:
$$\mathcal{L}_d = \frac{1}{GV} \sum_{g=1}^{G} -H(\bar{p}_g) = \frac{1}{GV} \sum_{g=1}^{G} \sum_{v=1}^{V} \bar{p}_{g,v} \log(\bar{p}_{g,v}) \tag{4.2}$$

The total loss is a weighted summation of the two losses with a hyper-parameter α,
$$\mathcal{L}_{ssl} = \mathcal{L}_m + \alpha \mathcal{L}_d \tag{4.3}$$

To accommodate for cross-language speech variations, we use a pre-trained wav2vec 2.0 model that has been trained on 56,000 hours of speech in 53 different languages. This model is denoted as XLSR-53 and the details can be found in [38].

4.3.2 Deep-WCCN

The Within-Class Covariance Normalisation (WCCN) was introduced in [70], and used in speaker recognition and verification applications [70, 41]. The WCCN approach aims to learn a feature map that minimises upper bounds on the false positive and false negative rates in a linear classifier. First, let $\boldsymbol{W} =$

$[\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_N]$ denote a set of N d-dimensional feature vectors belonging to C different classes $c \in \{1, \ldots, C\}$. In [41], $\boldsymbol{W}$ is a set of i-vectors and C is the number of different speakers. Differently in this paper, we define $\boldsymbol{W}$ to represent a set of intermediate features of the training set extracted from the front-end DNN feature extractor, and C is the total number of emotion classes.

Next, we define the expected within-class covariance matrix as:

$$\boldsymbol{S}_w = \frac{1}{C}\sum_{c=1}^{C}\frac{1}{N_c}\sum_{i=1}^{N_c}(\boldsymbol{w}_i^c - \bar{\boldsymbol{w}}^{\boldsymbol{c}})(\boldsymbol{w}_i^c - \bar{\boldsymbol{w}}^{\boldsymbol{c}})^T \tag{4.4}$$

where $\bar{\boldsymbol{w}}^{\boldsymbol{c}} = \frac{1}{N_c}\sum_{i=1}^{N_c}\boldsymbol{w}_i^c$ is the mean feature vector of class c, and $\boldsymbol{w}_i^c$ are the examples belonging to this class. N_c is the total number of examples belonging to class c in the training set.

Then the optimal feature map is given as:

$$\Phi(\boldsymbol{w}) = \boldsymbol{A}^T\boldsymbol{w} \tag{4.5}$$

where $\boldsymbol{A}$ is the Cholesky factors of $\boldsymbol{S}_w^{-1} = \boldsymbol{A}\boldsymbol{A}^T$, and $\boldsymbol{w}$ represents an arbitrary feature vector.

The conventional WCCN pools the necessary statistics from the feature vectors of the entire training set, which is not compatible with mini-batch training in DNNs. In [48], the authors propose to estimate $\boldsymbol{S}_w$ with mini-batches, and to maintain a moving average $\hat{\boldsymbol{A}}$ of the corresponding mini-batch projection matrix. We believe that this moving average operation proposed in [48], which is equivalent to a weighted average of new and old batches, where the old batches are subject to exponentially decaying weights, is not the best design choice for random mini-batch Stochastic Gradient Descent (SGD) training in DNNs. Therefore, we first define the class covariance matrix $\hat{\boldsymbol{S}}_{w,c}$, which is estimated on mini-batch, and also $\hat{\boldsymbol{S}}_w$, which is the average of $\hat{\boldsymbol{S}}_{w,c}$ across the classes. Then we propose to maintain a cumulative average $\bar{\boldsymbol{S}}_w$ for $\hat{\boldsymbol{S}}_w$,

$$\bar{\boldsymbol{S}}_w = \frac{N_{tot}}{N_{tot}+1}\bar{\boldsymbol{S}}_w + \frac{1}{N_{tot}+1}\hat{\boldsymbol{S}}_w \tag{4.6}$$

where N_{tot} is an accumulator that counts the total number of mini-batches during training. Next, we add a spectral smoothing term to the within-class covariance estimation similar to [70]:

$$\boldsymbol{S}'_w = (1-\beta)\bar{\boldsymbol{S}}_w + \beta I \tag{4.7}$$

where the hyper-parameter $\beta \in$[0, 1] controls the smoothness of the estimated within-class covariance matrix. Finally, $\boldsymbol{S}'_w$ is used to calculate $\boldsymbol{A}$ as mentioned before.

This mini-batch based WCCN is considered as a special DNN layer that has no trainable parameter and only updates $\boldsymbol{A}$ during training, denoted as Deep-WCCN. The last update of $\boldsymbol{A}$ during training is stored and used in testing. The output of Deep-WCCN is the result of the linear transformation in (4.5), which can be fed into the subsequent layers of the DNN model. We implement the Deep-WCCN using Pytorch[1], which is an automatic differentiation toolbox for deep learning. It should be noted that in order to avoid Pytorch to automatically generate gradients of $\boldsymbol{w}$ when calculating $\boldsymbol{A}$, a "detached" (from the computational graph) copy of $\boldsymbol{w}$ is used in the calculation during training.

4.3.3 Fine-tuning the wav2vec 2.0 model for SER

One problem of the pre-trained XLSR-53 model is that the training sets mostly consist of neutral speech, and as a consequence the pre-trained model may capture inadequate emotion structures embedded in the speech data. Therefore, we propose to fine-tune the XLSR-53 model with emotional speech data. We combine the XLSR-53 pre-trained model and the Deep-WCCN with a few extra layers into a unified model that maps time-domain raw waveforms to emotion class predictions, and this unified model is fine-tuned in a supervised manner. The model overview is shown in Figure 4.1. After converting the speech raw waveform to the sequence of latent speech representations $\boldsymbol{Z}$, we calculate an utterance-level feature by pooling and concatenating the statistics of $\boldsymbol{Z}$ along the time dimension as done in [95],

$$\boldsymbol{u} = \begin{bmatrix} \text{mean}(\boldsymbol{Z}) \\ \text{std}(\boldsymbol{Z}) \end{bmatrix} \tag{4.8}$$

The resulting feature vector $\boldsymbol{u}$ is a $2d_z$-dimensional vector that proceeds to a fully-connected layer which reduces its dimensionality to a pre-defined hidden dimension $d_h = \frac{1}{4}d_z$ and which is then followed by a Rectified Linear Unit (ReLU) non-linear activation and a dropout layer. The output from the dropout layer is the intermediate feature vector that will be fed into the Deep-WCCN to reduce the within-class variances. After that, the Deep-WCCN output feature vectors are normalised to have unit norm, and linearly transformed into predictions for emotion classification.

The fine-tuning loss is a weighted summation of the wav2vec 2.0 loss $\mathcal{L}_{ssl}$ and the emotion classification log-softmax cross-entropy loss. Specifically, given one prediction $\boldsymbol{p} = [p_1, p_2, \dots, p_C]^T$ and its corresponding one-hot emotion class label $\boldsymbol{y} = [y_1, y_2, \dots, y_C]^T$ (where only the true class label is 1, and other labels

[1] https://pytorch.org/

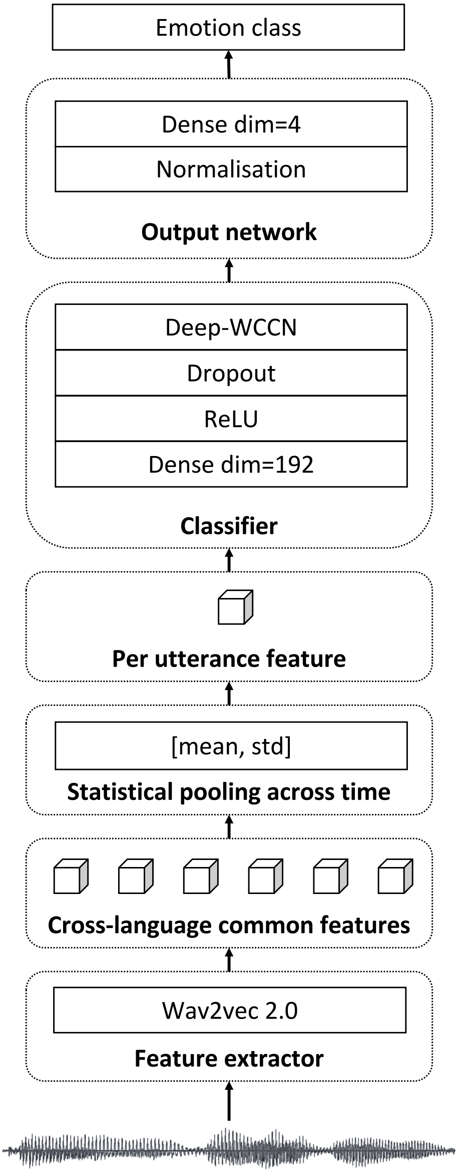


Figure 4.1: The network structure of the proposed end-to-end transfer learning model for cross-language/cross-corpus SER.

are 0) the total loss is:

$$\mathcal{L}_{tot} = -\sum_{c=1}^{C} y_c \log \frac{\exp(p_c)}{\exp(\sum_{i=1}^{C} p_c)} + \gamma \mathcal{L}_{ssl} \tag{4.9}$$

where the hyper-parameter γ controls the weight of $\mathcal{L}_{ssl}$.

4.4 Experiment set-ups

4.4.1 Datasets

To carry out simulations for cross-language SER, we choose three publicly available datasets including English, German, and Chinese emotive speech recordings. These speech recordings are performed by professional actors have a duration of a few seconds each. The datasets are summarised in Table 4.1, and described in detail below.

Emo-DB

The Emo-DB [29] dataset is a German emotive speech dataset that has been widely used in SER research. This dataset consists of 535 utterances including 7 basic emotion catalogues (anger, boredom, disgust, anxiety/fear, happiness, sadness, and neutral). The utterances are performed by 10 professional actors, with 10 pre-defined sentences. Each sentence is performed with all different emotions, and the sentence content should not deliver sentimental information.

RAVDESS

The RAVDESS [103] dataset contains recordings with 24 professional actors (12 female, 12 male), vocalising two lexically-matched statements (i.e. “Kids are talking by the door.”, “Dogs are sitting by the door.”). The speech recordings include calm, happy, sad, angry, fearful, surprise, and disgust expressions. Each expression is produced at two levels of emotional intensity (normal, strong), with an additional neutral expression, resulting in a total of 8 emotion catalogues.

	Berlin Database of Emotional Speech (Emo-DB)	**Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)**	**Emotional Speech Dataset (ESD)** train	**ESD** valid	**ESD** test
Language	German	English	Chinese	Chinese	Chinese
# female speakers	5	12	5	5	5
# male speakers	5	12	5	5	5
Reference	[29]	[103]	[175]	[175]	[175]
		Utterance duration (s)			
Average	2.54	1.72	2.47	2.41	2.49
Median	2.3	1.66	2.3	2.3	2.37
Maximum	8.88	3.41	7.07	5.6	5.66
Minimum	0.83	0.86	0.42	0.9	0.9
Total	860	1160	29634	1932	2990
		Number of utterances per selected class			
Angry	127	192	3000	200	300
Sad	62	192	3000	200	300
Neutral	79	96	3000	200	300
Happy	71	192	3000	200	300
Total	339	672	12000	800	1200

Table 4.1: Dataset information.

ESD

The ESD [175] dataset is a recent multilingual and multi-speaker emotional speech dataset designed for various speech synthesis and voice conversion tasks. The dataset consists of 350 parallel utterances spoken by 10 native English and 10 native Mandarin speakers. In this work, we only use the Mandarin utterances, which involve 5 male speakers and 5 female speakers, and are performed in 5 emotion catalogues (happy, sad, neutral, angry, and surprise).

Dataset preprocessing and partitioning

First, we only select 4 overlapping emotion catalogues (angry, happy, neutral, and sad) from the aforementioned datasets, and all recordings are re-sampled to 16 kHz. Then, we zero-pad or randomly crop the time-domain raw waveforms to have a 2 s duration. Next, we apply mean and variance normalisation across each waveform to match the requirements of the XLSR-53 pre-trained model.

Finally, we partition each dataset into training, validation, and testing subsets. Each dataset is divided into 5 groups containing different speakers, which results in 2 speakers per group for the Emo-DB dataset and the ESD dataset, and 5 speakers for the first four groups of the RAVDESS dataset and 4 speakers for the last group. To ensure there is no speaker overlap in the subsets, we use three groups for training, one group for validation, and one group for testing. This scheme results in 5 different partitionings of the datasets, and allows us to apply 5-fold cross-validation. The original speaker IDs used in validation and testing are listed in Table 4.2, and the remaining speakers are used for training. Note that, the ESD dataset has originally already partitioned into training, validation and testing sets for each speaker. Hence when creating ESD subsets, we firstly merge the original training, validation and testing sets per speaker, then partition the new subsets again by speaker IDs.

4.4.2 Baseline features

For comparison with the proposed model, we choose the baseline systems using the most widely used feature sets designed for SER. The Extended Geneva Minimalistic Acoustic Parameter Set (eGeMAPS) [51] and the Emobase [52] feature sets produce features per utterance, computed from a set of low-level-descriptors to which various statistical functions are applied. Both the eGeMAPS and the Emobase feature sets contain spectral features (e.g., pitch-related, formant-related features), energy/amplitude features (e.g., loudness), and filter-bank features (e.g., MFCC). In Emobase, statistical functions including

Division	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
			Emo-DB		
Valid	13, 14	11, 12	09, 10	03, 08	15, 16
Test	15, 16	13, 14	11, 12	09, 10	03, 08
			RAVDESS		
Valid	16, 17, 18, 19, 20	11, 12, 13, 14, 15	06, 07, 08, 09, 10	01, 02, 03, 04, 05	21, 22, 23, 24
Test	21, 22, 23, 24	16, 17, 18, 19, 20	11, 12, 13, 14, 15	06, 07, 08, 09, 10	01, 02, 03, 04, 05
			ESD		
Valid	0007, 0008	0005, 0006	0003, 0004	0001, 0002	0009, 0010
Test	0009, 0010	0007, 0008	0005, 0006	0003, 0004	0001, 0002

Table 4.2: Dataset partitioning for 5-fold cross-validation with reference to original speaker ID.

min/max, arithmetic mean, standard deviation, skewness, and kurtosis, are applied to all the low-level descriptors and their delta coefficients. In contrast, in eGeMAPS, only selected statistical functions are applied to some of the low-level descriptors, resulting a feature set containing a few extra temporal features such as the rate of loudness peaks features, and the mean length of voiced regions and unvoiced regions. The low-level descriptor selection criteria for eGeMAPS are based on the theoretical significance of the feature, the feature usage frequency in literature, and the potential of an acoustic parameter to indicate physiological changes in voice production during affective processes [51]. In term of the feature vector dimensionality, the Emobase feature vector has length 988, and the eGeMAPS feature vector has length 88.

4.4.3 Training on multiple languages

Similarly to [91, 63], we consider all combinations of merging two out of three language datasets for training and validation in order to meet the data requirements of the data-driven model and to allow the model to capture the variations across languages and corpora thus not to over-fit to one dataset.

In combination of each two datasets, we repeat the smallest dataset a few times so that for each training set, there is a similar amount of utterances from each language. This results in the following three training/validation sets:

1. **DECH**: training and validation using German and Chinese recordings, repeating the German training set 32 to 39 times depending on which fold is considered.
2. **DEEN**: training and validation using German and English recordings, repeating the German training set 2 to 3 times depending on which fold is considered.
3. **ENCH**: training and validation using Chinese and English recordings, repeating the English training set 18 to 19 times depending on which fold is considered.

4.4.4 Within-language and Cross-language settings

Within-language SER

In the within-language setting, we aim to evaluate the effectiveness of the baseline features and the wav2vec 2.0 features in SER. Therefore, for each training scheme, we test the performance of the methods on the testing set in one of the training languages, resulting in a situation where only the speaker identity is different among the training and testing sets.

Cross-language SER

To evaluate the methods in a cross-language setting, we employ a leave-one-language-out scheme, that is for each training scheme, we test the methods on the testing set of the third language which means the language, recording condition, and the speaker identity in the testing set is unseen during training.

We use “->” followed by the language abbreviations (EN, CH, DE) to indicate the dataset/language used in the testing.

4.4.5 Experimental settings

The Adagrad optimiser [47] is used to train the models, with a fixed learning rate of 3×10^{-4} and weight decay. The batch size is 14 due to memory constraints. The weighting parameter α in (4.3) is equal to 0.1 as in [38]. The weight decay parameter, the dropout rate, and the hyper-parameters β and γ in (4.7) and (4.9) are optimised using Hyperopt [18]. The optimal results are shown in Table 4.3 and correspond to the values that are used in the experiments.

We evaluate the model using two metrics, the Unweighted Accuracy (UA) and the Weighted Accuracy (WA). Specifically, UA is the average accuracy for each emotion class, which marginalises out the effect of class imbalance, and WA is the overall accuracy of the entire testing data, which indicates the overall model performance across all classes. Model selection uses early-stopping on the validation performance, and the reported testing results are averaged across cross-validation folds.

4.4.6 Baseline classifier

The baseline features are firstly tested with a neural network model that has the same structure as the classifier and the output network in the proposed model shown in Figure 4.1. However, this model essentially only contains one non-linear transformation and the Deep-WCCN operation, hence it is incapable to learn a good mapping from the baseline features to the emotion classes. In particular, when conducting a hyper-parameter search for the Deep-WCCN layer, the dropout rate, and the weight decay rate using Hyperopt for the neural network model, and then training it with the baseline features, the model SER performance is close to chance level (e.g., 25% WA for 4 emotion classes). Therefore, we instead use a Random Forest (RF) classifier, which is essentially an ensemble classifier that has good generalisation capability. This RF classifier has 100 trees with maximum tree-depth of 5.

Hyper-paramter	**DECH**	**DEEN**	**ENCH**
Learning rate	3×10^{-4}	3×10^{-4}	3×10^{-4}
Weight decay	4×10^{-4}	5×10^{-4}	5×10^{-4}
Dropout	0.45	0.05	0.3
β	0.2	0.5	0.4
γ	8×10^{-4}	1.2×10^{-3}	1.7×10^{-3}
Batch size	14	14	14

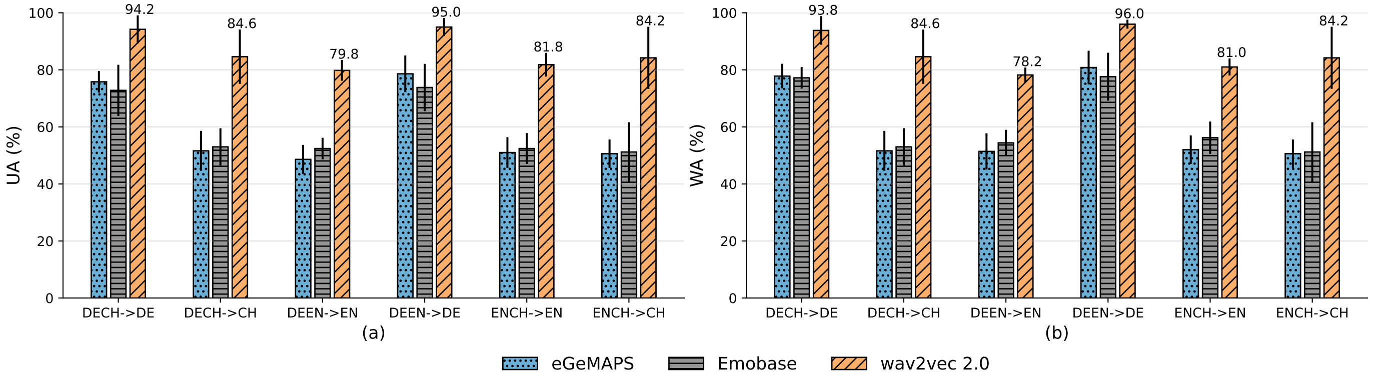
Table 4.3: Hyper-parameter configurations for the proposed method.

