Deep learning approaches for adaptive audio processing and binary classification in digital health

Shuo Liu

A Dissertation
for the degree of
Doctor of Engineering (Dr.-Ing.)
in the
Faculty of Applied Computer Science
University of Augsburg

Augsburg, Germany

2022

Advisor
Prof. Dr.-Ing. habil. Björn Schuller

Second Examiner
Prof. Dr. Frank Kramer

Date of submission
December 2022

Date of Defence:
August 16th 2023

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

This thesis explores the use of deep learning technology in the fields of audio processing and digital health from the perspectives of problem modelling, data fitting, data augmentation, neural network design and model architecture selection, training objectives and optimisation strategies, etc. To this end, we based our research on a few representative tasks in these two fields, specifically audio enhancement and tasks associated with Coronavirus disease 2019 (COVID-19), such as the disease diagnosis. First, the two scientific topics are researched independently. The two scientific subjects are first individually explored. Then, with relation to employing such diagnostic models in actual noisy scenarios, we investigate the idea of merging the two themes, namely applying audio enhancement at the front-end to boost a speech-based COVID-19 diagnostic model's noise robustness.

In particular, for audio enhancement, we first concentrate on achieving a variety of front-end processing goals, such as noise reduction and source separation, through a unified deep learning architecture. The neural network solutions are implemented in our open-source tool of Neural Holistic Audio Enhancement System (N-HANS). Additionally, we offer a joint training method to alleviate the problem of mismatch when coupling the models for audio enhancement and the target application. By doing this, we can eliminate the obtrusive distractions that the enhancement process introduces which degrade the audio quality and show detrimental effect on the target application. The target applications used to assess the effectiveness of our proposed training method include Automatic Speech Recognition (ASR), Speech Commands Recognition (SCR), Speech Emotion Recognition (SER) and Acoustic Scene Classification (ASC). These representative applications span classification and regression problems, English and non-English languages, and speech and non-speech tasks.

In regards to the tasks for COVID-19, we first seek for the potential methods of diagnosing the illness using signals from smart or wearable devices. For this purpose, we extend the spectrum of signal types studied beyond speech, to incorporate, for example, respiratory sounds like coughing and breathing, as well as other bio-signals like heartbeats. Besides, we explore deep learning based solutions to mask-wearing detection from speech, or whether a speaker is wearing or not wearing a mask while speaking to create an automated, real-time, effective, affordable solution to assist in curbing the spread of the COVID-19 virus. Speech is an alternative useful source for COVID-19 detection, however, it is frequently affected by ambient noises like those found in everyday life. We can utilise our audio enhancement methods to improve the audio used as the COVID-19 detection model's input, facilitating its wider use in practical situations. This is accomplished by using vast amounts of data for the training of our audio enhancement model, which manages to outperform most prior methods while increasing the dependability and robustness of the back-end application.

There are certain connections, similarities and commonalities in the research methodology, specifically the use of deep learning techniques in both research topics presented in this thesis, despite the seeming lack of a direct connection between them. In other words, similar or identical deep learning algorithms are applied to these two research themes. One such example is the deep fusion method implemented in the N-HANS toolkit, which employs an auxiliary network to acquire contextual information, can also be exploited as a data fusion solution used in this work for merging the information from two audio types for COVID-19

diagnosis. Overall, the methodologies suggested in this thesis should not be restricted by the problems under consideration, but rather should be viewed more macroscopically as broad answers to a larger range of applications.

AKNOWLEDGEMENTS

First of all, I would like to sincerely express my most profound gratitude towards my doctoral advisor Prof. Dr. Björn Schuller, for providing me the opportunity to carry out the research in the field of deep learning, audio processing and digital health. He encouraged me to sharpen my ability of identifying scientific problems and seeking for proper solutions from a professional perspective. With his valuable advises and inspirational guidance, I was able to expand my professional horizons, conduct my research considering practical applications, make breakthroughs in difficult scientific and technical problems. Moreover, I would like to thank my other colleagues and collaborators who ever taught me, supported me and helped me with their kind suggestions for accomplishing this dissertation: Gil Keren, Emilia Parada-Cabaleiro, Jing Han, Adria Mallol-Ragolta, Andreas Triantafyllopoulos, Kun Qian, Maximilian Schmitt, Nicholas Cummins, and many others. I humbly extend my thanks to the chair members for holding the very warm and comfortable working and studying environment. I humbly extend my thanks to the chair members for holding the very warm and comfortable working and studying environment. I am most appreciative of my family's expectations of me. The life philosophy and experience of my parents and my sister are always supplying me the strength and courage I need to get through the challenging times. I am often impressed by my nephew's curiosity about many scientific subjects, which motivates me to advance in my study. Together, they have helped me to shape my past few years beyond just the pursuit of a Ph.D degree.

TABLE OF CONTENTS

CHAPTER 1

Introduction

The rapid development of Artificial Intelligence (AI) technology has led to viable and effective solutions to a broad range of scientific and industrial problems, including those in the fields of computer vision (CV) and audition (CA), natural language processing (NLP), robotics, recommendation systems, among others. The success of these solutions can be attributed to the exploitation of deep learning (DL) techniques which are adept at information extraction, encoding, conveyance, and inference on big data sets. wFor this purpose, data scientists and DL researchers focus primarily on 1) ensuring reliable data, including the quality of recorded data and the accuracy of labelling, and 2) exploring appropriate neural networks that can better extract information from data, encode it into neural representations with sufficient expressiveness, and make accurate inferences regarding specific tasks or data types.

The modelling approach to a deep learning problem, in particular the construction of the network architecture, should take into consideration the type and structure of the model's input data and the anticipated output in relation to different tasks. Specifically, the appropriate modules aimed at feature extraction and data modelling should be designed in line with the inherent characteristic of the data. For instance, Convolutional Neural Network (CNN) is proficient in aggregating spatial features from image data, while Recurrent Neural Network (RNN) excels at processing sequential input such as audio and texts. The majority of neural network designs, particularly those CNN extensions, have been effectively validated in the CV domain before being applied to other domains. Several deep learning frameworks are advocated to accommodate the irregular structure of sequential data, such as the variation in signal lengths. Several deep learning frameworks are advocated to accommodate the irregular structure of sequential data, such as the variation in signal lengths. For example, sequence-to-one architecture is applied to classification tasks like document classification and keyword spotting, while sequence-to-sequence architecture is a prototypical framework for neural machine translation, voice synthesis, etc. Rather than searching for suitable neural networks for a particular data structure, it is now common practise to convert the data to a specific format to meet the requirements of a model. In this regard, a number of general feature-extraction modules have been studied and used to transform the data. A number of other factors can have a significant impact on the performance deep learning models, including the training objectives, optimisation methods as well as the type of normalisation, model dimensionality, and the selection of hyper-parameters.

Although AI technology has made numerous advances in the fields of CV and NLP, and some of these deep learning methods have also been successfully adapted for audio applications, the uncertainties appear in speech and audio recordings under real-world conditions can hinder the practical applications of these techniques. Taking speech as an example, such uncertainties can stem from the within- and cross-speaker variations, differences in language, backgrounds, and recording equipment and settings, etc. One of the most prominent uncertainty is caused by innumerable environmental noise or interference sources in recording environments, which can be detrimental to the performance of the audio models trained using clean signal. Numerous potential sources of ambient noise or interference in recording conditions are a substantial cause of uncertainty, which can be detrimental to the performance of audio models trained with clean input. These unwanted and uncontrollable sound sources may be continuous or instantaneous, stationary or non-stationary, constant or variable in intensity, of varying duration and amplitude, and of the same or a different class than the target signal. In practice, the disturbance can be ambient noise such as traffic and industrial noise, the voice of interference speakers, refracted noises from obstructions, or echos, to mention a few examples. Moreover, these various kinds of noises or interferences frequently occur concurrently, and the desired audio is susceptible to being obscured by the overlapping surrounding noises, therefore further limiting the comprehension of the audio component of interest. The aggregation of these uncertainties make it more difficult to apply deep learning to speech and audio processing than to other modalities. For NLP tasks, AI models analyse texts composed of a fixed number of possible words and characters, whereas there exist infinite possibilities of audio indicating the same information. Thus, we anticipate AI models will be more tolerant of a multitude of variables in audio processing.

The first primary focus of this work is circling around the uncertainty that can arise in audio recordings, i. e., the extraction of the audio of interest from noisy surroundings using audio enhancement approaches. Conventionally, separating the signal of interest from interference signals of the same class is referred to as a source separation problem, while extracting it from sounds of different classes is known as a noise reduction or denoising problem. We first unify the two definitions based on their shared objective, which is to extract a target audio from its surrounding contexts, and then provide a solution for combining the two audio enhancement applications into a single framework. Moreover, the improvement of the audio quality through enhancement techniques should not only focus on the human hearing experience, but also on machine comprehension for the following audio applications. To accomplish this, we present a multi-task learning approach that optimises the modules for audio enhancement and its subsequent computer audition tasks concurrently, with the purpose of reaching the global optimal performance for both front-end and back-end models. Unlike previous research, we develop all of our AE models using large-scale data to assure their best possible generalisability and precision. Moreover, we extend the presented algorithms to address speech robustness for the task of acoustic scene classification (ASC), where the presence of human voice in acoustic scene recordings is considered as noise. The difficulty of extracting clean environmental sounds from audio recordings while simultaneously compressing the speech components has not received significant attention. We refer to the challenge of voice suppression as the inverse problem of speech enhancement, with the goal of preserving ambient sound while minimising the speech-related distractions. We categorise it as an audio enhancement task in which we investigate the gains in speech robustness to ASC classifiers.