4.5 Results and discussion

4.5.1 Within-language performance

The results for within-language evaluation are illustrated in Figure 4.2. These results show that the baseline eGeMAPS feature and the Emobase feature with an RF classifier perform similarly in terms of UA and WA, and for some testing cases (e.g. testing on the German "->DE" dataset) the eGeMAPS performs better than the Emobase, but it is the other way around for the other testing cases (e.g. testing on the Chinese "->CH", and on the English "->EN" datasets). The largest UA and WA differences between these two methods are 4.8% and 4.2% that are obtained in the "DEEN->DE" and "ENCH->EN" scenarios, respectively.

From the same figure, comparing the proposed wav2vec 2.0 method with the baseline methods, the wav2vec 2.0 shows significant improvements in both UA and WA for within-language SER. Specifically, the wav2vec 2.0 improves UA with about 20.9% to 66.4% compared to the eGeMAPS method, and with about 28.7% to 64.5% compared to the Emobase method, and it improves WA with about 18.8% to 66.4% and 21.5% to 64.5% compared to the eGeMAPS and the Emobase, respectively. The increase in classification accuracy may indicate that the wav2vec 2.0 pre-trained feature extractor combined with the proposed



(a)

(b)

Figure 4.2: Within-language emotion classification accuracy measured by unweighted (UA) and weighted accuracy. The training and validation sets are in the languages indicated on the left hand side of ->, and the testing set is in the language indicated on the right hand side of ->. The testing set language is included in the training, but the testing speaker ID is unseen.

Deep-WCCN can extract contextual features that contribute to mixed-language SER capability and are robust to varying speaker identities.

4.5.2 Cross-language/cross-corpus performance

For the cross-language/cross-corpus experiments, the emotion classification accuracy is plotted in Figure 4.3. For comparison, the within-language results (with suffix "-WL") from Section 4.5.1 are averaged across the same testing language for every method and plotted along with the cross-language performance (with suffix "-CL").

Firstly, the wav2vec 2.0-CL performs significantly better than eGeMAPS-CL and Emobase-CL in all three testing languages, and it is followed by Emobase-CL which performs slightly better than eGeMAPS-CL in almost all the testing cases (except on English testing where eGeMAPS-CL and Emobase-CL have similar WA performance). The best UA and WA with wav2vec 2.0-CL, tested in English, Chinese, and German, are 64%, 75.4%, 94% and 61.2%, 75.4%, 94.6%, respectively. This is about 43% to 77% and 50.7% to 77% performance gain on UA and WA compared to the eGeMAPS-CL, and about 41% to 72.9% and 36.3% to 72.9% performance gain on UA and WA compared to the Emobase-CL.

Secondly, although all three methods show performance degradation when moving from within-language to cross-language/cross-corpus SER, the wav2vec 2.0 still maintains high performance. It is notable that in the "ENCH->DE" experiment, wav2vec 2.0-CL shows a very subtle degradation (i.e. about 0.6% and 0.3% degradation in UA and WA, respectively, compared to the wav2vec 2.0-WL), whereas the eGeMAPS-CL shows a 17.8% and 18.5% decrease in UA and WA, respectively, and the Emobase-CL shows a 8.5% and 8% decrease in UA and WA, respectively.

In summary, the results in cross-language/cross-corpus SER may indicate that the proposed wav2vec 2.0 with Deep-WCCN model can largely alleviate the performance degradation due to language mis-match, speaker identity mis-match, and channel mis-match that commonly occur in cross-language SER. As the proposed model shows equally satisfactory results in both UA and WA, we may also conclude that the wav2vec 2.0 with Deep-WCCN model is not over-fitting to one of the testing emotion classes.

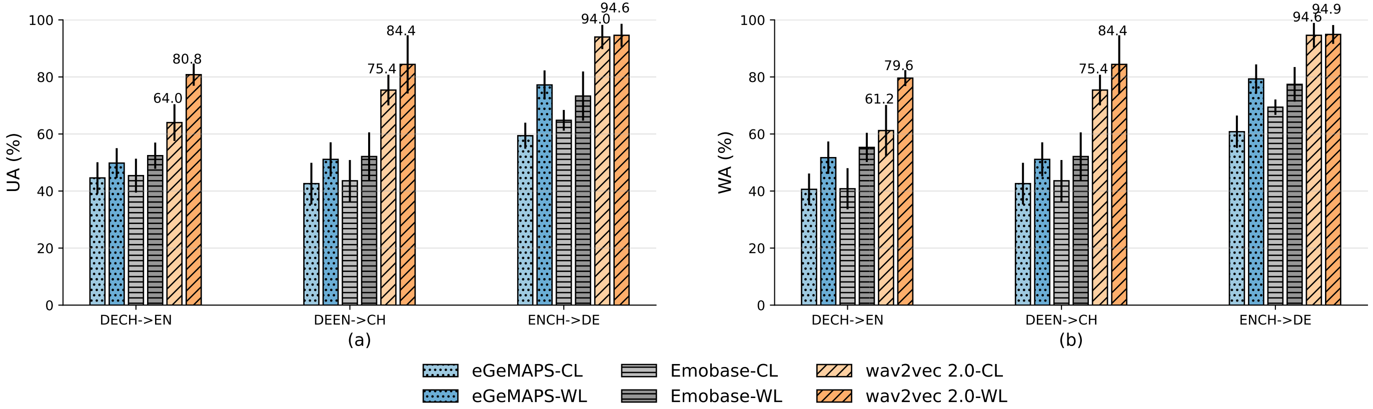
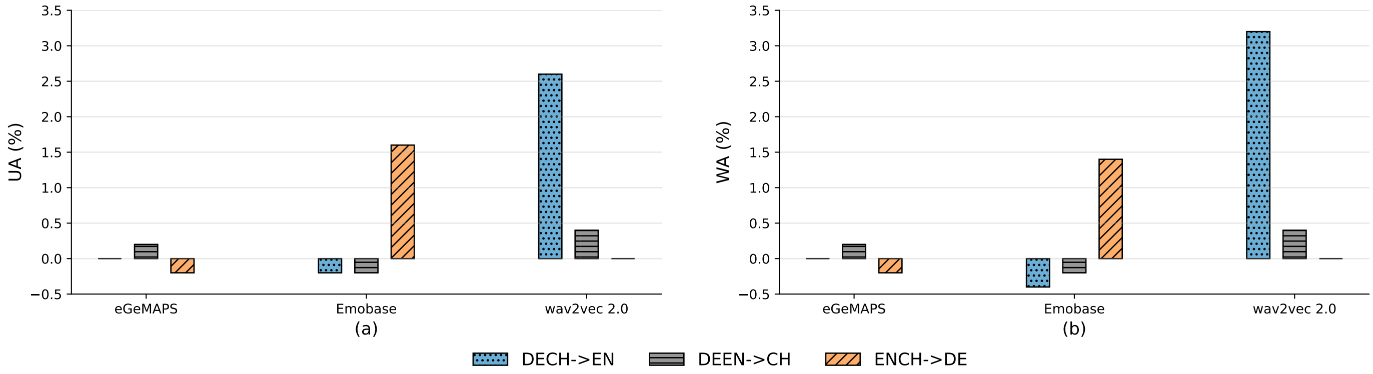


Figure 4.3: Cross-language/cross-corpus emotion classification accuracy measured by unweighted (UA) and weighted accuracy. The testing set language is not included in training, which means the testing speaker ID, the testing language, and the testing channel is unseen during training. Results are compared also with the within-language performance averaged across each testing language.



(a)

(b)

Figure 4.4: The performance gain in terms of UA and WA after applying Deep-WCCN for the proposed method (which is based on wav2vec 2.0), and the performance gain after applying conventional WCCN for eGeMAPS and Emobase methods.

4.5.3 Evaluation of the Deep-WCCN

In order to verify the effectiveness of Deep-WCCN, we compare the change in cross-language SER performance of wav2vec 2.0 with and without Deep-WCCN. The results are compared with other baseline methods. For eGeMAPS and Emobase, we calculate the change in their cross-language SER performance with and without WCCN. The increase in cross-language SER performances for all methods using either Deep-WCCN or WCCN are shown in Figure 4.4.

First, it can be clearly observed that using Deep-WCCN can improve the performance of wav2vec 2.0 in cross-language SER. The highest increase is observed in DECH->EN, where UA and WA increase by 2.6% and 3.2% respectively by applying Deep-WCCN, which is followed by increments in DEEN->CH, where both UA and WA increase by 0.4% after applying Deep-WCCN. No performance gain is observed in ENCH->DE when applying Deep-WCCN to the wav2vec 2.0 model, which might be due to the fact that the results on the German testing sets are already very good and there is limited room for improvement.

Secondly, WCCN did not significantly improve cross-language SER performance when using the eGeMAPS and the Emobase features. Specifically, by applying WCCN, the experiments on all three cross-language cases show only small performance gains, no gains, or even performance drops when applying WCCN to eGeMAPS and to Emobase. Only in the case of ENCH->DE, the Emobase method has increases of 1.6% and 1.4% in UA and WA, respectively, when applying WCCN. This is due to the fact that WCCN is designed for linear classifiers and is not significantly helpful for ensemble models like the RF. Conversely, the output layer of our proposed model can be seen as a linear classifier, which satisfies the design conditions of WCCN.

We also visualise the embeddings (which are the inputs to the output network) with and without Deep-WCCN. The embeddings are originally 192-dimensional vectors, but for visualisation purposes, we compute the first 2 principal components of the training set and validation set embeddings from each cross-validation fold using the Principal Component Analysis (PCA). The visualisations for the DECH setting are plotted in Figure 4.5.

First, the embeddings tend to form clear clusters per emotion class, which explains the effectiveness of the proposed method in SER. Second, by applying Deep-WCCN, the training set embedding clusters tend to form more compact shapes (i.e. smaller within-class variability) than without applying Deep-WCCN. This generally leads to the validation embeddings also forming compact clusters and having cleaner cluster boundaries than without Deep-WCCN.

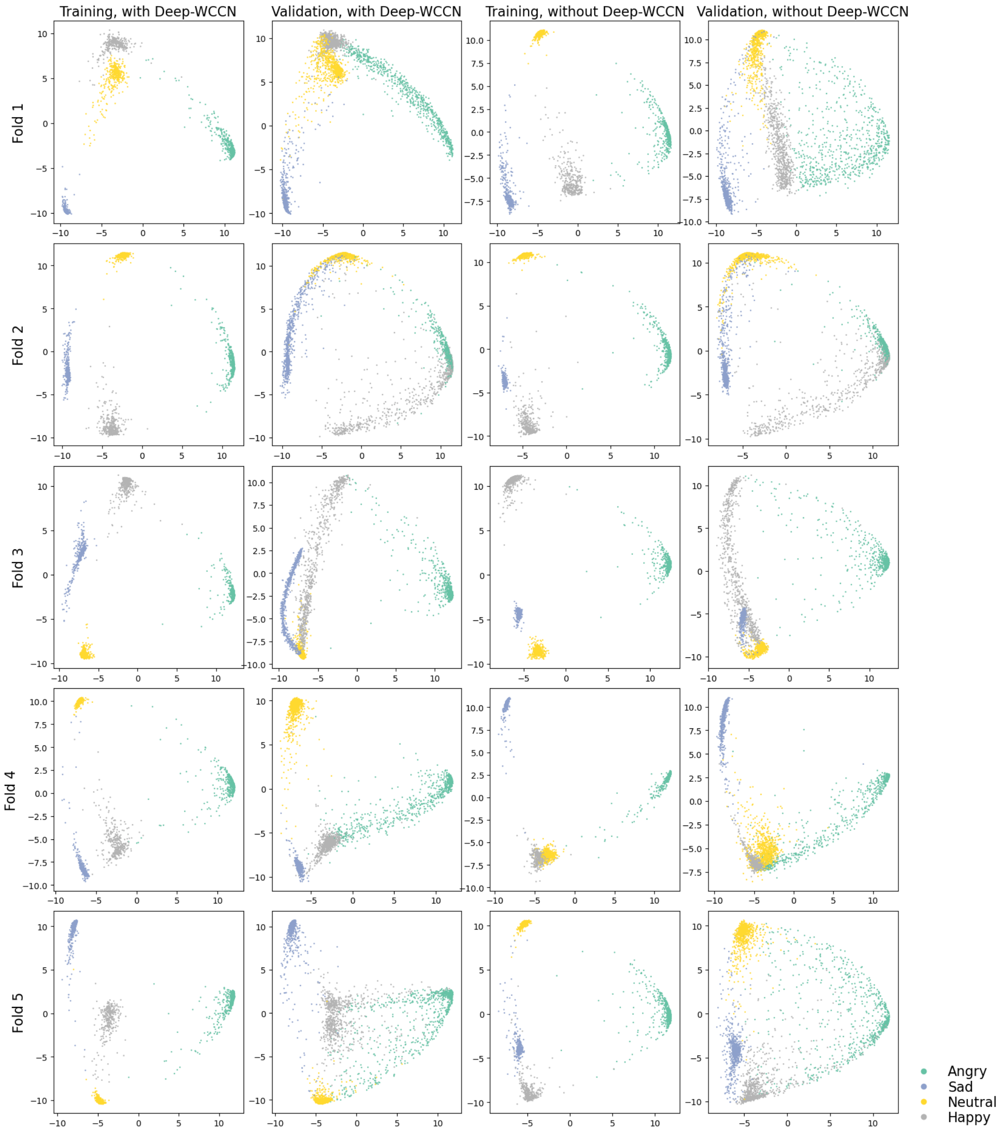


Figure 4.5: The PCA visualisation of training set and validation set embeddings when applying Deep-WCCN and not applying Deep-WCCN.

4.5.4 Influence of training set size of target language

As the proposed method is in the realm of transfer learning, it might exhibit the data efficiency property typically associated to transfer learning. That is, the transferred sub-network, which is trained on a large quantity of speech data, learns intrinsic speech structure and can easily generalise to similar tasks with a small amount of annotated target data. Therefore, in this experiment, we merge a small amount of target language data into the fine-tuning process, and evaluate the model performance when using 30, 80, and 150 two-second-target language inputs with their corresponding labels. The duration of extra target language data used in the training is hence equal to 60 s, 160 s, and 300 s in total duration, respectively. The extra target data are repeated to achieve a size that is similar to the original training data size to avoid data imbalance for different languages. The results for DECH->EN, DEEN->CH and ENCH->DE are plotted respectively in Figure 4.6, together with their performance when no extra target language data is used (i.e. the case corresponding to 0 s).

First, the three testing cases all show an increasing trend when more target language data is used in training, and their performance increases rapidly when even less than 160 s of target language data is used while it slows down when even more target language data is used. Second, the performance of DECH->EN increases the most, followed by DEEN->CH, and lastly by ENCH->DE. For the case of using 160 s of target language data, compared to no target language data being used in training, DECH->EN, DEEN->CH and ENCH->DE perform 13.8%, 3.8% and 0.4% better in UA, respectively, and 15.6%, 4.4% and 0.4% better in WA, respectively. The low performance increment in ENCH->DE might be due to the limited room for improvement. Third, except for ENCH->DE having limited improvement, DEEN->CH also has overall less improvement in UA compared to DECH->EN, when 300 s of target language data is used in training. This could be due to the difficulty of transferring Germanic languages to Sino-Tibetan languages, which probably can be alleviated by using more target language annotated data, however this hypothesis might need more investigation.

4.5.5 Comparison to existing works

We also compare our method with a few recent methods proposed for cross-language/cross-corpus SER. The comparison is in Table 4.4 where we only present the results reported on the same testing datasets (i.e. on the Emo-DB and RAVDESS) as our work. Since there is hardly any research using ESD for cross-language SER, we do not include the results for this dataset.

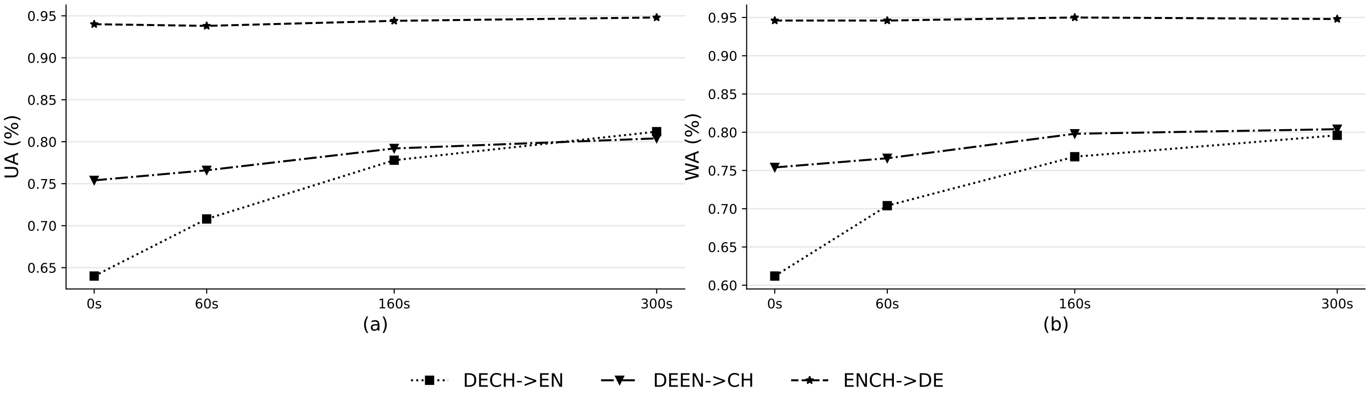


Figure 4.6: The SER performance when adding additional target language data (i.e. 60 s, 160 s and 300 s of target language recording) into training. Testing speaker IDs are still unseen during training.

The comparison shows that our proposed method has significantly improved the SER performance in both within-language and cross-language scenarios with both Emo-DB and RAVDESS datasets. This may indicate that transfer learning with pre-training on large speech data and Deep-WCCN plays an important role in DNN generalisation for SER.