Four representative audio applications, namely Automatic Speech Recognition (ASR), Speech Command Recognition (SCR), Speech Emotion Recognition (SER), as well as Acoustic Scene Classification (ASC), are cascaded to the audio enhancement systems we developed to assess the enhancement performance. However, the same methods, including the algorithms and neural network frameworks, can be broadly applied to more audio tasks, and one urgent task that could benefit from these methods is the detection of coronavirus disease 2019 (COVID-19) through speech, as these real-world audio recordings frequently contain life noises, such as television sounds and infant crying, etc.

Note that our exploration of the approaches for automated COVID-19 detection is not restricted to the use of speech signals, but also includes other human sounds such as coughing and breathing, as well as cardiac data from wearable devices. Since these kinds of data can be collected while the participants are engaged in their regular activities, it is possible for them to contain a variety of disturbances. In this study, we use speech as an example and examine our speech enhancement methods to see if the COVID-19 detection accuracy improves while using recordings of speech captured in the wild. In an effort to contribute further to the study of COVID-19, we expand our exploration into the use of deep learning methods for the automatic recognition of face masks from audio, with the hope that these approaches will assist in the containment of the virus' spread.

To summarise the above aspects, our study aims to resolve the following **research questions**:

- **Q1:** Can multiple audio enhancement problems be modelled inside a single framework using deep learning techniques?
- **Q2:** To what extent can audio enhancement systems improve the performance of audio applications that follow? How to optimise the audio enhancement models in order to maximise the benefits?
- **Q3:** Using deep learning, is it feasible to diagnose COVID-19 illness based on audio signals, such as speech, coughing, and breathing, as well as other biological data, such as heart rates? Can the use of speech enhancement improve COVID-19 detection from speech captured in real-world scenarios?
- **Q4:** How can deep learning methods be used to monitor the public's mask usage? How far away are practical applications of the mask detection solutions?

The **outline of the thesis** is structured as follows to answer these questions:

- **Chapter 2** provides a brief overview of deep learning methodologies, including an introduction to several fundamental neural network architectures and a discussion of some advanced deep learning techniques used throughout the research.
- **Chapter 3** proposes neural holistic audio enhancement solutions, with a particular emphasis on 1) the introduction of N-HANS, an open source toolkit that unifies multiple audio enhancement tasks, including two novel tasks: selective noise cancellation and voice suppression. 2) the optimisation of audio application performance by the deployment of audio enhancement.
- **Chapter 4** highlights our study on addressing COVID-19-related problems using deep learning techniques, including the automated illness diagnosis and mask detection tasks.
- **Chapter 5** presents a summary of the dissertation and proposes additional research questions for further study.

CHAPTER 2

Background Theory

2.1 Classic Deep Learning Models

Since the invention of the perceptron algorithm by Frank Rosenblatt in 1958, the machine learning technology has undergone numerous stages of evolution, from the extension of the approach to multilayer perceptrons (MLPs) that can approximate more complex non-linear functions to the construction of more versatile neural networks [1], such as CNN and RNN, specific to the structure of data used in the fields of CV, CA and NLP, etc. Moreover, a multitude of variants of these neural networks, such as Residual Network (ResNet) [2], a CNN with additional skip-connections, and Long Short-Term Memory model (LSTM) [3], a RNN employing memory mechanism, are found to be more successful in extracting information from data. Within these network structures, additional mechanisms, such as an attention module [4], may be included to discern the importance of different data portions. In particular, attention mechanism is victoriously utilised in Transformer model [4] and its several sophisticated variants, demonstrating superior performance in the most contemporary large-scale NLP tasks and audio applications. Vision Transformer (ViT) [5] has adopted the architecture of Transformer and attained the current state-of-the-art for CV problems. .

Despite the fact that some recent studies have prompted a trend of reconsidering the use of basic MLP models as their performance have been shown to be comparable to that of the neural networks tailored for specific applications, these elaborate neural networks have their own advantages such as computational complexity, memory consumption, and explainability, amongst others. Herein, we provide a summary of some widely-used deep learning models, which constitute the foundation for building the neural network solutions to our research problems.

Convolutional Neural Network (CNN)
CNN [1] is intended to extract spatial information from inputs with a grid-like structure, such as images. A CNN stacks multiple convolutional layers, allowing a hierarchical decomposition of its input, and hence a deeper CNN is able to learn more complex representations of the feature maps [6, 7, 8]. Subsequently, dense layers are typically used to map the learnt representations to predicted classes. Successful applications of such CNNs have surpassed classic signal processing solutions in many research and industrial domains, such as image and video recognition [9, 10], sequential data processing [11], and medical applications [12, 13, 14, 15] including those recent works for COVID-19 diagnosis [16, 17, 18].

A standard CNN model, such as LeNet-5 [19], AlexNet [20] and VGG [21], stacks several convolutional blocks in a sequence, with each block processing its input through one or more convolution, pooling, and activation operations. The convolution operation multiplies the input with kernel filters to aggregate the spatial information, and its output is constrained by an activation function like Sigmoid or *Rectified Linear Unit* (ReLU). Max-pooling is responsible for shrinking size of the resulting feature maps by only sampling the relatively more salient spatial activations. Typically, the final activations are flattened as a vector representation, and fully-connected layers can be used at the end of the CNN model to project the representation to predictions. To overcome the issue of *Internal Covariate Shift* (ICS), an effect caused by the slightly different distributions of different batches of training data [22], it is advisable to employ batch normalisation in the convolutional blocks.

CNN is also adopted to handle with other data types, such as audio signals in in a one-dimensional format. For this, the standard CNN should be reduced into a one-dimensional CNN. Alternately, the waveform may be converted into its time-frequency representation such that the 2D CNN can still be applied. Similarly, to analyse a text sentence using a 2D CNN, each word need to be turned into a vector representation using techniques like word-embedding.

Increasing the number of CNN layers can improve its capability to represent data, but may result in the well-known problem of gradient explosion or vanishing. To mitigate this problem and enable a substantially deeper architecture, ResNet [2] incorporates skip-connections, i. e., extra signal paths across convolutional layers. The smoothing effect of skip-connections on the loss landscape provides a more favourable environment for CNN model convergence [23], as shown by tasks in both the CV and CA domains [2, 24, 25]. The same concept is advanced further in DenseNet by applying skip-connections between all of the different convolutional layers that are conceivable. Other CNN designs, including Inception Net, SqueezeNet, EfficientNet, MobileNet, and Capsule network [26] were proposed to in relation to the needs of actual applications, such as memory space, inference efficiency, etc.

Recurrent Neural Network (RNN)

RNN [1] captures information from sequential data while accounting for underlying temporal dependencies. Each unit in a sequence is encoded by the RNN cell, and the information is carried forward when processing the subsequent unit. The conventional RNN cell, however, is susceptible to short-term memory. To facilitate the learning of lengthy sequences, two variants of RNN, namely LSTM and Gated Recurrent Unit (GRU) , integrate two respective memory mechanisms into the RNN cell to assist the model in choosing which information that learnt from earlier time-steps it should retain and forget for future processing.

The cell of an LSTM network consists of an input gate, a forget gate, and an output gate, with each gate calculating a threshold value depending on the input at time step t, $x_t \in \mathbb{R}^d$:

$$\begin{aligned}
f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f),\\
i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i),\\
o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o),
\end{aligned}$$

where $h_{t-1} \in \mathbb{R}^h$ represents the layer's hidden state at time $t-1$ or the initial hidden state at time 0. $W \in \mathbb{R}^{h \times d}$, $U \in \mathbb{R}^{h \times h}$, $b \in \mathbb{R}^h$ are the parameters, i. e., weights and biases, to be learnt. The symbol for Sigmoid function is denoted as σ. To aggregate the new input information, the input at the current time step and the hidden state at the previous time step are similarly processed, with the only difference being the activation function tanh:

$$g_t = \tanh(W_g x_t + U_g h_{t-1} + b_g).$$

The output of the LSTM layer relies on the input gate and forget gate, which controls the amount of information learnt from the current time step and prior states that needs to be stored in memory

$$\begin{aligned} c_t &= f_t \circ c_{t-1} + i_t \circ g_t, \\ h_t &= o_t \circ \tanh(c_t), \end{aligned}$$

where c_t is the cell state at time t, and $\circ$ denotes the Hadamard product. With the embedded memory mechanism, an LSTM is anticipated to be better capable of maintaining long-term dependencies in comparison to the conventional RNN.

The GRU model simplifies the cell unit to update and reset gates. The first gate defines the quantity of prior information that will be conveyed to the next hidden state, while the second gate specifies which information will be ignored. The gate thresholds are determined by

$$\begin{aligned} u_t &= \sigma(W_u x_t + U_u h_{t-1} + b_u), \\ r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r), \end{aligned}$$

thereafter applied successively to the current input and the last hidden state to produce the output of the cell at time step t:

$$\begin{aligned} \tilde{o} &= \tanh(W_o x_t + r_t * U_o h_{t-1} + b_u), \\ o_t &= u_t * h_{t-1} + (1 - u_t) * \tilde{o}. \end{aligned}$$

The standard RNN model and its advanced variants, such as LSTM and GRU, can process sequential input from both directions by utilising a forward RNN and a backward RNN to construct a bidirectional RNN, while maintaining information from both the past and the future time steps.

Convolutional Recurrent Neural Network (CRNN)

To extract spatial and sequential information simultaneously from certain data types, such as a video comprising a succession of images, it is feasible to model the data end-to-end by stacking CNN and RNN modules within a CRNN framework. In such a model, the CNN is responsible for learning a representation that conveys the spatial information of each frame, while the subsequent RNN aggregates the representations of the frame sequence and models the underlying temporal dynamics. In this instance, the RNN is tasked with summarising global sequence by accumulating each piece of local information acquired by the CNN. Alternately, the CNN may also be utilised in part to process global sequence. For example, the CNN can process the time-frequency representation of an audio signal and encode the global spatial information into embeddings as input of the RNN which further strengthen the temporal dependencies between frames. Due to the discrepancy of data dimensionality when connecting the CNN and RNN, additional effort is required at the interface between these two modules. Specifically, to create the RNN input, the CNN output is usually flattened along channel and feature directions in order to retain the information for each time step.