4.6 Conclusions and future work

To alleviate the performance degradation in cross-language/cross-corpus SER compared to within-language/within-corpus SER, we proposed a transfer learning method that firstly uses the wav2vec 2.0 pre-trained model to transfer a time-domain audio waveform into a contextual embedding space that is shared across different languages, thereby reducing the language variabilities in the speech features. Then, by applying a re-designed Deep-WCCN, which is adapted to cope with DNN training, this Deep-WCCN layer can further reduce within-class variance caused by other factors (e.g. speaker identity, channel variability). Experimental results first show that the proposed method largely increases both within-language and cross-language SER performance compared to the eGeMAPS and Emobase feature sets that have been designed for and widely used in SER. Furthermore, an ablation study shows that Deep-WCCN can reduce the within-class variances which further improves the performance for the proposed DNN model. In contrast, the conventional WCCN does not show improvements on the eGeMAPS and Emobase feature sets with a RF classifier. Next, we show that the proposed transfer learning method exhibits good data efficiency in merging target language data in the fine-tuning process. The model speaker-independent SER performance increases for all testing target languages, and a performance gain in WA up to 15.6% and in UA up to 13.8% is achieved when only 160 s of target language data is used in the training set for fine-tuning. Finally, a comparison with recent work in cross-language/cross-corpus SER demonstrates that the proposed method can significantly improve within-language and cross-language SER performance among multiple datasets.

Future work firstly include the evaluation of the wav2vec 2.0 model pre-trained on an even larger speech dataset and the assessment of its benefit to cross-language SER performance. A good candidate dataset is proposed in [10] and consist of half a million hours of publicly available speech audio in 128 languages. Second, we plan to evaluate the proposed method on more emotive speech datasets, and to fine-tune the model on emotive speech datasets from more languages to create a general data-driven model for SER. Lastly, it may be relevant to reduce the model size through model distillation [74] to make it feasible for mobile and low-power devices.

Ref.	**Method**	**Training language**	**Testing dataset**	**Num. of classes**	**WA**	**UA**
		Within-language SER				
Rehman et. al. [121]	Hybrid Neural Network (NN)	EN+DE	RAVDESS	4	-	56.2%
		EN+EN	RAVDESS	4	-	62.5%
		EN+DE	Emo-DB	4	-	61.2%
Latif et. al. [90]	Domain adaptation	DE	Emo-DB	5	-	81.3%
Parry et. al. [115]	Datasets aggregation	EN+IT+DE	Emo-DB	3	-	69.72%
		EN+IT+DE	RAVDESS	3	-	65.67%
Desplanques and Demuynck [44]	Emotion i-vector	DE	Emo-DB	4	90.5%	-
Proposed	Transfer learning + Deep-WCCN	EN+CH EN+DE	RAVDESS	4	**79.6%**	**80.8%**
		EN+DE CH+DE	Emo-DB	4	**94.9%**	**94.6%**
		Cross-language SER				
Rehman et. al. [121]	Hybrid NN	EN+EN	Emo-DB	4	-	40.2%
Zhang et. al. [172]	Subspace learning	EN	Emo-DB	5	47.35%	-
		TU	Emo-DB	5	49.26%	-
Latif et. al. [90]	Domain adaptation	IT + UR + EN	Emo-DB	5	-	68%
Parry et. al. [115]	Datasets aggregation	EN	Emo-DB	3	-	41.99%
Desplanques and Demuynck [44]	Emotion i-vector	EN	Emo-DB	4	81.4%	-
Proposed	Transfer learning + Deep-WCCN	EN+CH	Emo-DB	4	**94.6%**	**94%**
		DE+CH	RAVDESS	4	**61.2%**	**64%**

Table 4.4: Comparison to recent work in cross-language/cross-corpus SER. EN, DE, IT, CH, TU, and UR indicate English, German, Italian, Chinese, Turkish, and Urdu language, respectively.

Chapter 5

End-to-end speech emotion recognition using a novel context-stacking dilated convolution neural network

This chapter was previously published as:
Tang, D., Kuppens, P., Geurts, L., and van Waterschoot, T., End-to-end speech emotion recognition using a novel context-stacking dilated convolution neural network. *EURASIP Journal on Audio, Speech, and Music Processing*, 2021(18), 2021.

Tang, D., Kuppens, P., Geurts, L., and van Waterschoot, T., Adieu recurrence? End-to-end speech emotion recognition using a context stacking dilated convolutional network. *In Proc. 2020 28th European Signal Processing Conference (EUSIPCO)*, 2021 January.

Abstract

Amongst the various characteristics of a speech signal, the expression of emotion is one of the characteristics that exhibits the slowest temporal dynamics. Hence, a performant Speech Emotion Recognition (SER) system requires a predictive model that is capable of learning sufficiently long temporal dependencies in the analysed speech signal. Therefore, in this work, we propose a novel end-to-end neural network architecture based on the concept of dilated causal convolution with context stacking. Firstly, the proposed model consists only of parallelisable layers and is hence suitable for parallel processing, while avoiding the inherent lack of parallelisability occurring with Recurrent Neural Network (RNN) layers. Secondly, the design of a dedicated dilated causal convolution block allows the model to have a receptive field as large as the input sequence length, while maintaining a reasonably low computational cost. Thirdly, by introducing a context stacking structure, the proposed model is capable of exploiting long-term temporal dependencies hence providing an alternative to the use of RNN layers. We evaluate the proposed model in SER regression and classification tasks and provide a comparison with a state-of-the-art end-to-end SER model. Experimental results indicate that the proposed model requires only 1/3 of the number of model parameters used in the state-of-the-art model, while also significantly improving SER performance. Further experiments are reported to understand the impact of using various types of input representations (i.e., raw audio samples vs log mel-spectrograms) and to illustrate the benefits of an end-to-end approach over the use of hand-crafted audio features. Moreover, we show that the proposed model can efficiently learn intermediate embeddings preserving speech emotion information.

Keywords— End-to-end learning, Speech Emotion Recognition, Dilated Causal Convolution, Context-Stacking

5.1 Introduction

Emotion recognition is a crucial component in present-day human-computer interaction systems. A SER system utilizes vocal expression to recognize emotions, and has inherent benefits compared to other modalities. Vocal expression is a fairly direct way to express emotions and is often easier to capture than facial expressions, for which a careful camera positioning is needed. Therefore, an SER system is complimentary to an image/video based emotion recognition system. Example applications include an SER system intended to analyse the users' emotions in a call centre to improve their services, and an intelligent robot that understands the users' emotions. Emotion research makes use of both categorical and dimensional approaches to qualify emotional experience. In the categorical approach, discrete emotion labels are used to represent qualitatively different emotional states (e.g., happy, angry, and so on). In the dimensional approach, emotional experience is described in terms of a number of basic dimensions, such as valence (ranging from positive to negative) and arousal (ranging from low to high arousal), see [128].

Early SER systems use pre-defined acoustic features to represent the audio recordings. The definition of emotion-related features in this case is a key aspect towards an accurate and robust SER system. Many hand-crafted features have been proposed for this purpose, the Geneva Minimalistic Acoustic Parameter Set (GeMAPS) includes various acoustic features such as frequency-related (e.g., pitch, formant), energy-related (e.g., loudness) and spectral (e.g., spectral slope) features, of which their effectiveness in SER has been evaluated in [51]. Also, the well-known Mel-Frequency Cepstral Coefficients (MFCC) features, which have been used in various other speech analysis tasks, including automatic speech recognition, have been applied to SER [123, 137]. However, the design of hand-crafted features requires specialized knowledge, exhaustive selection and massive experiments. Also, the feature extraction process suffers from a potentially huge information loss [102], which could be harmful to the SER performance.

With the rapid developments in Deep Neural Networks (DNNs), the feature extraction for an SER system has shifted to data-driven feature learning. In the area of image processing, Convolutional Neural Networks (CNNs) have been proven to be able to learn abstract features that have intuitively desirable properties while ascending the network layers [170]. Similar work in the area of audio processing has shown that the CNN layers can learn meaningful features by acting as onset extractors, melody extractors, low-pass filters and so on [35]. As a result, the end-to-end learning approach, in which raw microphone recording samples or shallow features are fed into a DNN, becomes feasible and attractive. In the case of SER, this DNN normally consists of both CNN layers and different types of RNN layers [153, 154, 131, 155, 33, 173, 97, 96]. The CNN layers are generally applied to the raw recording samples to produce higher-level features. A large receptive field is desired so that the DNN model can receive and learn the long-term temporal information that might be beneficial for SER, as a consequence, the number of parameters in the CNN layers will largely increase when aiming for a larger receptive field.

From the perspective of modelling sequential data, a good model should be able to learn the temporal dependencies or relations within the input sequences. SER in this case has inherent difficulties because the time constants of emotion dynamics can range from just a few seconds to over an hour [162], and these dynamics are regulated by both internal and external excitations [89]. If we assume that the human voice characteristics are a good indication of a person's internal emotional status, a good SER model should then be able to model sufficiently long temporal dependencies in recorded speech sequences. Many state-of-the-art end-to-end SER systems use Long Short-term Memory (LSTM) layers or Gated Recurrent Unit (GRU) layers as default network architecture for this purpose [153, 154, 131, 155, 33, 173, 97, 96]. However, the RNN type of layers used in the state-of-the-art SER systems suffer from several disadvantages. For example, due to the existence of the recurrent connections, the RNN layer has a sequential type of processing which results in a polynomial growth of computation time with increasing input sequence length. This type of processing is not capable to be parallelised. Also, RNN suffers from the gradient vanishing/exploding problem in processing long sequences, although this problem has been alleviated by the developments in LSTM and GRU [75, 34].

We are aiming to solve these problems that are inherited by the state-of-the-art SER systems from their RNN-type layers and we will propose a different approach to enlarge the model receptive field without largely increasing its computational complexity. This paper provides details of an improved version of the end-to-end SER model which has been proposed earlier by the authors [147]. The main contributions of this paper can be summarised as follows:

1. We provide more details and propose an updated dedicated dilated convolution block for our neural network model for end-to-end SER. This updated model remains to have a significantly large receptive field while largely reducing the number of model parameters compared to the model proposed in [147].
2. We further explore the context stacking idea, originally proposed in the WaveNet paper [157] and applied to SER in [147], with more thorough comparisons between several model variations.

3. We provide a more in-depth analysis of this new model architecture, and evaluate it on both an emotion classification task and an emotion regression task. Abundant simulations have been conducted with two well-known datasets, the REmote COLlaborative and Affective (RECOLA) [125] and the Interactive Emotional Dyadic Motion Capture (IEMOCAP) [30] datasets. Simulation results show that the proposed model surpasses the state-of-the-art CNN-RNN based models.
4. We also evaluate the effectiveness of end-to-end learning in SER by comparing SER performance using raw audio samples or log mel-spectrogram features with more traditional audio features proposed in earlier work.

The rest of the paper is organized as follows. Section 2 provides a brief overview about the most recent studies related to our work. In Section 3 we introduce the proposed model structure, the parameters of which will be optimized on the Concordance Correlation Coefficient (CCC) objective function [100] for emotion regression. In Section 4, we describe the datasets used and the experimental settings. And then, we will present the simulation results and discuss these in Section 5. Finally, Section 6 presents the conclusions and suggestions for future work.

5.2 Related work

Our proposed SER model has very large receptive field, which makes it suitable for long sequence modelling, and is based on the concepts of dilated convolutions, context-stacking, and end-to-end SER.

5.2.1 Dilated convolution

The dilated convolution is intended to increase the receptive field, which has shown successful outcomes in various audio processing tasks [157, 159]. By stacking many dilated convolution blocks in a DNN, the network will largely increase its receptive field while the model complexity and thus its computational cost will remain reasonably low. Empirical research also shows that this dilated type of convolution outperforms the canonical RNN in various sequence modelling problems [14]. Our work is inspired by the WaveNet model proposed in [157], however, we propose several modifications to suit the specific application of SER. First, we redesign the dilated causal convolution block inspired by the WaveNet model. Second, we expand the context stacking idea of [157]. Third, we add pooling layers in between the dilated convolution blocks to reduce the sequence length, so that the proposed model is able to deal with very long sequences while posing only moderate memory requirements.

There is hardly any work investigating dilated convolution in an SER framework. In [97], a dilated residual convolution is used to further process the extracted acoustic features. However, the use of the dilated residual convolution in [97] served a different purpose than ours, which in [97] is to facilitate the reduction of the receptive field and hence yield a strong ability to learn local context. Temporal modelling in [97] was achieved by using LSTM and a self-attention mechanism.

5.2.2 Context stacking

The context stacking idea was originally proposed in the WaveNet paper [157]. In this idea, multiple trainable DNNs are connected/stacked by local conditioning. More generally, local conditioning has in fact been widely used in various audio-related tasks. In [139], the Text-to-speech (TTS) model is conditioned on predicted Mel-spectrograms. In another TTS system [82], the model is conditioned on pre-trained speaker embeddings to synthesize speech for a particular person. A similar idea to local conditioning was applied in an SER system [96] by concatenating both the handcrafted acoustic features for SER and the lexical text features to obtain an emotion classifier. Throughout this paper, however, we refer to context stacking only when both the conditioned context and the model itself are trained jointly.

5.2.3 End-to-end SER

End-to-end learning has attracted vast attention from the deep learning community. The modelling follows a data-driven approach, and little or no specialised knowledge is exploited in this learning process. In audio processing, the traditional learning pipeline (e.g., pre-processing, feature extraction, modelling, inferencing.) is taken over by a single DNN, where at the input end raw audio samples or shallow frequency features such as mel-spectrograms are provided, and at the output inference results are obtained (e.g., class labels, parameter estimates).

In SER, Trigeorgis *et al.* proposed to use a CNN to extract features from the raw audio samples, and then use two bidirectional LSTM layers to model the temporal information [153, 154, 155]. Satt *et al.* afterwards proposed to apply the CNN and the LSTM layers on a modified log-spectrogram, which is harmful for clean SER performance, but it is more robust to noise [131]. In [130], Sarma *et al.* replaced the CNN layers with Time-Delay Neural Network (TDNN) layers. The TDNN tends to increase the receptive field of the network, somewhat resembling the dilated convolution, and was originally used in the speaker recognition problem to extract the x-vector [140].

Some other end-to-end SER systems make use of an attention mechanism which has its origins in Natural Language Processing (NLP) [160]. Chen *et al.* proposed an SER model consisting of 3-D CNN layers, LSTM layers and the attention layer. This model can learn time-frequency relations from a time stack of log mel-spectrograms [33]. In [96], Li *et al.* applied a self-attention mechanism along with CNN layers, which can import emotion-salient information from the audio feature inputs. Similarly, in [174, 173] an end-to-end CNN-RNN based model was combined with the attention mechanism.

5.2.4 Difference with speaker recognition

To some extent, the emotion classification task resembles the automatic speaker recognition problem where the target classes are speaker identities. Similar to SER, the MFCC and frequency-related features (e.g., pitch) are widely used in early automatic speaker recognition research [122, 53]. In [122], Reynolds and Rose propose to use the Gaussian Mixture Model (GMM) to model the speakers' MFCC distribution. The concatenation of the mean vectors of the GMM (referred to as the supervector in [122]) can be used to represent the speakers' identities. This high-dimensional representation is then used to extract the i-Vector, which is a low-dimensional representation of the total variability (i.e., speaker variability and channel variability) [41]. A similar work in SER proposes to extract the emotion representations from the speaker i-Vectors [44]. However, in SER, we aim to model very long temporal dependencies of the input features, especially in speech emotion regression, because it is expected that the variation of the features over time is distinguishing for different emotions. This is different from the automatic speaker recognition problem, in which the modelling of the temporal information is less important since the speaker identity is not expected to change within a given speech frame.

5.3 Method

Our proposed end-to-end SER model is shown in Figure 5.1, denoted as the DiCCOSER-CS (Dilated-Causal-Convolution-Only Speech Emotion Recognition with Context Stacking). The network consists of dilated causal convolution

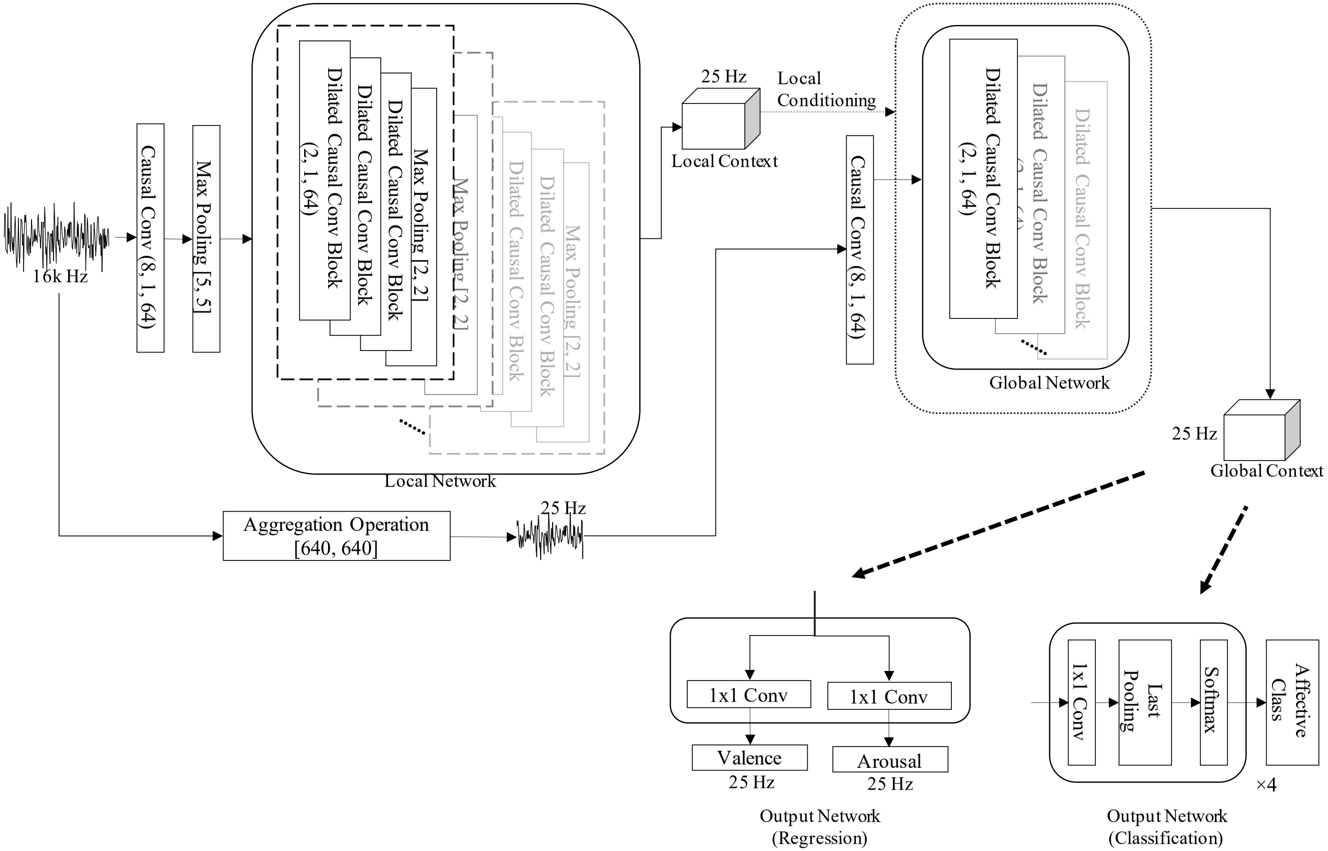

16k Hz
Causal Conv (8, 1, 64)
Max Pooling [5, 5]
Dilated Causal Conv Block (2, 1, 64)
Dilated Causal Conv Block
Dilated Causal Conv Block
Max Pooling [2, 2]
Max Pooling [2, 2]
Dilated Causal Conv Block
Dilated Causal Conv Block
Max Pooling [2, 2]
Local Network
Aggregation Operation [640, 640]
25 Hz
25 Hz
Local Context
Local Conditioning
Causal Conv (8, 1, 64)
Dilated Causal Conv Block (2, 1, 64)
Dilated Causal Conv Block
Dilated Causal Conv Block
Global Network
25 Hz
Global Context
1x1 Conv
1x1 Conv
Valence
25 Hz
Arousal
25 Hz
Output Network (Regression)
1x1 Conv
Last Pooling
Softmax
Affective Class
×4
Output Network (Classification)

Figure 5.1: The proposed end-to-end SER model denoted as the Dilated-Causal-Convolution-Only Speech Emotion Recognition with Context Stacking (DiCCOSER-CS). The convolution filter width, stride and filter depth is listed in round brackets, the pooling width and stride is listed in square brackets.

blocks that are used for increasing the receptive field [157, 169] (Sec. 5.3.1), then two sub-networks are stacked and trained jointly (Sec. 5.3.2). Finally, the model outputs the arousal and valence estimates for a SER regression task and the class posterior probability for a SER classification task. For regression, the model is trained to minimize the CCC objective function (Sec. 5.3.3).