Attention Mechanisms

Attention mechanism has made great progress since its resounding success in sequence-to-sequence learning, such as for the problem of neural machine translation (NMT) [27, 28, 29]. Instead of assuming that all data input to a neural network is of equal relevance, attention mechanisms allow deep learning models to prioritise more informative data portions. Several types of attention mechanisms, including context attention, local attention, and component attention, can learn to assign attention weights to each time step in a sequence, with promising outcomes demonstrated for audio processing [30]. Higher attention values are placed on the more informative time steps.

Attention weights are computed, given an input sequence *value* (V), by the compatibility between its corresponding *key* (K) and a *query* (Q). The query can be generated based on the previous hidden state of the decoder. To aggregate the information of the whole sequence, the alignment scores can be multiplied with the input sequential value:

$$\mathbf{Attention}(Q, K, V) = \mathbf{softmax}(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}}V), \tag{1}$$

where d_k is the dimensionality of the input and $\sqrt{d_k}$ is a scaling factor that prevents the softmax function from being pushed into regions with extremely tiny gradients. Self attention aims to relate the different time steps of a sequence input, assuming that the input serves as the query, key and value.

Multi-head attention suggests to split the representation vector of one time step into shorter representation of equal length, which are then processed concurrently through the same attention mechanism in parallel. To this end, the query, key and value parameters are first divided into H-ways and each split individually passes through a separate attention head. Specifically, given the i-th attention head, a dense layer performs affine projection on Q_i, K_i, and V_i respectively, and the results flow through the attention:

$$head_i = \mathbf{Attention}(Q_i W_i^Q, K_i W_i^K, V_i W_i^V), \tag{2}$$

where W^Q, W^K and W^V are matrices to be learnt. The outputs from each head are then concatenated, and linearly transformed to the desired dimension through a dense layer:

$$\mathbf{MultiHead}(Q, K, V) = \mathbf{Concat}(head_1, head_2, ..., head_H)W^O, \tag{3}$$

where W^O is a matrix to be learnt.

Transformer

A Transformer [4] neural network is capable of performing sequence-to-sequence learning by adopting an encoder-decoder structure to attribute attention weights to the source sequence, signalling the importance of each time-step of the source sequence for each time-step of the target sequence. The model is adept at preserving long-term time-dependencies and allows parallel computing in training; hence, it is considered as a potent and effective alternative to RNN. Positional encoding annotates the relative position of input sequence and adds this information directly to the input, so that the subsequent processing in Transformer consistently carries the position information. Transformer-based algorithms have attained state-of-the-art performance for numerous sequential data processing applications in the NLP and CA fields.

The encoder and decoder of a Transformer are comprised of multiple layers. Each layer consists of a multi-head attention module, feed-forward layers and normalisation modules. Similar to ResNet, skip-connections are applied for more efficient model convergence. The multi-head attention in Transformer serves two primary purposes. The goal of the attention implemented inside within either the encoder or the decoder is to enrich the representation by learning the relative compatibility between the data of different time-steps; The attention bridging the encoder and the decoder assigns the attention weights indicating the importance of each frame of the source sequence to each frame of the target sequence.

A basic Transformer encoder can be used independently for sequence-to-one learning, essentially reducing the model to multi-head attention and feed-forward modules, two processing steps that are performed repeatedly. Surprisingly, this straightforward application succeeds in achieving highly promising outcomes for a variety of NLP and CA tasks. Another example is Visual Transformer (ViT), which has broadened the use of Transformer to the CV domain and inspired a number of works that have led to the current state-of-the-art.

Auto-encoder

An auto-encoder [31] is comprised of an encoder that encodes input data into latent attributes, and a decoder that reconstructs the original input from the learnt latent attributes. The model is optimised to minimise the reconstruction error in an unsupervised way, and the size of latent attributes must be selected with care to ensure the representation capability. MLPs, CNNs, RNNs, and even Transformers, etc., can be utilised as the backbone model for constructing an auto-encoder encoder's and decoder. To obtain more representative attributes and prevent the auto-encoder from falling into identity function, several techniques for enhancing the effectiveness of the model have been presented. A popular method is to introduce particular types of noise to the encoder's input, and the decoding objective is to recover the original clean input from the learnt latent attributes from the noisy input, so constructing a denoising auto-encoder [32]. An alternate solution is to use sparse auto-encoder [32], which encourages sparsity in learning latent attributes, and so enables the length of latent attributes to be increased. Variational Auto-Encoder (VAE) [33] can enforce an expected distribution in the latent space, and hence such model provides generative solutions to statistical inference problem. All of these auto-encoder types are usually trained in an unsupervised learning manner, but their effectiveness has also been shown for semi-supervised and supervised learning, when incorporating some or all of the data labels in the model optimisation.