5.3.1 Dilated causal convolution blocks with local conditioning

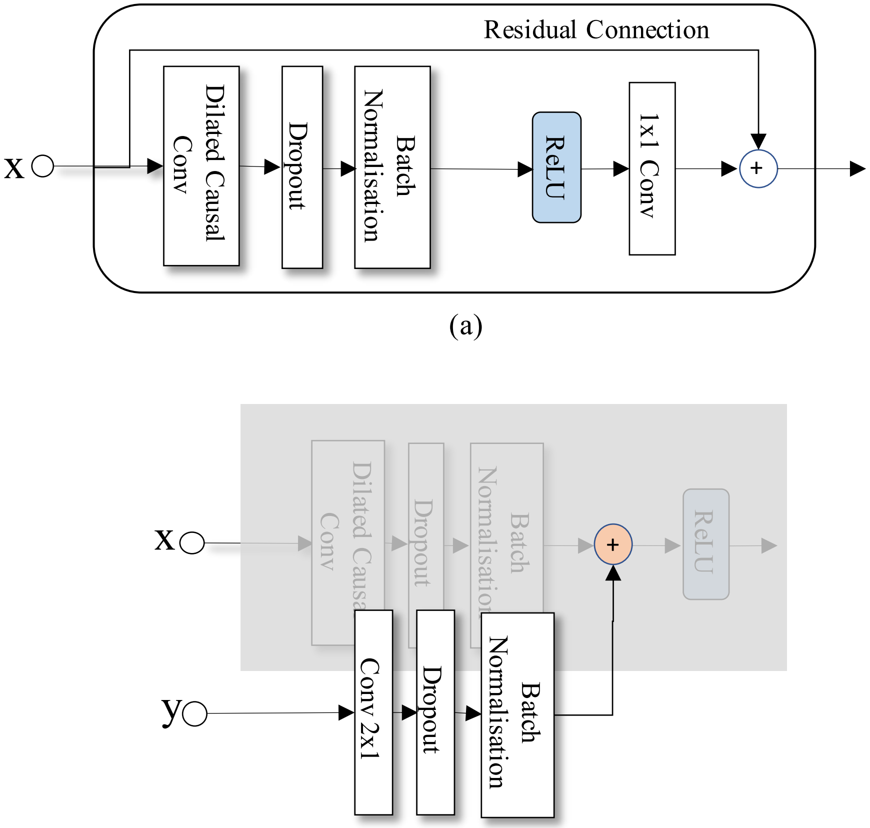
The dilated causal convolution block, shown in Figure 5.2 (a), is one of the basic building blocks in the proposed model. This block is inspired by [157], but from our experiments, we found that using the original dilated causal convolution block in [157] lead to a slow training convergence. Thus, we have redesigned the block in the following aspects. Every dilated causal convolution block consists of two paths, one being the residual connection path, and the other being the convolution path. Firstly, the residual path connects the input directly to the output, which has been shown to allow to learn an identity mapping, thus it can speed up the training and avoid over-fitting [72]. Secondly, in the convolution path, a dilated causal convolution is applied to the input, and it is immediately followed by a dropout layer to prevent the model from over-fitting [143] and a batch-normalisation layer [80] to further speed up the training. Thirdly, we applied the Rectified Linear Unit (ReLU) non-linear activation function [65] to the convolution output. This non-linear activation function has been widely used in modern DNNs for its easy gradient calculation and its effect of giving the model a faster and better convergence. Finally, after going through a 1 × 1 convolution, the dilated causal convolution path is summed together with the residual path to generate the final output of the block.

We implement the local conditioning similarly to [157]. Consider an input consisting of a sequence of examples, and another sequence $\boldsymbol{y}$ having the same length as $\boldsymbol{x}$ containing the conditioning information. If the filter output is $\boldsymbol{z}$, a local conditioning is defined as follows, adopting the notation from [157]:

$$\boldsymbol{z} = \mathrm{ReLU}(W_{f,k} * \boldsymbol{x} \,+\, V_{f,k} * \boldsymbol{y}) \tag{5.1}$$

where $W_{f,k}, V_{f,k}$ are the learnable "filter" parameters in the k^{th} layer, $*$ is the convolution operation, and $\mathrm{ReLU}(\cdot)$ is the ReLU activation function. Dropout and batch-normalisation layers are added after the convolution as well, as shown in Figure 5.2 (b).

The dilated causal convolution blocks are then stacked many times with different dilation number in the network. The dilation number defines the time lag, i.e., the number of samples that are skipped in between two input samples used in



(a)

(b)

Figure 5.2: (a) The dilated causal convolution block, and (b) the local conditioning.

the convolution with the filter $W_{f,k}$. Figure 5.3 shows a stack of dilated causal convolution blocks with filter width 2, and dilation numbers 1, 2 and 4.

5.3.2 Context stacking using local conditioning

Referring to the context stacking idea in [157], we propose a stacked structure using local conditioning for end-to-end SER, see Figure 5.1. This proposed structure consists of three learnable sub-networks. First, one sub-network has a

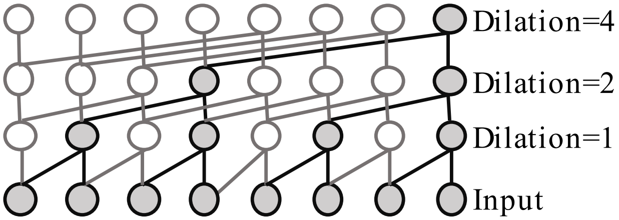


Figure 5.3: A stack of dilated causal convolution blocks.

relatively small receptive field, denoted as the “local network”, that receives raw input samples and produces the local context. The local network is capable to be locally conditioned on extra information relating to the input frame (e.g., the speaker gender information, the lexical text information and so on), however, we do not investigate the impact of adding such extra information in this paper. Second, the other sub-network, denoted as “global network”, has a relatively wide receptive field that receives downsampled input audio samples, and is aiming to learn global (i.e., long-term) temporal dependencies. The two networks connect by letting the “local network” define the local conditioning on all the layers in the “global network”. We also add pooling layers in the “local network” to down-sample the sequence with the aim of reducing computational costs and memory requirements. Finally, the output from the “global network” can be processed by successive convolution-type layers to generate the desired, task-dependent outputs. The reason why we propose to use convolution layers for processing in the final stage is because we are aiming to design a parallelisable network.

5.3.3 CCC objective function

For a regression task, e.g., an SER model inferencing arousal/valence values, when the training labels are given for very short time intervals (e.g. 40 ms), the levels of affect then can be predicted on the same time scale, i.e., for every 40 ms of a speech recording. In this case, an affective evolution curve can be obtained, e.g. for visualisation purposes. Thus, not only the prediction values should be close to the corresponding ground-truth labels, but also the correlation between the entire prediction sequence and label sequence is important. The Mean Squared Error (MSE) or Mean Absolute Error (MAE) loss on the sample level as used in [123], does not consider this correlation. A loss function that has a direct link to the evaluation metric based on the CCC (ρ_c) [100] has been proposed in [153, 154, 155].

Given the predicted sequence (denoted by index m) of arousal/valence values and its corresponding ground-truth sequence (denoted by index n), the CCC loss is defined as:

$$\mathcal{L}_c = 1 - \rho_c = 1 - \frac{2\rho\sigma_m\sigma_n}{\sigma_m^2 + \sigma_n^2 + (\mu_m - \mu_n)^2} = 1 - \frac{2\sigma_{mn}^2}{\sigma_m^2 + \sigma_n^2 + (\mu_m - \mu_n)^2} \tag{5.2}$$

where ρ is the Pearson Correlation Coefficient (PCC), μ_m and μ_n are the sample means, σ_m^2 and σ_n^2 are the sample variances, and σ_{mn}^2 denotes the covariance between the two sequences. Therefore, prediction sequences exhibiting a weak correlation with the ground-truth sequences as well as shifted amplitude of the prediction values are both penalised in one loss function.

5.4 Experimental setup

5.4.1 Implementation Details

In this section, we describe how to implement the DiCCOSER model. The raw audio samples are firstly fed into a causal convolution layer which has a filter width equal to 8. The filter width of this convolution layer on one hand should be wider than the successive pooling size so that it has a sufficiently large receptive field and is capable of extracting salient features which will be selected by the following pooling operation, on the other hand the filter width should be kept small to maintain a low model complexity. This causal convolution layer maps the single-channel audio sample sequences to 64-dimensional vector sequences. Next, these sequences are downsampled in a max-pooling layer with size 5, and are then ready for the processing by the "local network". The configuration of the pooling size is task-dependent, such that after the pooling operations, the output sequence has the desired sampling rate.

Stride one and zero-padding is used across all convolution layers in the entire network to retain the same sequence length after each operation.

Local network

Firstly, we construct the local network, which contains a stack of dilated causal convolution blocks. These blocks have a filter depth equal to 64, and their dilation numbers are chosen to correspond to a subset of a geometric series, and are repeated a few times. Max-pooling layers are included after every stack of dilated causal convolution blocks, and the number of max-pooling layers is task-dependent. Specifically, we design each stack in our local network to have a set of dilation numbers $D^{\text{local}} = \{2^k, k = 0, 1, 2\}$, and a total of 7 such stacks are used. There is one pooling-size-2 max-pooling layer after each stack of dilated causal convolution blocks, which results in 7 pooling layers in total. The pooling layers in the local network progressively downsample the processed data from 16 kHz at the input to 25 Hz at the output of the local network, which is the same as the label sampling rate for the regression task.

In parallel to the local network, the raw audio input sequence is directly downsampled from 16 kHz to 25 Hz. We propose two aggregation operations that extract useful features from the signal frames in this parallel feedforward branch: a) a *max-pooling aggregation* that extracts the maximum value from the frames, and b) an *RMS aggregation* that calculates the RMS value per frame. We believe that these features can be representative of the original audio frames at the reduced sampling rate of 25 Hz, and they are also related to the expression of emotions [145]. The downsampled sequences are finally mapped to 64-dimensional vector sequences by a causal convolution layer with filter size 8.

Global network

Secondly, in the global network, the dilated causal convolution blocks have filter depth equal to 64 as well. The aim of the global network is to learn the long-term temporal dependencies from a more global perspective. It can ensure this aim by processing the downsampled input sequences because this not only reduces the computational cost of the global network, but also prevents the global network from attempting to model subtle changes in the original raw input samples. In addition, thanks to the context-stacking structure, some information lost in the downsampling operation is selectively passed to the global network. In order to be able to effectively perform the dilated convolutions, we propose to have the largest dilation number equal to the length of the processed input frames. We set the global network dilation numbers as $D_{\text{global}} = \{1, 2, 2^2, \ldots, 2^{\lfloor log_2(L) \rfloor}, L\}$, where L is the input frame length and $\lfloor \cdot \rfloor$ denotes the flooring operation. For example, for a 20 s audio input frame, the global network operates on the downsampled 25 Hz sequence of length 500, such that $D_{\text{global}} = \{1, 2, 4, 8, 16, 32, 64, 128, 256, 500\}$.

Finally, all the dilated causal convolution blocks are conditioned on the local contexts. We implement these context stacking filters (filter "$V_{f,k}$" in (5.1)) by normal CNN layers with filter width 2 and filter depth 64.

Output network

Finally, the output network converts the global network output to the desired output formats. In our regression task, there are two 1×1 convolution layers with filter depth 1 in the output network that yield the arousal and valence predictions.

In our classification task, there are four 1×1 convolution layers with filter depth 1 corresponding to 4 different class outputs. These are then processed by a last-pooling layer to only keep the last convolution output for every layer. This is because our model has a receptive field as large as the input sequence length, so that the last convolution output can be trained to contain global information. Finally, a softmax layer is applied, and the output of the softmax layer can be interpreted as the class posterior probability.

5.4.2 Datasets

To evaluate the regression and classification performance of the proposed model and to compare it with the state-of-the-art end-to-end CNN-LSTM based SER model, we used two widely used affectively labelled datasets: the RECOLA dataset [125] for the regression task and the IEMOCAP dataset [30] for the classification task.

RECOLA

The RECOLA dataset [125] contains abundant affective data with both arousal and valence annotations per 0.04 s. The arousal and valence annotations are continuous values in the range [-1, 1]. The speech data consist of interviews in which people talk about real-life stories. However, since the database is not fully publicly available, we can only acquire a sub-partition that is used in the 2015 and 2016 Audio/Visual Emotion Challenge and Workshop (AVEC) [124, 156] competitions. We only use the raw audio out of four modalities (audio, video, Electrocardiogram (ECG), and Electro-dermal activity (EDA)) provided by the competition, and their corresponding labels. This sub-partition contains eighteen 5-minute long audio clips, equally divided into a training set and a development set. In both sets, there are 5 clips with female speakers and 4 clips with male speakers, as indicated in Table 5.1. The language of all audio clips is French. The sampling frequency is 44.1 kHz, and we downsampled all data to 16 kHz for our simulations. We further divide this sub-partition into 5 folds. The partitions are summarized in Table 5.2.

Train	train_1 (M)	train_2 (F)	train_3 (M)	train_4 (F)	train_5 (F)
	train_6 (M)	train_7 (F)	train_8 (F)	train_9 (M)	
Dev	dev_1 (F)	dev_2 (M)	dev_3 (M)	dev_4 (M)	dev_5 (F)
	dev_6 (M)	dev_7 (F)	dev_8 (F)	dev_9 (F)	

Table 5.1: RECOLA database sub-partition naming and speaker gender in brackets

	Fold 1	***Fold 2***	***Fold 3***	***Fold 4***	***Fold 5***
Valid	train_1, dev_1	train_3, dev_3	train_5, dev_5	train_7, dev_7	train_9
Test	train_2 dev_2	train_4, dev_4	train_6, dev_6	train_8, dev_8	dev_9

Table 5.2: RECOLA database 5-fold cross-validation partitions

IEMOCAP

The IEMOCAP dataset [30] contains English acted speech dialogues by 10 professional actors. There are in total 5 sessions, each featuring one actor and one actress performing the dialogues with a script or in an improvised manner. There is no speaker overlap across the 5 sessions. We have used improvised utterances from four emotional categories {neutral, anger, sadness and happiness}, which in total are 2280 utterances (neutral: 1099, anger: 289, sadness: 608, happiness: 284). We trim or pad zeros in front of the recordings to make them into 3 s long clips. We divide the cross-validation folds by the session numbers (i.e. session 1 to 5). The audio files have also been downsampled to 16 kHz.

Data augmentation

To overcome over-fitting, we propose to apply two simple yet distinct data augmentation methods to the regression task and the classification task respectively. For the regression task, we propose to use a sliding window data augmentation, that is, we use a 20 s long sliding window to generate the training frames from the original 5 min recordings. Successive sliding windows are shifted by 4 s.

For the classification task, instead of using speed perturbation as in [1], which we believe may change the affective expression of the recording, we propose to use a random flipping data augmentation. That is, with a given probability (e.g. equal to 50%), the input sequence is flipped (i.e., time reversed) before being fed into the model. This data augmentation method will largely retain the affective meaning of the original speech recording, and will prevent the model to become biased by some disturbing context factors having a specific temporal pattern or dependence (e.g. room reverberation).

Training and evaluation settings

For the regression task, we optimise the model by minimizing the CCC loss as described in Section 5.3.3, and for the classification task we minimize the cross-entropy loss. The RMSProp optimizer [151] is used to train the models, with a fixed learning rate of 10^{-4}. Batch-size is 5 for the regression task and 10 for the classification task, due to memory constraints. l_2 regularization with a regularization parameter of 10^{-4} is applied and the dropout rate equals 0.5. No post-processing is performed on the output predictions.

We conduct all the experiments using 5-fold cross-validation. More specifically, for the RECOLA dataset, in every run, we use 4 folds of data to train the model, then we split the last fold into a validation set and a testing set as listed in Table 5.2. In this way, there is no speaker overlapping across the three subsets. For the IEMOCAP dataset, since we aim to evaluate the model's generalisability across speakers and sessions, for each run we use 3 folds for training, and the remaining 2 folds for validation and testing respectively. In each experiment, we employ early stopping, i.e., we stop the training on the highest validation performance, and then use that model to compute the testing performance.

The CCC [100] is used to evaluate the regression task, whereas the Weighted Accuracy (WA) and the Unweighted Accuracy (UA) are used to evaluate the classification task. WA is the overall accuracy of the entire testing data, which indicates the overall model performance across all classes, whereas UA is the average accuracy for each emotion class, which marginalises out the effect of the existence of class imbalance. All the experiments are repeated 5 times using data from different folds as stated above. The results are averaged across the 5 folds before being reported.

5.5 Results and discussions

5.5.1 Evaluation of the context-stacking idea and the model architectures

First, we evaluate the context stacking which is considered as one of the contributions of this paper to SER, and a few model architecture variations. More specifically, we aim to answer the following questions:

1. Can the context-stacking architecture increase the model performance?
2. What is the optimal choice between the max-pooling aggregation and the RMS aggregation?

We propose and evaluate four model variations, and their topologies are shown in Figure 5.4:

a. A model that contains only one sequential network (denoted as DiCCOSER), in which the context stacking is not used, is shown in Figure 5.4 (a).

b. A model that deploys context stacking, but the local contexts are directly fed to the global network, and the global network is locally conditioned on

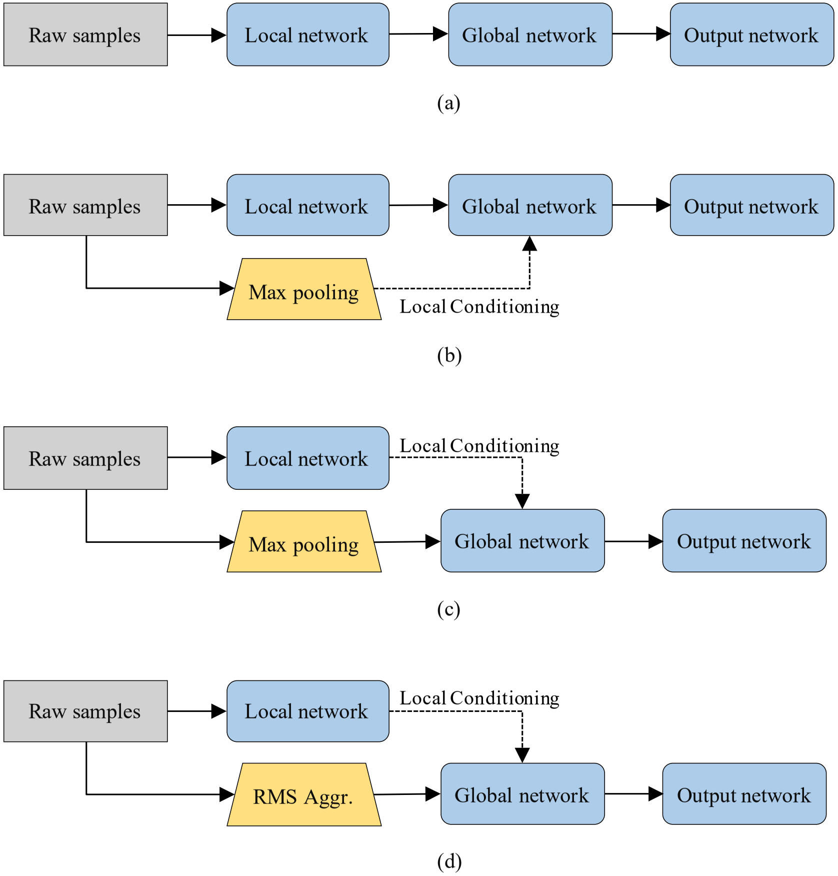


Figure 5.4: The proposed model variations: (a) DiCCOSER, (b) DiCCOSER-CS-V2, (c) DiCCOSER-CS max, and (d) DiCCOSER-CS rms. “RMS Aggr.” indicates using the RMS aggregation.

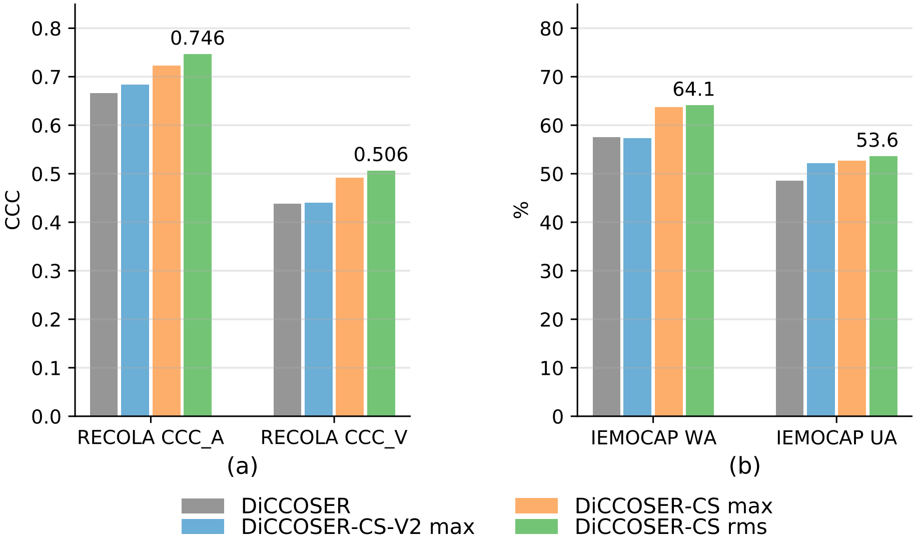


Figure 5.5: Performances of four proposed model variations on (a) the RECOLA dataset and (b) the IEMOCAP dataset. Suffix "-CS" indicates using the context-stacking, "max" and "rms" indicates using the max-pooling aggregation and the RMS aggregation respectively.

the downsampled input sequences (denoted as DiCCOSER-CS-V2 max), is shown in Figure 5.4 (b). In addition, the max-pooling aggregation is used.

c. The proposed model that deploys context stacking as described in Sections 5.4.1 and 5.4.1 (i.e., the global network receives the aggregated downsampled input sequences and context stacking is performed using the local contexts), and uses the max-pooling aggregation method in the downsampled stream (denoted as DiCCOSER-CS max), is shown in Figure 5.4 (c).

d. The proposed model having similar architecture as the model variation c), but using the RMS aggregation method (denoted as DiCCOSER-CS rms), is shown in Figure 5.4 (d).

All the model variations have dilation numbers in the local network and the global network as described in Sec 5.4.1 and Sec 5.4.1.

We evaluate the proposed model variations on both the regression task (on the RECOLA dataset) and the classification task (on the IEMOCAP dataset). The average testing results for both tasks are illustrated in Figure 5.5. First, we can see that the model variations with context stacking (i.e. the DiCCOSER-CS-V2 max, the DiCCOSER-CS max, and the DiCCOSER-CS rms variations) perform better than the DiCCOSER model that does not use context stacking in both tasks. Compared to the DiCCOSER model the DiCCOSER-CS max variation improves arousal CCC, valence CCC, WA and UA with about 8.5%, 12.3%, 10.7%, and 8.2% respectively, and the DiCCOSER-CS rms variation improves arousal CCC, valence CCC, WA and UA about 12%, 15.5%, 11.5%, and 10.3% respectively. On the other hand, the improvements for the DiCCOSER-CS-V2 model on the RECOLA dataset is not significant (it improves arousal CCC and valence CCC with about 2.7%, 0.5%), and the DiCCOSER-CS-V2 model improves UA on the IEMOCAP dataset with about 7.2%, but shows a small degradation in WA (about 0.3%). Second, in the case with context stacking, the model with the RMS aggregation method performs better than the model with the max-pooling method, and both DiCCOSER-CS variations perform better than the DiCCOSER-CS-V2 variation. However, the UA difference between the variations DiCCOSER-CS-V2 max (UA equal to 52.1%) and DiCCOSER-CS max (UA equal to 52.7%) is not large. Overall, the best performance is obtained with the DiCCOSER-CS rms model. In the regression task, its arousal CCC equals 0.746, and its valence CCC equals 0.506. In the classification task, the DiCCOSER-CS rms model achieves a WA equal to 64.1% and an UA equal to 53.6%.

We can conclude that the context-stacking architecture does improve the SER performance. Also, the classification performance of the DiCCOSER-CS max and the DiCCOSER-CS rms variations shows a similar trend in WA and UA, which indicates that the improved accuracy is not due to the bias towards an individual class. Furthermore, the RMS aggregation method works better than the max-pooling method, so that we keep using the DiCCOSER-CS rms model for the further experiments.

5.5.2 Comparisons with the state-of-the-art model in the speaker-independent setting

In the second experiment, we compare our proposed model with the state-of-the-art CNN-LSTM model. In our dataset partitions, in each fold, the speakers in the training, validation and testing sets are not overlapping, so that the model performance is speaker-independent. The baseline model proposed by Tzirakis et al. [155] was originally proposed for the emotion regression task on the RECOLA dataset, and consists of three CNN layers with max-pooling and dropout layers in between, and subsequently two LSTM layers. The model has

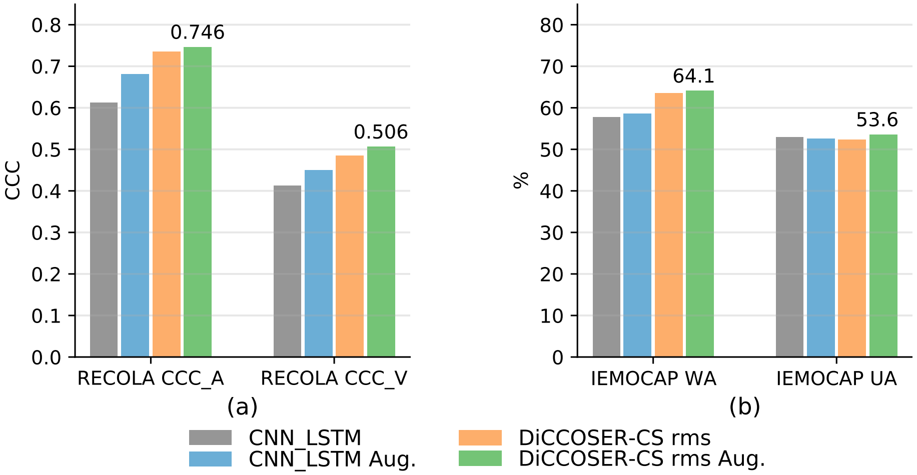


Figure 5.6: Performance comparisons to the baseline CNN-LSTM model. "Aug." indicates using the sliding window augmentation in (a), or the random flipping augmentation in (b).

been implemented here with settings as stated in [155], and in the classification task, we have added a global average pooling layer and a softmax layer on the outputs of the last LSTM layer of the baseline model. This model is optimised on the CCC loss for the regression task, and on the cross-entropy loss for the classification task, identically to our proposed model. Lastly, we also evaluate the baseline model trained with the proposed data augmentation methods (the sliding window augmentation and the random flipping augmentation method, denoted as "Aug."), which was not proposed in their original paper [155]. To have fair comparisons, we also conduct experiments using our proposed model without data augmentation. The results are shown in Figure 5.6.

The results firstly illustrate that the proposed model, with the proposed data augmentation methods, outperforms the CNN-LSTM model with or without data augmentation in all the testing cases. More specifically, in the regression task, the DiCCOSER-CS rms model with sliding window augmentation, achieves arousal CCC equal to 0.746 and valence CCC equal to 0.506, compared to the CNN-LSTM model with sliding window where arousal CCC is 0.681 and valence CCC is 0.449. Hence the DiCCOSER-CS rms model with sliding window has improved the arousal CCC and valence CCC with 9.5% and 12.7% respectively. In the classification task, the DiCCOSER-CS rms model with random flipping augmentation achieves a WA equal to 64.1% and a UA equal to 53.6%, whereas the CNN-LSTM model with random flipping augmentation yields WA and UA equal to 58.6% and 52.6% respectively. Secondly, the proposed data augmentation methods can improve the SER performance for both the baseline CNN-LSTM model and the proposed DiCCOSER-CS rms model. More specifically, with regards to the regression task, the sliding window augmentation improves the CNN-LSTM model performance from 0.613 to 0.681 on arousal CCC and from 0.412 to 0.45 on valence CCC. Also, it improves the DiCCOSER-CS rms model performance on arousal CCC from 0.734 to 0.746, and valence CCC from 0.484 to 0.506. However, in the classification task, the random flipping augmentation only gently improves the WA performance (the CNN-LSTM model with augmentation improves WA from 57.8% to 58.6%, and the DiCCOSER-CS rms model with augmentation improves WA from 63.5% to 64.1%), and also the UA of the DiCCOSER-CS model (from 52.3% to 53.6%), but there is a small decrement in the WA of the CNN-LSTM (about 0.5%). Nevertheless, these results may indicate that these data augmentation methods are helpful to improve the SER testing performance.

Finally, an interesting comparison on the number of parameters is given in Table 5.3. It is shown that even if the proposed DiCCOSER-CS model has the best overall performance on both the regression and classification tasks, it only has about 1/3 of the number of parameters compared to the baseline CNN-LSTM model. The low number of parameters implies that the proposed model can be processed much faster in both training and inferencing even with a single thread.

	CNN-LSTM [155]	DiCCOSER-CS RECOLA	DiCCOSER-CS IEMOCAP
Number of parameters	$\approx 1300 \cdot 10^3$	$\approx 475 \cdot 10^3$	$\approx 430 \cdot 10^3$

Table 5.3: Summary of the number of model parameters.

5.5.3 Comparisons with different input features

In this experiment, we aim to evaluate the impact of using different input features on the SER performance. More specifically, we conduct experiments using log mel-spectrogram features in both the baseline CNN-LSTM model and the proposed DiCCOSER model, since log mel-spectrogram features are widely used in end-to-end SER[131, 33, 174]. The results are then compared with the aforementioned models which use raw time domain audio samples as an input. Because the log mel-spectrogram features have a much lower sampling rate than the raw audio samples, we modify the baseline CNN-LSTM model and our proposed DiCCOSER model with/without context stacking to have less pooling layers and a smaller pooling width to work with these features. It is worth to mention that, in the case of using the DiCCOSER-CS model, we only change the local network to receive the log mel-spectrogram features, in which the local context is generated to be used by the global network via local conditioning. The global network still uses the aggregated raw samples as input. To extract the log mel-spectrogram features, we use a Short-time Fourier transform (STFT) with a window size equal to 0.04 s, 50% overlap, and 40 mel-frequency bands in the range of [0, 7600] Hz. Finally, we use the logarithmic values for numerical stability.

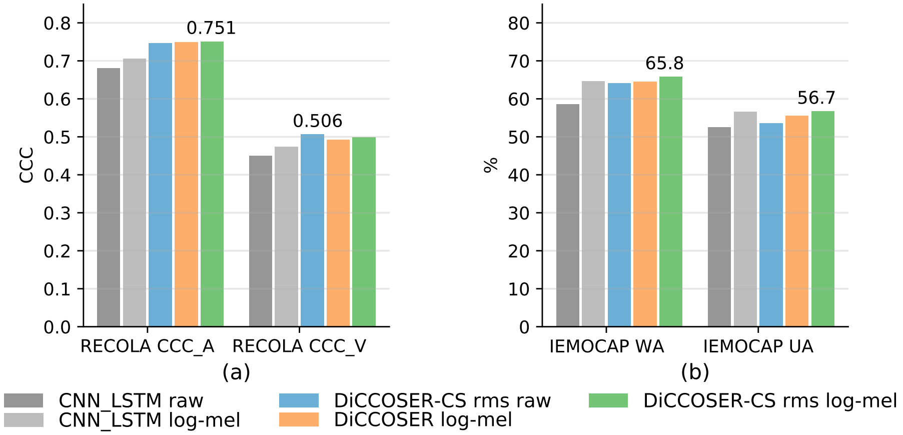


Figure 5.7: Comparisons between using raw time domain samples and using log mel-spectrograms as the SER model input.

The results are illustrated in Figure 5.7. We can see that using the log mel-spectrogram features can improve the SER performance of the CNN-LSTM model, however, the improvements for the DiCCOSER models are less pronounced. Firstly, with regard to the regression task on the RECOLA dataset in Figure 5.7 (a), using log mel-spectrogram features significantly improves the CNN-LSTM model performance (arousal CCC and valence CCC increase with about 3.5% and 5.1% respectively). Similarly, the CNN-LSTM model with log mel-spectrogram features also largely improves the WA and UA on the IEMOCAP dataset (WA and UA increase with 10.4% and 7.6% respectively). This makes the performance of the CNN-LSTM model with log mel-spectrogram features comparable to the performance of some of the proposed DiCCOSER models. Its UA achieves 56.6% which is higher than the DiCCOSER-CS rms model with raw sample inputs (UA equal to 53.6%), and is higher than the DiCCOSER model with log mel-spectrogram features (UA equal to 55.5%). The best WA (equals to 65.8%) and UA (equal to 56.7%) are however still obtained with the DiCCOSER-CS rms model with log mel-spectrogram input.

Secondly, for the DiCCOSER models, there is no obvious difference among using the raw audio samples or the log mel-spectrogram features as an input. Regarding the performance for the regression task on the RECOLA dataset, the best arousal CCC (equal to 0.751) is obtained with the DiCCOSER-CS rms model with the log mel-spectrogram input, which is slightly higher than the DiCCOSER-CS rms model with the raw sample input (arousal CCC equal to 0.746), and is slightly higher than the DiCCOSER model with log mel-spectrogram input (arousal CCC equal to 0.749). Analogously, with respect to valence CCC, the three model variations perform equally well (valence CCC from high to low: DiCCOSER-CS rms model with raw audio sample input gives a valence CCC equal to 0.506, the DiCCOSER-CS rms model with log mel-spectrogram input gives a valence CCC equal to 0.498, and the DiCCOSER model with log mel-spectrogram input gives a valence CCC equal to 0.492). Similar conclusions can be drawn from the classification results on the IEMOCAP dataset, although the log mel-spectrogram features slightly improve the DiCCOSER model and the DiCCOSER-CS model performance, especially in UA, which may indicate that using log mel-spectrogram input features provides higher robustness to class imbalance than the raw audio input, but there is no apparent winner among the DiCCOSER variations. The best results for the classification task are obtained with the DiCCOSER-CS rms model using the log mel-spectrogram features (WA equal to 65.8% and UA equal to 56.7%).

Thirdly, the conclusions from the previous experiments are still valid, i.e., the DiCCOSER model with context stacking performs slightly better than the DiCCOSER model without context-stacking, and the DiCCOSER-CS rms model outperforms the CNN-LSTM baseline model when the same input features are used.

Finally, we also compare our end-to-end SER performance with some existing SER models that use traditional audio features. The comparisons are listed in Table 5.4. Sahu *et al.* [129] and Jiang *et al.* [83] have evaluated the GeMAPS features on the IEMOCAP dataset, and Jiang *et al.* [83] have evaluated MFCCs as well. The results indicate that the end-to-end learning models (including the baseline CNN-LSTM model [155] and the proposed DiCCOSER models), using time-domain raw audio samples or shallow features such as the log mel-spectrogram features, outperfom the models using traditional audio features.

		RECOLA		IEMOCAP	
	CCC_A	***CCC_V***	***CCC Mean***	***WA***	***UA***
Sahu et al. [129] (GeMAPS)	-	-	-	56.85%	-
Jiang et al. [83] (GeMAPS)	-	-	-	-	41%
Jiang et al. [83] (MFCCs)	-	-	-	-	35%
Valstar et al. [156] (GeMAPS)	0.683	0.375	0.529	-	-
Ringeval et al.[123] (low-level descriptors)	**0.757**	0.26	0.509	-	-
CNN-LSTM Tairakis et al. [155] (raw samples)	0.681	0.5	0.591	58.6%	52.6%
CNN-LSTM Tairakis et al. [155] (log mel-spectrogram)	0.705	0.473	0.589	64.7%	56.6%
Proposed DiCCOSER-CS (raw samples)	0.746	**0.506**	**0.626**	64.1%	53.6%
Proposed DiCCOSER-CS (log mel-spectrogram)	0.751	0.498	0.6245	**65.8%**	**56.7%**

Table 5.4: Performance comparisons to SER models trained using traditional audio features, best performances are in bold.

Similarly with the RECOLA dataset, the CNN-LSTM model and the proposed DiCCOSER model with raw or shallow features (i.e. log mel-spectrogram features) outperform the traditional audio features (such as the GeMAPS or Low-level descriptor features evaluated in Valstar *et al.* [156] and Ringeval *et al.* [123] respectively) in terms of the mean CCC performance.

5.5.4 Visualisations of the local contexts

In this experiment, we aim to investigate the local contexts produced by the proposed SER model. Ideally, the local contexts should learn the information that has been lost during the aggregation step applied to the input sequence in the parallel feedforward branch. To this end, we visualise the local contexts computed from the IEMOCAP test sets in different cross-validation folds. These local contexts are originally 64-dimensional, but for visualisation purposes here we first average them across time per input audio frame, then we compute their first 2 principal components using Principal Component Analysis (PCA). In Figure 5.8, where the horizontal axis corresponds to the first principal component, and the vertical axis corresponds to the second principal component. Finally, we assign colours to the local contexts corresponding to their class labels. Columns (a) and (b) in Figure 5.8 are corresponding to the DiCCOSER-CS model with the max-pooling aggregation and with the RMS aggregation respectively.