2.2 Advanced Deep Learning Approaches

This section describes some sophisticated neural frameworks and learning systems used in this thesis. These generalised frameworks can be constructed utilising, but are not limited to, the typical neural networks outlined in Section 2.1. We present a neural network model with auxiliary networks aimed to learn additional reference information from extrinsic signal, as it is used in N-HANS [34]. The paradigm can also be used to fuse data of different modalities or types. Another model combines an encoder and a decoder in symmetry, with additional short-connections between the encoder and decoder layers, resulting in a U-shaped network [35]. Its success in the precise segmentation of high-resolution medical images has been applied to the separation of audio sources. We apply the same method to a similar problem, i. e., audio denoising, in an effort to extract the audio of interest from recordings made in the wild. Besides, the performance of a neural network architecture may benefit from an appropriate training approach, which should be applied in accordance with the formulation of loss functions derived from particular learning objectives. These learning objectives should thus reflect the intended direction for model optimisation. Herein, we additionally describe the learning principles with a focus on the model architectures and training loss formulations used to optimise these models.

Auxiliary Network

The auxiliary network [36] enables a model to extract additional information from supplemental input. Given supplementary data, the need for human-annotated labels is reduced when optimising the model. For this, the information gleaned from the auxiliary network should be incorporated into the model using fusion methods, such as early fusion, late fusion, or the deep fusion methods proposed in N-HANS [34]. The auxiliary network in N-HANS is used to learn context and speaker information, allowing the model to integrate different audio enhancement tasks.

The approach of deploying an auxiliary network is versatile and may be used for diverse purposes, such as a method of feature fusion that combines information from data of different types or modalities. Unlike previous fusion methods, the method using auxiliary network treats one data type as primary and the others as supplementary. The performance of the model may vary depending on the selection of the primary data, as seen in Section 4.3 , which compares two distinct audio types, breathing and cough sounds, as the primary data for diagnosing COVID-19.

Furthermore, the deployment of an auxiliary network can provide a scaling effect for personalised models. An example can be seen in the work of personalised SER [37], which seeks to train a model that can identify a speaker's emotional state given a neural speech of the same person as a reference sample. It is worth to highlight that an auxiliary network may be trained independently or in conjunction with the primary network during model optimisation.

U-Net

An U-shaped neural network [35] introduces skip-connections, inspired by ResNet, to an auto-encoder architecture, to feed the information learnt from each encoder layer to their corresponding decoder layer. The architecture was initially designed to conduct rapid and precise image segmentation. To accomplish this, two symmetric CNNs are used, one as the encoder, which builds a contracting path to encode its input into context information; and the other as the decoder, which constructs an expansive path to propagate the context information to reconstruct the high-resolution output at the decoder side through up-sampling operators.

Similar network designs have been investigated for audio processing applications [38], in which one-dimensional CNN and complex-valued CNN, in addition to traditional real-valued CNN, are considered as the encoder and the decoder. Similar to its efficacy in image segmentation, the numerous feature channels of U-Net are conductive to audio enhancement, enabling the encoder to more completely decompose the input noisy audio and the decoder to produce high-resolution output which is required for outstanding output speech quality.

Contrastive Learning

Constrastive learning [39] trains a model to encode similar data samples into closer embeddings in the representation space, while pushing the dissimilar data samples far apart. To satisfy this training objective, several forms of contrastive losses, such as triplet loss [40], margin loss [41], multi-class N-pair loss [42], noise contrastive estimation (NCE) [43] and its extended InfoNCE [44], soft-nearest neighbors loss [45], and others, have been presented.

The conventional max-margin contrastive loss formulation pairs an anchor data x in a batch $\mathscr{X}$ with a sample from the same class and a sample from another class, denoted as the positive sample x^+ and the negative sample x^-, respectively. A neural network is optimised to minimise the embedding distance between the input pairs from the same class and maximise the distance otherwise according to

$$\begin{aligned} L(x, x^+) &= ||f(x) - f(x^+)||_2^2 \\ L(x, x^-) &= \max(0, \epsilon - ||f(x) - f(x^-)||_2)^2, \end{aligned} \tag{4}$$

where ϵ specifies the minimum margin expected between the embeddings of the samples from two distinct classes. Triplet loss combines the individual optimisations for positive and negative samples:

$$L(x, x^+, x^-) = \sum_{x \in \mathscr{X}} \max(0, ||f(x) - f(x^+)||_2^2 - ||f(x) - f(x^-)||_2^2 + \epsilon). \tag{5}$$

Furthermore, including more positive and negative samples in a training batch can be useful to contrastive learning. This can be accomplished, for example, by using multi-class N-pair loss [42]

$$\begin{aligned} L(x, x^+, x^-_{n \in [1, 2N-1]}) &= log(1 + \sum_{n=1}^{2N-1} e^{f(x)^T f(x_n^-) - f(x)^T f(x^+)}) \\ &= -log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{n=1}^{2N-1} e^{f(x)^T f(x_n^-)}}, \end{aligned} \tag{6}$$

which generalises triplet loss by enabling joint comparisons between the anchor and its multiple negative samples. Normalised temperature-scaled cross-entropy loss (NT-Xent) or InfoNCE [44], inspired by Noise Contrastive Estimation (NCE) [43], incorporates a temperature parameter to punish the impact of negative samples, analogous to the effect of ϵ in Equation (4) and Equation (5):

$$L(x, x^+, x^-_{n \in [1, N-1]}) = -log \frac{e^{f(x)^T f(x^+)/\tau}}{\sum_{n=1}^{N-1} e^{f(x)^T f(x_n^-)/\tau}}. \tag{7}$$

The task of optimising a model using InfoNCE loss function is often turned into the minimisation of a cross-entropy loss of N classes, with the goal of identifying a positive sample from the remaining $N - 1$ negative samples.