We can make the following observations from the visualisations of the local contexts. First, we can see that the local contexts tend to form clusters that correspond to the class labels. The most discriminative cluster is the cluster corresponding to “Sad” (in light blue), and another discriminative cluster is the “Angry” cluster (in yellow). The “Neutral” and “Happy” clusters are overlapping in most of the cases. This can be explained by the fact that the angry emotion has a very high energy (high arousal), and the sad emotion has a very low energy (low arousal) which are both more distinguishable than the happy (medium to high energy/arousal) and neutral emotion. Second, we found that the first principal component represents the arousal properties of the emotions. That is, since arousal indicates the activation/energy of an emotion [128], if we look at the emotion clusters along the first principal component axis (from left to right along the horizontal axis), these emotion clusters are arranged from low activated emotion clusters to high activated emotion clusters or the other way around. Thus, the first principal component of the local context is actually highly positively or negatively correlated with the arousal, even if we only supervise the training with class labels (i.e. categorical labels such as sad, happy, angry and neutral). However, the second principal component of the local contexts does not show a significant correlation to the valence, which indicates the pleasurableness of an emotion. Finally, we observe that the

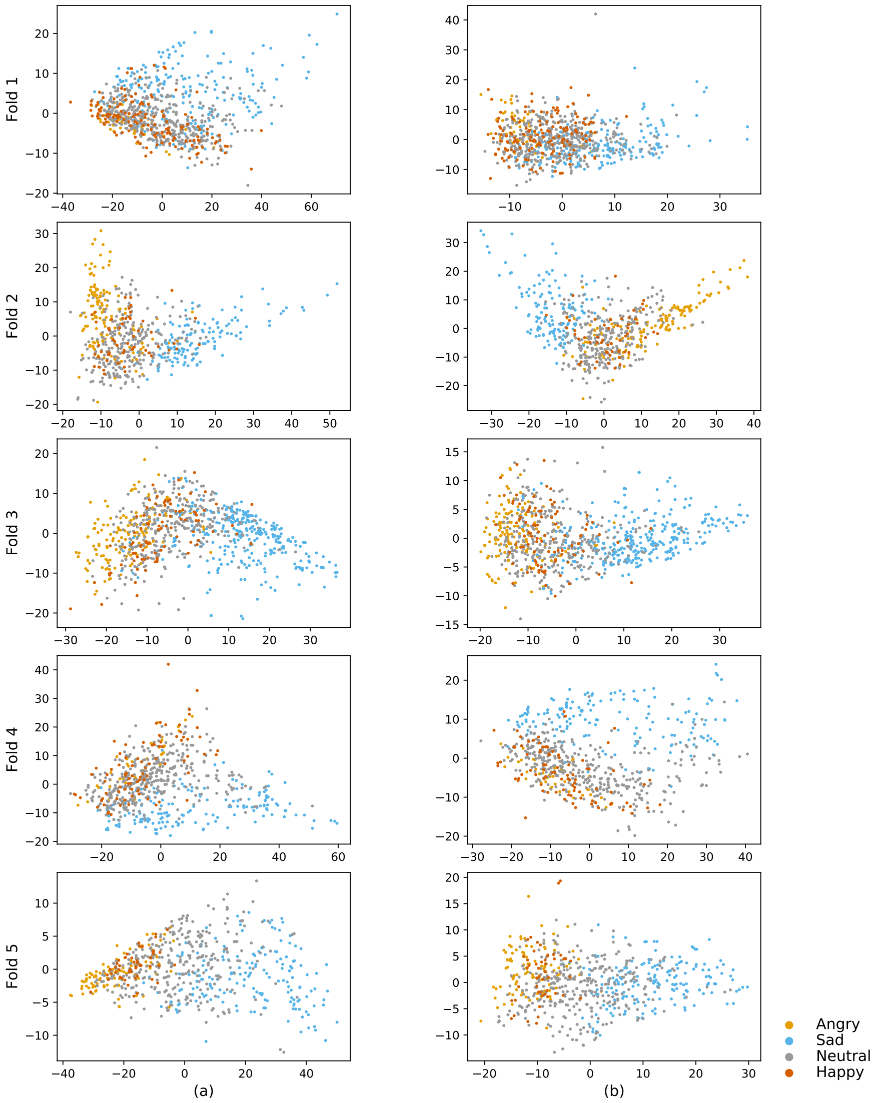


Figure 5.8: Visualisations of the local contexts of the testing sets from the IEMOCAP folds. (a) and (b) are corresponding to the DiCCOSER-CS model with the max-pooling aggregation and with the RMS aggregation respectively.

local contexts learned from the DiCCOSER-CS model with the max pooling aggregation and with the RMS aggregation show similar geometrical properties in this low-dimensional PCA space. In other words, the points from the same test set are spread similarly even when using different aggregation methods, and in particular they form similar shapes in this low-dimensional space but mirrored along the first or second principal component axis.

5.6 Conclusions and future work

In this work, we have proposed a novel end-to-end DNN model for SER that does not consist of any recurrent or fully connected layers. A dedicated dilated causal convolution block is designed to increase the model receptive field while keeping the number of model parameters low. Simulation results firstly indicate that the proposed model with context stacking and the RMS aggregation method achieves the best SER performance among several model variations, which confirms the effectiveness of the novel context stacking structure for SER. Secondly, the simulation results also indicate that the proposed model variations, which only contained about 1/3 of the number of model parameters, outperformed the baseline state-of-the-art CNN-LSTM model. Thirdly, we have shown that the proposed sliding window augmentation and random flipping augmentation methods improve the SER performance for both the baseline model and the proposed model. Fourthly, using log mel-spectrogram features instead of raw audio samples as an input can significantly improve the CNN-LSTM model SER performance, and slightly improves the proposed model SER performance, which indicates that the log mel-spectrogram features can be alternative input features for end-to-end SER. Furthermore, by using either the raw audio samples or the shallow log mel-spectrogram features as an input, the baseline model and the proposed model both achieve better SER performance compared to the SER systems using traditional audio features. Finally, we reported results that allow to interpret the local contexts. We found that 1) the local contexts form clusters that corresponded to their emotion class labels, which indicated that the local contexts tend to learn the affective information not included in the downsampled input sequence, 2) the first principal component of the local context is highly positively or negatively correlated with the arousal of the target emotion, even if we only supervised the model training with the emotion class label.

Future work could investigate using several parallel local networks operating on different input hop lengths, similarly to the idea in [45], which is aiming to learn the local contexts from the input at different levels of compression. In addition, it appears highly attractive to apply the dilated causal convolution to similar tasks where a large receptive field is required, for example, by using the proposed architecture in an autoencoder structure to learn emotion representations from raw speech signals or applying the proposed architecture to other speech-related problems.

Chapter 6

Conclusion

In this thesis, we have identified challenges in speech/audio processing first from the signal processing point of view, in which a processing system needs to be capable to deal with signal artefacts (e.g., reverberation, background noise) while retrieving useful information. Then, from a data-driven modelling point of view, we have argued that the common presence of such artefacts, which are potentially affecting the target signal and causing a shift in its distribution, can be considered jointly with other influencing factors creating a distributional shift challenge, and can be jointly treated in a unified data-driven sequence modelling approach.

We proposed three approaches based on Deep Neural Networks (DNNs) tackling a few common speech/audio processing tasks from a representation learning and sequence modelling point of view, and targeting two fields in binaural Sound Source Localisation (SSL) and Speech Emotion Recognition (SER). Below is a summary of each chapter in response to the sub-objectives set out in the introduction, as well as suggestions for future research.

6.1 Summary

Robust contrastive embeddings for binaural SSL

Sub-objective: To propose a non-linear dimension reduction technique that can preserve sound source proximities from binaural cues.

In Chapter 3, to overcome the generalisation problem of the non-parametric non-linear dimensionality reduction method that relies on the smoothness of the measurement space, we first proposed a DNN framework for supervised dimensionality reduction of binaural cue measurements, followed by a nearest-neighbor regression source localisation. This method defines the embedding similarities in the source latent space and has better binaural SSL performance than the baseline method in known and unknown reverberant conditions and in the small training set condition. In comparison with a feed-forward learning method, our proposed method not only has better visualisation ability, but also has similar or better performance in binaural SSL. Moreover, our proposed method can capture a smooth manifold structure for low data density regions and outperforms the baseline manifold learning method and the feed-forward method in scenarios with few training data.

Second, we also proposed a weakly-supervised embedding, i.e. Weakly supervised Contrastive Embedding (WSCE) that only requires an implicit latent space proximity label. This weakly-supervised embedding can simultaneously estimate the azimuth and elevation angle of the sound source, and is also robust to unknown reverberation. Quantitative experimental results demonstrated that this WSCE has almost similar localisation performance as the supervised method, and it is much better than the traditional unsupervised embedding in the varying reverberation scenario.

Cross-language SER

Sub-objective: To improve the performance of data-driven models in cross-language/cross-corpus SER settings.

To alleviate the performance degradation in cross-language/cross-corpus SER, in Chapter 4, we proposed a transfer learning method that firstly uses the wav2vec 2.0 pre-trained model to transfer a time-domain audio waveform into a contextual embedding space that is shared across different languages, thereby reducing the language variabilities in the speech features. Then, by applying a re-designed Deep-Within-Class Covariance Normalisation (Deep-WCCN), which is adapted to be compatible with DNN training, this Deep-WCCN layer can further reduce within-class variance caused by other factors (e.g. speaker identity, channel variability). Experimental results first show that the proposed method largely increases both within-language and cross-language SER performance compared to the Extended Geneva Minimalistic Acoustic Parameter Set (eGeMAPS) and Emobase feature sets that have been designed for and widely used in SER. Furthermore, an ablation study has shown that Deep-WCCN can reduce the within-class variances which further improves the performance of the proposed DNN model. In contrast, the conventional Within-Class Covariance Normalisation (WCCN) does not show effectiveness on the eGeMAPS and Emobase feature sets with a Random Forest (RF) classifier. Next, we have shown that the proposed transfer learning method exhibits good data efficiency in merging target language data in the fine-tuning process. The speaker-independent SER performance increases for all testing target languages, and a performance gain in Weighted Accuracy (WA) up to 15.6% and in Unweighted Accuracy (UA) up to 13.8% is achieved when only 150 s of target language data is used. Finally, a comparison with recent work in cross-language/cross-corpus SER demonstrates that the proposed method can significantly improve within-language and cross-language SER performance across multiple datasets.

Long sequence modelling and end-to-end SER

> Sub-objective: To design a sequence model that can learn long-term temporal dependencies relevant to end-to-end SER.

In Chapter 5, in order to facilitate long sequence modelling, we propose a novel end-to-end DNN model for SER that does not consist of any recurrent or fully connected layers. A dedicated dilated causal convolution block is designed to increase the model receptive field while keeping the number of model parameters low. Simulation results firstly indicate that the proposed model with context stacking and the RMS aggregation method achieves the best SER performance among several model variations, which confirms the effectiveness of the novel context stacking structure for SER. Secondly, the simulation results also indicate that the proposed model variations, which only contain about 1/3 of the number of model parameters, outperform the baseline state-of-the-art CNN-LSTM model. Thirdly, we have shown that the sliding window augmentation and random flipping augmentation methods improve the SER performance for both the baseline model and the proposed model. Fourthly, using log mel-spectrogram features instead of raw audio samples as an input can significantly improve the CNN-LSTM model SER performance, and slightly improves the proposed model SER performance, which indicates that the log mel-spectrogram features can be alternative input features for end-to-end SER. Furthermore, by using either the raw audio samples or the shallow log mel-spectrogram features as an input, the baseline model and the proposed model both achieve better SER performance compared to the SER systems using traditional audio features. Finally, we report results that allow to interpret the local contexts. We found that 1) the local contexts form clusters that correspond to their emotion class labels, which indicates that the local contexts tend to learn the affective information not included in the downsampled input sequence, 2) the first principal component of the local context is highly positively or negatively correlated with the arousal of the target emotion, even if we only supervised the model training with the emotion class label.

6.2 Suggestions for future research

6.2.1 Future work for binaural SSL

To further increase the generalisation capability of the model introduced in Chapter 3, we hope to learn the Supervised Contrastive Embedding (SCE) and WSCE embeddings with big variety of training data covering more real-life conditions, such as using more Head-related Transfer Functions (HRTFs) recorded at finer azimuth/elevation angles and using Room Impulse Responses (RIRs) from more complex rooms.

Furthermore, we also hope to investigate how to apply the proposed SCE and WSCE in data synthesis. Through retrieving in the embedding space, and combining with a generative model (e.g., an auto-encoder), the embeddings can be used to synthesise binaural features or even audio waveforms to aid data-driven binaural source localisation models.

6.2.2 Future work for cross-language SER

The transfer learning method introduced in Chapter 4 uses the wav2vec 2.0 model that is trained on 53 thousands of hours of speech, showing supreme cross-language generalisation capability in SER. It might be interesting to evaluate the wav2vec 2.0 model pre-trained on an even larger speech dataset and to assess its benefit to cross-language SER performance. A good candidate dataset is proposed in [10] and uses half a million hours of publicly available speech audio in 128 languages.

Moreover, we plan to evaluate the proposed method on more emotive speech datasets, and to fine-tune the model on emotive speech datasets from more languages to create a general data-driven model for SER, and to evaluate the model in the wild (i.e. in real-life conditions).

6.2.3 Future work for end-to-end SER

The novel end-to-end DNN for SER, introduced in Chapter 5, has a local network to capture the fine temporal structure of the input. It will be interesting to investigate the use of several parallel local networks operating on different input hop lengths, similarly to the idea in [45], for learning the local contexts from the input at different levels of compression.

In addition, it appears highly attractive to apply the dilated causal convolution to similar tasks where a large receptive field is required, for example, by using the proposed architecture in an auto-encoder structure to learn emotion representations from raw speech signals or applying the proposed architecture to other speech-related problems.

6.2.4 General future work

Combining physical models with data-driven models

Although this thesis focuses on data-driven deep learning models, combining physical models with a data-driven model is a promising research direction, because training such a hybrid model would reduce the model reliance on large amounts of annotated data and improve the generalisation capability of the model.

Combining data-driven learning models with auditory physical models is particularly interesting, for example the human auditory periphery model [163] where the human listening perception channel (i.e. from the ear canal entrance to the sound processing nerve) is described by a concatenation of several physical models. This type of physical models can be used to extract audio features and can thus be considered as front-end processing for human listening related tasks. In [11], Baby and Verhulst show that using biophysically-inspired features (calculated with a cochlear model) can improve the noise suppression quality of an Artificial Neural Network (ANN) model for speech enhancement.

Therefore, it might be interesting to combine the auditory models used in other speech/audio related tasks such as SSL and SER, in order to take full advantage of the noise suppression capability offered by physical models.

Self-supervised learning for SSL with synthetic data

Self-supervised learning aims to learn intrinsic contextual structure embedded in the speech recordings. The self-supervised learning model, which is designed to predict masked high-level embeddings or future embeddings through neighbourhood embeddings, typically requires a large amount of training data in various acoustic conditions (e.g., with different types of noise and reverberation) [158, 135, 38, 12, 10].

The training of a self-supervised learning model does not need annotations, thus it is easy to acquire data for training purposes. However, it is not always easy to acquire and maintain such big dataset, and moreover the variety of the data is not ensured since an individual can impossible gather as much data as the big IT companies (e.g., Facebook, Google, Apple, etc.). One interesting solution might be to first train the self-supervised model using synthetic data and augmented data, since these data to some extent contain the common speech structures (e.g., phoneme, word, and language structures), and using different types of noise which are essential for the model robustness. In addition, synthetic data provide annotations for "free" since the label can be manually set during data generation. An example using synthetic data to facilitate training in image processing is proposed in [164], which shows that using synthetic data only can achieve similar performance in the wild as using real-life data.

Model size reduction

One problem of the present-day DNN models is that the model size tends to become vary large in order to achieve good performance, and this causes both high computational requirements and high memory requirements. For low-power devices (e.g., smartphones), the need of performant models is high, but the resources for computing the model locally (i.e. on the device) are limited. Therefore, model size reduction is one interesting research direction towards ubiquitous deep learning.

Except model pruning (i.e. removing neurons that are associated with very small weights), there are mainly two ways to reduce the model size, the first being model quantisation, and the second being model distillation [74]. In [101] and [13], the authors approach model size reduction through post-quantisation and distillation, respectively, and show good training efficiency while the resulting model can still maintain relatively high accuracy. These training paradigms might be interesting to be applied to speech/audio processing related tasks such as in SSL and in SER.

Availability of data and materials

The datasets generated and/or analysed during the current study are available in the Interactive Emotional Dyadic Motion Capture (IEMOCAP) repository (https://sail.usc.edu/iemocap), the REmote COLlaborative and Affective (RECOLA) repository (https://diuf.unifr.ch/main/diva/recola), the CAMIL repository (https://team.inria.fr/perception/the-camil-dataset), the VAST repository (http://thevastproject.inria.fr/dataset), the Berlin Database of Emotional Speech (Emo-DB) repository (https://team.inria.fr/perception/the-camil-dataset), the Emotional Speech Dataset (ESD) repository (https://github.com/HLTSingapore/Emotional-Speech-Data), and the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) repository (https://smartlaboratory.org/ravdess/).

However, restrictions apply to the availability of some of these data, which were used under license for the current study, and so are not publicly available. The RECOLA Data are however available from the authors upon reasonable request and with permissions of the SAIL lab at the University of Southern California, and the RECOLA group at the Université de Fribourg.

Acknowledgements

This research work was carried out at the ESAT Laboratory of KU Leuven. The research leading to these results has received funding from the Chinese Scholarship Council (CSC) (grant no. 201707650021), the KU Leuven Internal Funds C2-16-00449 and VES/19/004, and the European Research Council under the European Union's Horizon 2020 research and innovation program / ERC Consolidator Grant: SONORA (no. 773268). This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information.

Bibliography

[1] Aldeneh, Z., and Provost, E. M. Using regional saliency for speech emotion recognition. In *Proc. of 2017 IEEE Int. Conf. Acoust. Speech Signal Process (ICASSP '17)* (Jun 2017), pp. 2741–2745.

[2] Algazi, V., Duda, R., Thompson, D., and Avendano, C. The cipic hrtf database. In *Proc. of IEEE Work. Appl. Signal Process. to Audio Acoust. (WASPAA 2001)* (2001), pp. 99–102.

[3] Ali, R. *Multi-microphone speech enhancement.* PhD thesis, KU Leuven, 2020.

[4] Allen, J. B., and Berkley, D. A. Image method for efficiently simulating small-room acoustics prediction of energy decay in room impulse responses simulated with an image-source model image method for efficiently simulating small-room acoustics. *J. Acoust. Soc. Am. 65* (1979), 943–950.

[5] Alwan, A., Ozun, O., Steurer, P., and Thell, D. Wide Band Speech Coding with LPC. Tech. rep., University of California at Los Angeles, Department of Electrical Engineering, 2002.