Self-supervised Learning (SSL)

Self-supervised learning [46] is a method of representation learning that requires no human annotations on the training data. To do this, a model is trained to solve a designed pretext problem, allowing it to encode the data directly into its general representations. Using simple extra modules, such as MLPs, the model can be fine-tuned to accommodate multiple downstream tasks.

SSL frameworks are divided into several categories based to the method used to generate pseudo-labels that are associated with the training objectives. A typical such framework employs the means of auto-encoding, which embeds the information of a signal's distorted version within latent attributes and compel the decoder to reconstruct the signal into its original form. The input deformation is advantageous to the generalisation of the learnt latent attributes. A SSL model can be trained in the framework of Siamese or triplet networks. The former optimises the model by minimising the distance between the two representations of the same data without incorporating any negative samples. To do this, two sub-networks should be placed in parallel, and when they are identical, asynchronous training must be considered to prevent mode collapse. The latter makes use of a single network and contrasts the data representation to that of negative samples. Clustering techniques are also used to create pseudo-labels for training a SSL model. The learning of data representation aims to approach the cluster centroids; nevertheless, the region surrounding each cluster centroid is left to ensure the representation diversity.

SSL has been explored for the CV [47], CA [48] and NLP [49] domains, as well as cross-modality tasks.

CHAPTER 3

Neural Holistic Audio Enhancement

3.1 Introduction

High audio quality, including attributes such as clarity, fidelity, and intelligibility, is crucial not only for auditory perception, but also for applications in the field of computer audition. Nonetheless, while audio recording in the real world, the signal may be deteriorated by a variety of noise sources, such as ambient noise, the voices of unwanted speakers and reflections from surrounding obstacles, etc. These disruptions may severely impair the performance of audio models that are trained on clean data, especially those that process speech data [50, 51, 52].

To eliminate this detrimental effect and ensure the signal quality, audio enhancement methods aimed at extracting the audio of interest from a noisy recording are usually exploited as a front-end module of the following audio application. In the context of hearing aids, for instance, a noise reduction module implemented in advance of other functional modules, such as that for volume amplification, can assist the hearing impaired in lessening the difficulty of auditory understanding. Similarly, applying audio enhancement to other applications such as Automatic Speech Recognition (ASR) [52] and Speech Emotion Recognition (SER) [50], can reduce the uncertainty in audio caused by, for example, the difference in recording conditions and devices, thereby enabling the robust running of these models in practise.

In the era of deep learning, neural network-based solutions for the two most prominent audio enhancement problems, i. e., speech denoising [53, 54, 55, 56, 57, 58, 59, 60, 61, 62] and source separation [63, 64, 65, 66, 67], have shown more effective than traditional signal processing algorithms [68]. The former task aims to suppress the background noise in a speech recording, while the latter separates a mixture of sounds into their respective origins. However, for processing audio recorded in the wild, a number of challenges remain:

- First of all, the generalisability of an audio enhancement model is constrained by the quantity and diversity of training data, which cannot exhaustively cover all kinds of real-world audio environments.

- An audio recording can contain numerous types of noise simultaneously, including non-stationary noise [69], which poses a difficulty for audio enhancement techniques that process only a single type of noise [70, 56, 58].

- The uncertainty in audio recording can substantially increase when the audio has varying Signal-to-Noise Ratios (SNRs) over time due to near-field and far-field effects, or differing room impulse responses. Without awareness of the surrounding environment, an audio enhancement system cannot cope with circumstances of such complexity.

- The protection of the target audio signal is also a factor in determining the effectiveness of an audio enhancement model. This is associated with the accurate estimation of the noise and interference components of the noisy audio. If the interfering components and the target signal share the similar acoustic properties, noise estimation becomes exceedingly challenging and aggressive estimating can compromise the audio naturalness.

- The solutions developed in accordance with the conventional formulation of audio enhancement fall short of the freedom to select the audio content to be removed and preserved. This limitation can restrict their application scenarios, and an additional adaptation might be required for each circumstance . For instance, a speech enhancement model meant to eliminate all background sounds can result in perilous conditions if the background contains vital signals such as an aerial defence alarm.