[6] Antonello, N. *Solving inverse problems in room acoustics using physical models , sparse regularization and numerical optimization.* PhD thesis, KU Leuven, 2018.

[7] Apple. Hey Siri: An On-device DNN-powered Voice Trigger for Apple's Personal Assistant. Tech. rep., 2017.

[8] Argentieri, S., Danès, P., and Souères, P. A survey on sound source localization in robotics: From binaural to array processing methods. *Comput. Speech Lang. 34*, 1 (2015), 87–112.

[9] Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein GAN. *arXiv:1701.07875* (2017).

[10] Babu, A., Wang, C., Tjandra, A., Lakhotia, K., Xu, Q., Goyal, N., Singh, K., von Platen, P., Saraf, Y., Pino, J., Baevski, A., Conneau, A., and Auli, M. XLS-R: Self-supervised Cross-lingual Speech Representation Learning at Scale. *arXiv:2111.09296* (2021).

[11] Baby, D., and Verhulst, S. Biophysically-inspired features improve the generalizability of neural network-based speech enhancement systems. In *Proc. of Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH* (2018), vol. 2018-Septe, pp. 3264–3268.

[12] Baevski, A., Zhou, H., Mohamed, A., and Auli, M. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proc. of IEEE Conf. Adv. Neural Inf. Process Syst. (NeurIPS 2017)* (California, USA, Dec. 2020), vol. 33, pp. 12449–12460.

[13] Bai, H., Hou, L., Shang, L., Jiang, X., King, I., and Lyu, M. R. Towards Efficient Post-training Quantization of Pre-trained Language Models. *arXiv:2109.15082* (2021).

[14] Bai, S., Kolter, J. Z., and Koltun, V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv:1803.01271* (2018).

[15] Barber, D. *Bayesian Reasoning and Machine Learning.* Cambridge University Press, 2011.

[16] Belkin, M., and Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput. 6*, 15 (2003), 1373–1396.

[17] Bengio, Y., Paiement, J.-F., Vincent, P., Delalleau, O., Le Roux, N., and Ouimet, M. Out-of-sample extensions for LLE, Isomap, MDS, Eigenmaps and Spectral Clustering. In *Proc. of IEEE Conf. Adv. Neural Inf. Process Syst. (NeurIPS 2003)* (Barcelona, Spain, 2003), pp. 177–184.

[18] Bergstra, J., Yamins, D., and Cox, D. D. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *30th Int. Conf. Mach. Learn. ICML 2013* (2013), vol. 28, pp. 115–123.

[19] Bhaykar, M., Yadav, J., and Rao, K. S. Speaker dependent, speaker independent and cross language emotion recognition from speech using GMM and HMM. In *2013 Natl. Conf. Commun. (NCC 2013)* (2013).

[20] Bird, J. J., Faria, D. R., Ekart, A., and Ayrosa, P. P. From Simulation to Reality: CNN Transfer Learning for Scene Classification. In *Proc. of 10th Int. Conf. Intell. Syst. (IS 2020)* (Varna, Bulgaria, Aug 2020), pp. 619–625.

[21] Bird, J. J., Faria, D. R., Premebida, C., Ekart, A., and Ayrosa, P. P. Overcoming Data Scarcity in Speaker Identification: Dataset Augmentation with Synthetic MFCCs via Character-level RNN. In *Proc. of Int. Conf. Auton. Robot Syst. Compet. (ICARSC 2020)* (Apr. 2020), pp. 146–151.

[22] Blauert, J. *Spatial Hearing: The Psychophysics of Human Sound Localization.* MIT Press, 1997.

[23] Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L., Goel, K., Goodman, N., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y., Ruiz, C., Ryan, J., Ré, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K., Tamkin, A., Taori, R., Thomas, A. W., Tramèr, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. On the Opportunities and Risks of Foundation Models. *arXiv:2108.07258* (2021).

[24] Boser, B. E., Guyon, I. M., and Vapnik, V. N. A training algorithm for optimal margin classifiers. In *Proc. of the Fifth Annual Workshop on Computational Learning Theory (COLT '92)* (1992), p. 144–152.

[25] Breiman, L. Bagging predictors. *Mach. Learn. 24*, 2 (Aug 1996), 123–140.

[26] Breiman, L. Random Forests. *Mach. Learn. 45*, 1 (Oct. 2001), 5–32.

[27] Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., and Shah, R. Signature Verification Using a "Siamese" Time Delay Neural Network.

In *Proc. of IEEE Conf. Adv. Neural Inf. Process Syst. (NeurIPS 1993)* (Colorado, USA, 1993), vol. 6, pp. 737–744.

[28] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In *Proc. of IEEE Conf. Adv. Neural Inf. Process Syst. (NeurIPS 2020)* (Online, Dec. 2020).

[29] Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W., and Weiss, B. A database of German emotional speech. In *Proc. of 9th Eur. Conf. Speech Commun. Technol. (EUROSPEECH)* (2005), pp. 1517–1520.

[30] Busso, C., Bulut, M., Lee, C. C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S., and Narayanan, S. S. IEMOCAP: Interactive emotional dyadic motion capture database. *Lang. Resour. Eval. 42*, 4 (2008), 335–359.

[31] Castillo, J. C., Castro-González, Á., Alonso-Martín, F., Fernández-Caballero, A., and Salichs, M. Á. Emotion detection and regulation from personal assistant robot in smart environment. In *Intell. Syst. Ref. Libr.*, vol. 132. Springer, Cham, 2018, pp. 179–195.

[32] Chahal, K. S., Grover, M. S., Dey, K., and Shah, R. R. A Hitchhiker's Guide On Distributed Training Of Deep Neural Networks. *J. Parallel Distrib. Comput. 137* (Mar 2020), 65–76.

[33] Chen, M., He, X., Yang, J., and Zhang, H. 3-D Convolutional Recurrent Neural Networks with Attention Model for Speech Emotion Recognition. *IEEE Signal Process. Lett. 25*, 10 (Oct. 2018), 1440–1444.

[34] Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proc. of the 2014 Conf. Empir. Methods in Nat. Lang. Process. (EMNLP)* (Oct. 2014), pp. 1724–1734.

[35] Choi, K., Fazekas, G., Sandler, M., and Kim, J. Auralisation of Deep Convolutional Neural Networks : Listening To Learned Features. *Int. Symp. on Music Information Retrieval (ISMIR 2015)* (2015), 4–5.

[36] Chopra, S., Hadsell, R., and LeCun, Y. Learning a Similarity Metric Discriminatively, with Application to Face Verification. In *Proc.*

IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR 2005) (California, USA, 2005), vol. 1, pp. 539–546.

[37] Chung, F. R. K. *Spectral Graph Theory.* Providence, RI: American Mathematical Society, 1997.

[38] Conneau, A., Baevski, A., Collobert, R., Mohamed, A., and Auli, M. Unsupervised Cross-lingual Representation Learning for Speech Recognition. *arXiv:2006.13979* (2020).

[39] Datum, M. S., Palmieri, F., and Moiseff, A. An artificial neural network for sound localization using binaural cues. *J. Acoust. Soc. Am. 100*, 1 (1996), 372–383.

[40] Deacon, T. W. *The symbolic species: The coevolution of language and the brain.* 1997.

[41] Dehak, N., Kenny, P. J., Dehak, R., Dumouchel, P., and Ouellet, P. Front-end factor analysis for speaker verification. *IEEE Trans. Audio, Speech Lang. Process. 19*, 4 (2011), 788–798.

[42] Deleforge, A., Forbes, F., and Horaud, R. Acoustic space learning for sound source separation and localization on binaural manifolds. *Int. J. Neural Syst. 25*, 1 (2015).

[43] Deleforge, A., and Horaud, R. 2D sound-source localization on the binaural manifold. In *IEEE Int. Work. Mach. Learn. Signal Process (MLSP 2012)* (Santander, Spain, 2012).

[44] Desplanques, B., and Demuynck, K. Cross-lingual speech emotion recognition through factor analysis. In *Proc. Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH 2018)* (Sep. 2018), pp. 3648–3652.

[45] Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., and Sutskever, I. Jukebox: A Generative Model for Music. *arXiv:2005.00341* (2020).

[46] Dionisio, J. D. N., Burns, W. G., and Gilbert, R. 3D virtual worlds and the metaverse: Current status and future possibilities. *ACM Comput. Surv. 45*, 3 (2013).

[47] Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. In *23rd Conf. Learn. Theory (COLT 2010)* (2010), vol. 12, pp. 257–269.

[48] Eghbal-zadeh, H., Dorfer, M., and Widmer, G. Deep Within-Class Covariance Analysis for Robust Audio Representation Learning. *arXiv:1711.04022* (2017).

[49] Ekman, P. Facial Expressions of Emotion: New Findings, New Questions. *Psychol. Sci. 3*, 1 (Apr. 1992), 34–38.

[50] El Ayadi, M., Kamel, M. S., and Karray, F. Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognit. 44*, 3 (2011), 572–587.

[51] Eyben, F., Scherer, K. R., Schuller, B. W., Sundberg, J., André, E., Busso, C., Devillers, L. Y., Epps, J., Laukka, P., Narayanan, S. S., and Truong, K. P. The geneva minimalistic acoustic parameter set (gemaps) for voice research and affective comput. *IEEE Trans. Affect. Comput. 7*, 2 (2016), 190–202.

[52] Eyben, F., Wöllmer, M., and Schuller, B. OpenEAR - Introducing the Munich open-source emotion and affect recognition toolkit. In *Proc. of 3rd Int. Conf. Affect. Comput. Intell. Interact. Work. (ACII 2009)* (2009).

[53] Ezzaidi, H., Rouat, J., and O'Shaughnessy, D. Towards combining pitch and MFCC for speaker identification systems. In *Proc. 7th Eur. Conf. Speech Commun. Technol. (EUROSPEECH 2001)* (2001), pp. 2825–2828.

[54] Fano, R. M. The Information Theory Point of View in Speech Communication. *J. Acoust. Soc. Am. 22*, 6 (1950), 691–696.

[55] Farmani, M., Pedersen, M. S., and Jensen, J. Sound source localization for hearing aid applications using wireless microphones. In *IEEE Sens. Array Multichannel Signal Process. Work. (SAM 2018)* (Sheffield, UK, Aug. 2018), pp. 455–459.

[56] France, D. J., and Shiavi, R. G. Acoustical properties of speech as indicators of depression and suicidal risk. *IEEE Trans. Biomed. Eng. 47*, 7 (2000), 829–837.

[57] Freund, Y., and Schapire, R. E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci. 55*, 1 (aug 1997), 119–139.

[58] Friedman, J. H. Greedy function approximation: A gradient boosting machine. *Annals of statistics 29*, 5 (Oct. 2001), 1189–1232.

[59] Gales, M. J. F. Model-based techniques for noise robust speech recognition. *Univ. Cambridge Phd Diss.*, Sep. (1995).

[60] Gardner, W. G., and Martin, K. D. HRTF measurements of a KEMAR. *J. Acoust. Soc. Am. 97*, 6 (1995), 3907–3908.

[61] Garofolo, J., Lamel, L., Fisher, W., Fiscus, J., Pallett, D., Dahlgren, N., and Zue, V. TIMIT Acoustic-Phonetic Continuous Speech Corpus.

[62] Gaultier, C., Kataria, S., and Deleforge, A. VAST: The Virtual Acoustic Space Traveler Dataset. In *Proc. of Int. Conf. Latent Var. Anal. Signal Sep.* (2017), pp. 68–79.

[63] Gerczuk, M., Amiriparian, S., Ottl, S., and Schuller, B. EmoNet: A Transfer Learning Framework for Multi-Corpus Speech Emotion Recognition. *IEEE Trans. Affect. Comput.* (2021).

[64] Gideon, J., McInnis, M. G., and Provost, E. M. Improving cross-corpus speech emotion recognition with adversarial discriminative domain generalization (ADDoG). *IEEE Trans. Affect. Comput. 12*, 4 (Mar. 2021), 1055–1068.

[65] Glorot, X., Bordes, A., and Bengio, Y. Deep sparse rectifier neural networks. *J. Mach. Learn. Res. 15* (2011), 315–323.

[66] Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning.* MIT Press, 2016.

[67] Grant, D., McLane, I., and West, J. Rapid and Scalable COVID-19 Screening using Speech, Breath, and Cough Recordings. In *2021 IEEE EMBS Int. Conf. Biomed. Health Inform. (BHI)* (aug 2021), pp. 1–6.

[68] Hadsell, R., Chopra, S., and LeCun, Y. Dimensionality reduction by learning an invariant mapping. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR 2006)* (2006), vol. 2, pp. 1735–1742.

[69] Hafen, R. P., and Henry, M. J. Speech information retrieval: A review. *Multimed. Syst. 18*, 6 (2012), 499–518.

[70] Hatch, A. O., and Stolcke, A. Generalized linear kernels for one-versus-all classification: Application to speaker recognition. In *Proc. of 2006 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '06)* (2006), vol. 5.

[71] He, C., Ma, M., and Wang, P. Extract interpretability-accuracy balanced rules from artificial neural networks: A review. *Neurocomputing 387* (Apr. 2020), 346–358.

[72] He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR 2016)* (2016), vol. 2016-Dec, pp. 770–778.

[73] Hermus, K., Wambacq, P., and Van hamme, H. A review of signal subspace speech enhancement and its application to noise robust speech recognition. *EURASIP J. Adv. Signal Process. 2007* (2006).

[74] Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv:1503.02531* (2015).

[75] Hochreiter, S., and Schmidhuber, J. Long Short-Term Memory. *Neural Comput. 9*, 8 (Nov. 1997), 1735–1780.

[76] Hornik, K., Stinchcombe, M., and White, H. Multilayer feedforward networks are universal approximators. *Neural Networks 2*, 5 (1989), 359–366.

[77] Houben, M., Van, W., Noortgate, D., and Kuppens, P. The Relation Between Short-Term Emotion Dynamics and Psychological Well-Being: A Meta-Analysis. *Psychological Bulletin 141*, 4 (2015), 901–930.

[78] Hozjan, V., and Kačič, Z. Context-independent multilingual emotion recognition from speech signals. *Int. J. Speech Technol. 6*, 3 (2003), 311–320.

[79] Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., and Mohamed, A. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units. *IEEE/ACM Trans. Audio, Speech, Language Process. 29* (2021), 3451–3460.

[80] Ioffe, S., and Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proc. of the 32nd Int. Conf. on Machine Learning* (2015), vol. 37, pp. 448–456.

[81] Ji, S., Pan, S., Cambria, E., Marttinen, P., and Yu, P. S. A Survey on Knowledge Graphs: Representation, Acquisition, and Applications. *IEEE Trans. Neural Networks Learn. Syst. 33*, 2 (2022), 494–514.

[82] Jia, Y., Zhang, Y., Weiss, R. J., Wang, Q., Shen, J., Ren, F., Chen, Z., Nguyen, P., Pang, R., Moreno, I. L., and Wu, Y. Transfer learning from speaker verification to multispeaker text-to-speech synthesis. In *Proc. of IEEE Conf. Adv. Neural Inf. Process Syst. (NeurIPS 2018)* (Quebec, Canada, 2018), vol. 2018-Dec, pp. 4480–4490.

[83] Jiang, W., Wang, Z., Jin, J. S., Han, X., and Li, C. Speech emotion recognition with heterogeneous feature unification of deep neural network. *Sensors 19*, 12 (Jun 2019), 2730.

[84] Kahneman, D., Krueger, A. B., Schkade, D., Schwarz, N., and Stone, A. Toward national well-being accounts. In *Am. Econ. Rev.* (May. 2004), vol. 94, pp. 429–434.

[85] Karthik, G. R., and Ghosh, P. K. Binaural speech source localization using template matching of interaural time difference patterns. In *Proc.*

of 2018 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '18) (2018), pp. 5164–5168.

[86] Keyrouz, F., and Diepold, K. Binaural Source Localization and Spatial Audio Reproduction for Telepresence Applications. *Presence Teleoperators Virtual Environ. 16*, 5 (2007), 509–522.

[87] Kingma, D. P., and Ba, J. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980* (2015).

[88] Krueger, A. B., and Stone, A. A. Progress in measuring subjective well-being. *Science 346*, 6205 (2014), 42–43.

[89] Kuppens, P., Oravecz, Z., and Tuerlinckx, F. Feelings Change: Accounting for Individual Differences in the Temporal Dynamics of Affect. *J. Pers. Soc. Psychol. 99*, 6 (2010), 1042–1060.

[90] Latif, S., Qadir, J., and Bilal, M. Unsupervised Adversarial Domain Adaptation for Cross-Lingual Speech Emotion Recognition. In *Proc. of 8th Int. Conf. Affect. Comput. Intell. Interact. (ACII 2019)* (Sep. 2019).

[91] Latif, S., Rana, R., Younis, S., Qadir, J., and Epps, J. Transfer learning for improving speech emotion classification accuracy. In *Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH* (2018), vol. 2018-Septe, pp. 257–261.

[92] Laufer, B., Talmon, R., and Gannot, S. Relative transfer function modeling for supervised source localization. In *Proc. of IEEE Work. Appl. Signal Process. to Audio Acoust. (WASPAA 2013)* (2013), pp. 1–4.

[93] Laufer-Goldshtein, B., Talmon, R., and Gannot, S. A Study on Manifolds of Acoustic Responses. In *Proc. of Int. Conf. Latent Var. Anal. Signal Sep. (LVA/ICA 2015)* (2015), pp. 203–210.

[94] LeCun, Y. Generalization and Network Design Strategies. *Connectionism in perspective 19* (1989), 143–155.

[95] Li, M., Yang, B., Levy, J., Stolcke, A., Rozgic, V., Matsoukas, S., Papayiannis, C., Bone, D., and Wang, C. Contrastive Unsupervised Learning for Speech Emotion Recognition. In *Proc. of 2021 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '21)* (May. 2021), pp. 6329–6333.

[96] Li, R., Wu, Z., Jia, J., Bu, Y., Zhao, S., and Meng, H. Towards discriminative representation learning for speech emotion recognition. In *Int. Jt. Conf. Artif. Intell. (IJCAI 2019)* (Aug. 2019), pp. 5060–5066.

[97] Li, R., Wu, Z., Jia, J., Zhao, S., and Meng, H. Dilated Residual Network with Multi-head Self-attention for Speech Emotion Recognition. In *Proc. of 2019 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '19)* (May 2019), pp. 6675–6679.

[98] Li, X., Girin, L., Horaud, R., and Gannot, S. Estimation of the Direct-Path Relative Transfer Function for Supervised Sound-Source Localization. *IEEE/ACM Trans. Audio Speech Lang. Process. 24*, 11 (2016), 2171–2186.

[99] Liang, J., Chen, S., Zhao, J., Jin, Q., Liu, H., and Lu, L. Cross-culture Multimodal Emotion Recognition with Adversarial Learning. In *Proc. of 2019 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '19)* (May. 2019), pp. 4000–4004.

[100] Lin, L. I.-K. A Concordance Correlation Coefficient to Evaluate Reproducibility. *Biometrics 45*, 1 (Mar 1989), 255.

[101] Liu, Z., Wang, Y., Han, K., Ma, S., and Gao, W. Post-Training Quantization for Vision Transformer. In *Proc. of IEEE Conf. Adv. Neural Inf. Process Syst. (NeurIPS 2021)* (Online, Dec 2021), vol. 34.

[102] Liu, Z., Wu, M., Cao, W., Mao, J., Xu, J., and Tan, G. Speech emotion recognition based on feature selection and extreme learning machine decision tree. *Neuro Comput. 273* (2018), 271–280.

[103] Livingstone, S. R., and Russo, F. A. The ryerson audio-visual database of emotional speech and song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in north American english. *PLoS One 13*, 5 (May. 2018), e0196391.

[104] Ma, N., Gonzalez, J. A., and Brown, G. J. Robust Binaural Localization of a Target Sound Source by Combining Spectral Source Models and Deep Neural Networks. *IEEE/ACM Trans. Audio Speech Lang. Process. 26*, 11 (Nov. 2018), 2122–2131.

[105] Ma, N., May, T., and Brown, G. J. Exploiting Deep Neural Networks and Head Movements for Robust Binaural Localization of Multiple Sources in Reverberant Environments. *IEEE/ACM Trans. Audio Speech Lang. Process. 25*, 12 (2017), 2444–2453.

[106] Mandel, M., Weiss, R., and Ellis, D. Model-based expectation-maximization source separation and localization. *IEEE Trans. Audio, Speech, Lang. Process. 18*, 2 (2010), 382–394.

[107] Markidis, S., Chien, S. W. D., Laure, E., Peng, I. B., and Vetter, J. S. NVIDIA tensor core programmability, performance & precision. In *Proc. of 2018 IEEE 32nd Int. Parallel Distrib. Process. Symp. Work. (IPDPSW 2018)* (Aug. 2018), pp. 522–531.

[108] May, T., Van De Par, S., and Kohlrausch, A. A probabilistic model for robust localization based on a binaural auditory front-end. *IEEE Trans. Audio, Speech Lang. Process. 19*, 1 (2011), 1–13.

[109] Mehrabian, A. *Basic dimensions for a general psychological theory.* Oelgeschlager, Gunn & Hain, 1980.

[110] Mikolov, T., Chen, K., Corrado, G., and Dean, J. Efficient estimation of word representations in vector space. In *Proc. of 1st Int. Conf. Learn. Represent. (ICLR 2013)* (2013).

[111] Mundici, D. *Logic: a Brief Course.* Springer, 2012.

[112] Neumann, M., and Thang Vu, N. G. CRoss-lingual and Multilingual Speech Emotion Recognition on English and French. In *Proc. of 2018 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '18)* (Sep. 2018), pp. 5769–5773.

[113] Opochinsky, R., Laufer-Goldshtein, B., Gannot, S., and Chechik, G. Deep ranking-based sound source localization. In *Proc. of IEEE Work. Appl. Signal Process. to Audio Acoust. (WASPAA 2019)* (New Paltz, NY, USA, 2019), pp. 283–287.

[114] Pak, J., and Shin, J. W. Sound Localization Based on Phase Difference Enhancement Using Deep Neural Networks. *IEEE/ACM Trans. Audio Speech Lang. Process. 27*, 8 (Aug. 2019), 1335–1345.

[115] Parry, J., Palaz, D., Clarke, G., Lecomte, P., Mead, R., Berger, M., and Hofer, G. Analysis of deep learning architectures for cross-corpus speech emotion recognition. In *Proc. of Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH 2019)* (Sep. 2019), pp. 1656–1660.

[116] Pepino, L., Riera, P., and Ferrer, L. Emotion Recognition from Speech Using wav2vec 2.0 Embeddings. In *Proc. of Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH 2021)* (Sep. 2021), pp. 3400–3404.

[117] Pieraccini, R. From AUDREY to Siri: Is Speech recognition a solved problem? *Int. Comput. Sci. Inst. Berkeley* (2012).

[118] Quinlan, J. Induction of decision trees. *Mach. Learn. 1* (1985), 81–106.

[119] Rabiner, L. R., and Gold, B. Theory and Application of Digital Signal Processing. *IEEE Trans. Acoust. 23*, 4 (1975), 394–395.

[120] Raspaud, M., Viste, H., and Evangelista, G. Binaural source localization by joint estimation of ILD and ITD. *IEEE Trans. Audio, Speech Lang. Process. 18*, 1 (2010), 68–77.

[121] Rehman, A., Liu, Z. T., Li, D. Y., and Wu, B. H. Cross-Corpus Speech Emotion Recognition Based on Hybrid Neural Networks. In *Proc. of Chinese Control Conf. (CCC 2020)* (Jul. 2020), pp. 7464–7468.

[122] Reynolds, D. A., and Rose, R. C. Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. *IEEE Trans. Speech Audio Process. 3*, 1 (1995), 72–83.

[123] Ringeval, F., Eyben, F., Kroupi, E., Yuce, A., Thiran, J. P., Ebrahimi, T., Lalanne, D., and Schuller, B. Prediction of asynchronous dimensional emotion ratings from audiovisual and physiological data. *Pattern Recognit. Lett. 66* (2015), 22–30.

[124] Ringeval, F., Schuller, B., Valstar, M., Cowie, R., and Pantic, M. AVEC 2015 - the 5th international audio/visual emotion challenge and workshop. In *Proc. of the 5th Audio/Visual Emotion Challenge and Workshop (AV+EC'15)* (2015), pp. 3–8.

[125] Ringeval, F., Sonderegger, A., Sauer, J., and Lalanne, D. Introducing the RECOLA multimodal corpus of remote collaborative and affective interactions. In *Proc. of IEEE Int. Conf. and Workshops on Automatic Face and Gesture Recognition (FG)* (2013), pp. 22–26.

[126] Risoud, M., Hanson, J. N., Gauvrit, F., Renard, C., Lemesre, P. E., Bonne, N. X., and Vincent, C. Sound source localization. *Eur. Ann. Otorhinolaryngol. Head Neck Dis. 135*, 4 (2018), 259–264.

[127] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning representations by back-propagating errors. *Nature 323*, 6088 (1986), 533–536.

[128] Russell, J. A. Core Affect and the Psychological Construction of Emotion. *Psychol. Rev. 110*, 1 (2003), 145–172.

[129] Sahu, S., Mitra, V., Seneviratne, N., and Espy-Wilson, C. Multi-modal learning for speech emotion recognition: An analysis and comparison of ASR outputs with ground truth transcription. In *Proc. Annu. Conf. Int. Speech Commun. Assoc (INTERSPEECH 2019)* (Sep. 2019), pp. 3302–3306.

[130] Sarma, M., Ghahremani, P., Povey, D., Goel, N. K., Sarma, K. K., and Dehak, N. Emotion identification from raw speech signals using DNNs. In *Proc. of Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH 2018)* (Sep. 2018), pp. 3097–3101.

[131] Satt, A., Rozenberg, S., and Hoory, R. Efficient emotion recognition from speech using deep learning on spectrograms. In *Proc. of Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH 2017)* (Aug. 2017), pp. 1089–1093.

[132] Saxena, A., Khanna, A., and Gupta, D. Emotion Recognition and Detection Methods: A Comprehensive Survey. *J. Artif. Intell. Syst. 2*, 1 (2020), 53–79.

[133] Schimmel, S. M., Muller, M. F., and Dillier, N. A fast and accurate shoebox room acoustics simulator. In *Proc. of 2009 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '09)* (2009), pp. 241–244.

[134] Schmidt, R. O. Multiple Emitter Location and Signal Parameter Estimation. *IEEE Trans. Antennas Propag. 34*, 3 (1986), 276–280.

[135] Schneider, S., Baevski, A., Collobert, R., and Auli, M. wav2vec: Unsupervised pre-training for speech recognition. In *Proc. of Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH 2019)* (Sep. 2019), pp. 3465–3469.

[136] Schuller, B., Rigol, G., and Lang, M. Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine - Belief network architecture. In *Proc. of 2004 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '04)* (2004), vol. 1.

[137] Schuller, B., Vlasenko, B., Eyben, F., Rigoll, G., and Wendemuth, A. Acoustic emotion recognition: A benchmark comparison of performances. In *Proc. of 2009 IEEE Work. Autom. Speech Recognit. Understanding (ASRU 2009)* (2009), pp. 552–557.

[138] Schuller, B. W. Speech emotion recognition: Two decades in a nutshell, benchmarks, and ongoing trends. *Commun. ACM 61*, 5 (2018), 90–99.

[139] Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., Saurous, R. A., Agiomvrgiannakis, Y., and Wu, Y. Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions. In *Proc. of IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '18)* (Apr. 2018), pp. 4779–4783.

[140] Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., and Khudanpur, S. X-Vectors: Robust DNN Embeddings for Speaker Recognition. In *Proc. of 2018 IEEE Int. Conf. Acoust. Speech Signal Process (ICASSP '18)* (Apr. 2018), pp. 5329–5333.

[141] Song, P. Transfer linear subspace learning for cross-corpus speech emotion recognition. *IEEE Trans. Affect. Comput. 10*, 2 (Apr. 2019), 265–275.

[142] Sotelo, J., Mehri, S., Kumar, K., Santos, J. F., Kastner, K., Courville, A., and Bengio, Y. Char2wav: End-to-end Speech Synthesis. In *Proc. of 5th Int. Conf. Learn. Represent. (ICLR 2017)* (2017).

[143] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res. 15* (2014), 1929–1958.

[144] Suykens, J., Van Gestel, T., De Brabanter, J., Moor, D., and Vandewalle, J. *Least Squares Support Vector Machines.* World Scientific Publishing Company, 2002.

[145] Swain, M., Routray, A., and Kabisatpathy, P. Databases, features and classifiers for speech emotion recognition: a review. *Int. J. Speech Technol. 21*, 1 (2018), 93–120.

[146] Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR 2014)* (2014), pp. 1701–1708.

[147] Tang, D., Kuppens, P., Geurts, L., and van Waterschoot, T. Adieu recurrence? End-to-end speech emotion recognition using a context stacking dilated convolutional network. In *Proc. of the 28th Eur. Signal Process. Conf. (EUSIPCO '28)* (Amsterdam, Netherlands (Virtual), 2020).

[148] Tang, D., Kuppens, P., Geurts, L., and van Waterschoot, T. End-to-end speech emotion recognition using a novel context-stacking dilated convolution neural network. *Eurasip J. Audio, Speech, Music Process. 2021*, 1 (Dec. 2021), 1–16.

[149] Tang, D., Taseska, M., and van Waterschoot, T. Supervised contrastive embeddings for binaural source localization. In *Proc. of IEEE Work. Appl. Signal Process. to Audio Acoust. (WASPAA 2019)* (New Paltz, NY, USA, Oct. 2019), pp. 358–362.

[150] Taseska, M., and van Waterschoot, T. On spectral embeddings for supervised binaural source localization. In *Proc. of the 27th Eur. Signal Process. Conf. (EUSIPCO '27)* (A Coruña, Spain, 2019), pp. 1–5.

[151] Tieleman, T., and Hinton, G. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. In *COURSERA Neural networks Mach. Learn.* (2012), pp. 26–31.

[152] Tonin, F., Patrinos, P., and Suykens, J. A. Unsupervised learning of disentangled representations in deep restricted kernel machines with orthogonality constraints. *Neural Networks 142* (oct 2021), 661–679.

[153] Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., and Zafeiriou, S. Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In *Proc. of 2016 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '16)* (2016), pp. 5200–5204.

[154] Tzirakis, P., Trigeorgis, G., Nicolaou, M. A., Schuller, B. W., and Zafeiriou, S. End-to-end multimodal emotion recognition using deep neural networks. *IEEE J. Sel. Topics Signal Process. 11*, 8 (2017), 1301–1309.

[155] Tzirakis, P., Zhang, J., and Schuller, B. W. End-to-end speech emotion recognition using deep neural networks. In *Proc. of 2018 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '18)* (2018), pp. 5089–5093.

[156] Valstar, M., Gratch, J., Schuller, B., Ringevaly, F., Lalanne, D., Torres, M. T., Scherer, S., Stratou, G., Cowie, R., and Panticz, M. AVEC 2016 - Depression, mood, and emotion recognition workshop and challenge. In *Proc. of the 6th Audio/Visual Emotion Challenge and Workshop (AV+EC'16)* (2016), pp. 3–10.

[157] van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. Wavenet: A generative model for raw audio. In *Int. Symp. on Comput. Archt. (ISCA)* (2016).

[158] van den Oord, A., Li, Y., and Vinyals, O. Representation Learning with Contrastive Predictive Coding. *arXiv:1807.03748* (2018).

[159] van den Oord, A., Vinyals, O., and Kavukcuoglu, K. Neural Discrete Representation Learning. In *Proc. of IEEE Conf. Adv. Neural Inf. Process Syst. (NeurIPS 2017)* (California, USA, 2017), vol. 30, pp. 6306–6315.

[160] Vaswani, A., Brain, G., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention Is All You Need. In *Proc. of IEEE Conf. Adv. Neural Inf. Process Syst. (NeurIPS 2017)* (California, USA, 2017), pp. 5998–6008.

[161] Vecchiotti, P., Ma, N., Squartini, S., and Brown, G. J. End-to-end binaural sound localisation from the raw waveform. In *Proc. of 2019 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '19)* (2019), pp. 451–455.

[162] Verduyn, P., Van Mechelen, I., Kross, E., Chezzi, C., and Van Bever, F. The relationship between self-distancing and the duration of negative and positive emotional experiences in daily life. *Emotion 12*, 6 (2012), 1248–1263.

[163] Verhulst, S., Altoè, A., and Vasilkov, V. Computational modeling of the human auditory periphery: Auditory-nerve responses, evoked potentials and hearing loss. *Hear. Res. 360* (Mar. 2018), 55–75.

[164] Wood, E., Baltrušaitis, T., Hewitt, C., Dziadzio, S., Cashman, T. J., and Shotton, J. Fake it till you make it: face analysis in the wild using synthetic data alone. In *Proc. of 2021 IEEE/CVF Int. Conf. Comput. Vis. (ICCV 2021)* (2021), pp. 3661–3671.

[165] Woodruff, J., and Wang, D. L. Binaural localization of multiple sources in reverberant and noisy environments. *IEEE Trans. Audio, Speech Lang. Process. 20*, 5 (Jul. 2012), 1503–1512.

[166] Xiao, Z., Wu, D., Zhang, X., and Tao, Z. Speech emotion recognition cross language families: Mandarin vs. Western Languages. In *Proc. of 2016 IEEE Int. Conf. Prog. Informatics Comput. (PIC 2016)* (Jun. 2017), pp. 253–257.

[167] Yalta, N., Nakadai, K., and Ogata, T. Sound source localization using deep learning models. *J. Robot. Mechatronics 29*, 1 (Feb. 2017), 37–48.

[168] Yang, B., Li, X., and Liu, H. Supervised direct-path relative transfer function learning for binaural sound source localization. In *Proc. of 2021 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '21)* (2021), pp. 825–829.

[169] Yu, F., Koltun, V., and Funkhouser, T. Dilated residual networks. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR 2017)* (Jan. 2017), pp. 636–644.

[170] Zeiler, M. D., and Fergus, R. Visualizing and understanding convolutional networks. In *Comput. Vision (ECCV 2014)* (2014), pp. 818–833.

[171] Zeng, W., Ren, X., Su, T., Wang, H., Liao, Y., Wang, Z., Jiang, X., Yang, Z., Wang, K. M., Zhang, X., Li, C., Gong, Z., Yao, Y., Huang, X., Wang, J., Yu, J., Guo, Q., Yu, Y., Zhang, Y., Wang, J., Tao, H., Yan, D., Yi, Z., Peng, F., Jiang, F., Zhang, H., Deng, L., Zhang, Y., Lin, Z., Zhang, C., Zhang, S., Guo, M., Gu, S., Fan, G., Wang, Y., Jin, X., Liu, Q., and Tian, Y. Pangu-α: Large-scale autoregressive pretrained chinese language models with auto-parallel computation. *arXiv:2104.12369* (2021).

[172] Zhang, W., and Song, P. Transfer sparse discriminant subspace learning for cross-corpus speech emotion recognition. *IEEE/ACM Trans. Audio Speech Lang. Process. 28* (2020), 307–318.

[173] Zhang, Z., Wu, B., and Schuller, B. Attention-augmented End-to-end Multi-task Learning for Emotion Prediction from Speech. In *Proc. of 2019 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '19)* (May 2019), pp. 6705–6709.

[174] Zhao, Z., Bao, Z., Zhang, Z., Cummins, N., Wang, H., and Schuller, B. Attention-enhanced connectionist temporal classification for discrete speech emotion recognition. In *Proc. of Annu. Conf. Int. Speech Commun. Assoc (INTERSPEECH 2019)* (Sep. 2019), pp. 206–210.

[175] Zhou, K., Sisman, B., Liu, R., and Li, H. Seen and unseen emotional style transfer for voice conversion with a new emotional speech dataset. In *Proc. of 2021 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP '21)* (Jun. 2021), pp. 920–924.

Curriculum Vitae

Duowei Tang obtained a joint B.Sc. degree in Engineering Technology (Electronics Engineering) from KU Leuven, Belgium, and Southwest Jiaotong University, China, in 2014. In 2015 and 2016, he received two M.Sc. degrees in Electronics Engineering and in Artificial Intelligence, respectively, from KU Leuven, Belgium. During the year after his M.Sc., he worked as a researcher focusing on instrument recognition and sound event detection with machine learning, in the "m-sense" project hosted in the research group of e-Media, in KU Leuven, Belgium. From 2017 to 2021, he has been a doctoral researcher at the Electrical Engineering Department (ESAT), in the research group of Stadius Center for Dynamical Systems, Signal Processing, and Data Analytics (STADIUS) under the supervision of Prof. Toon van Waterschoot, in KU Leuven, Belgium. His research is focusing on speech and audio processing with deep learning, specifically on designing speech emotion recognition, and sound source localisation algorithms. He served as a reviewer for the journal of Speech Communication.

List of publications

International Journal Paper

- **Tang, D.**, Kuppens, P., Geurts, L. and van Waterschoot, T., End-to-end speech emotion recognition using a novel context-stacking dilated convolution neural network. *EURASIP Journal on Audio, Speech, and Music Processing*, 2021(18), 2021.

International Conference Papers

- **Tang, D.**, Taseska, M. and van Waterschoot, T., Supervised Contrastive Embeddings for Binaural Source Localization. *In Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2019, pp. 358-362, doi: 10.1109/WASPAA.2019.8937177.
- **Tang, D.**, Kuppens, P., Geurts, L. and van Waterschoot, T., Adieu recurrence? End-to-end speech emotion recognition using a context stacking dilated convolutional network. *In Proc. 28th European Signal Processing Conference (EUSIPCO)*, 2021 January.

Papers under review

- **Tang, D.**, Taseska, M., and van Waterschoot, T., Towards learning robust contrastive embeddings for binaural sound source localisation.

Papers to be submitted

- **Tang, D.**, Kuppens, P., Geurts, L. and van Waterschoot, T., End-to-end transfer learning for speaker-independent cross-language speech emotion recognition.

FACULTY OF ENGINEERING TECHNOLOGY
DEPARTMENT OF ELECTRICAL ENGINEERING
STADIUS-ESAT
Kasteelpark Arenberg 10, bus 2446
B-3001 Leuven
duowei.tang@kuleuven.be
https://invincibleo.github.io/phd