- For the purpose of applying audio enhancement at the front-end of a target audio application, the enhancement model should be optimised towards the target model. In contrast, an independently trained audio enhancement model that disregards its subsequent processing may yield sub-optimal results. In fact, the distortions and artefacts unintentionally introduced by the enhancement model may contaminate the audio of interest and hinder the system's overall performance.

To address these issues, we first concentrate on deep learning solutions for the unification of diverse audio enhancement capabilities, with three objectives: 1) relaxing the requirements and assumptions on training data, 2) reducing the confusion between audio of interest and interference, and 3) permitting the preservation and removal of a selection of audio components. Then, we investigate the training paradigm that optimises the audio enhancement model towards its following computer audition tasks. Based on these factors, the training methods given in this chapter for the audio enhancement systems offer the following traits and merits :

- In contrast to prior efforts on audio enhancement, which treat audio denoising and source separation as two distinct tasks, we realise both functionalities in a single model architecture and integrate the neural network into *N-HANS*, an open-source toolkit. The model employs auxiliary networks to learn reference information from extra audio examples, allowing it to instantly adapt to any contextual background or speaker.

- In case when a particular noise needs to be suppressed from a noisy audio, the remaining background noise can be maintained to pertain a natural audio surrounding. To this end, we add a third functionality called selective noise suppression (SNS) in N-HANS, which allows the selection of desired noise, referred to as "positive noise", and the suppression of unwanted noise, referred to as "negative noise".

- The presence of speech in surrounding environments is considered as interference or noise for Acoustic Scene Classification (ASC), a task that classifies an audio sample to the type of environment in which it was recorded. [71, 72, 73, 74]. To overcome this issue, we design a task, *voice suppression*, which inverts the roles of speech and environmental background. A voice suppression model should eliminate human speech from audio recordings while maintaining only clean environmental sounds of adequate audio quality for ASC models. Voice suppression is seen as a significant byproduct of N-HANS.

- To advance the effect of audio enhancement for audio models operating in noisy situations, we explore several joint optimisation techniques. Based on the experimental results from four representative audio applications, i. e., Automatic Speech Recognition (ASR), Speech Emotion Recognition(SER), Speech Command Recognition (SCR), and Acoustic Scene Classification (ASC), our two presented solutions, i. e., multi-task learning and iterative training, manage to outperform the methods that do not account for joint optimisation.

The remainder of this chapter introduces N-HANS along with its key features and the realisation approaches for its four primary functionalities, namely speech denoising, source separation, selective noise cancellation, and voice suppression. The ensuing section discusses the joint optimisation of an audio enhancement system and its subsequent audio model, with an emphasis on our iterative training approach and multi-task learning framework. In addition to describing model architectures and evaluating their performance, we illustrate the effect of the audio enhancement methods in each section.

3.2 Related Work

Speech Enhancement

The task of speech enhancement is typically formulated as a supervised learning problem, and its solutions can be broadly categorised as frequency- and time-domain techniques [63, 58, 75]. The frequency-domain solutions either learn a spectral mapping from the time-frequency (TF) representation of the noisy audio to that of the clean audio, or they estimate a mask that approximates the proposition of the clean component on each TF-bin of the noisy spectrogram. In these methods, the phase information of the noisy spectrogram can be used to reconstruct the enhanced audio without alteration. However, recent research has highlighted the significance of phase information to the audio quality of reconstructed speech. Two effective methods for enhancing phase information in this process are complex-valued neural networks [38] and iterative phase estimation [76, 77, 78]. The time-domain SE models, including Fully Convolution Network (FCN) [79], waveNet [80], and Wave-U-Net [81, 82], operate directly on the raw audio waveform while naturally preserving the phase information in the signal during processing. Both the time-domain and frequency-domain SE models may benefit from a global view of the audio input. For this purpose, an additional discriminator network is used to evaluate the quality of enhanced audio, resulting in a Generative Adversarial Network (GAN) [83] for SE, named as SEGAN, that compromises the denoising fineness and global performance. Within the SEGAN framework [56], a generator network and a discriminator network are optimised iteratively. The generator is trained to recover clean speech from a given noisy audio, and its output attempts to deceive the discriminator network that is used to assess the recovered speech in comparison to the ground-truth clean signal. The discriminator network is proficient at acquiring global audio information when it is trained to be capable of differentiate between the recovered speech and its associated ground-truth. Numerous strategies have been presented to improve the GAN-based SE models in order to produce enhanced audio of higher quality [84, 85, 86, 87, 88, 89]

Source Separation

Constructing an automatic source separation system capable of extracting a target speech signal from two overlapping speech sources remains challenging. In case that the speakers share similar acoustic features, the process becomes substantially more difficult. Traditional signal processing algorithms, including Principle Component Analysis (PCA) [90], Independent Component Analysis (ICA) [91] and Non-negative Matrix Factorisation (NMF) [92], have been found to be effective for multi-channel source separation. However, these methods necessitate additional assumptions such as source independence, space sparsity and non-negative constraints. Statistical techniques for single-channel source separation, such as Bayesian models [93], sparse non-negative matrix factorisation [94, 95] and Empirical Mode Decomposition (EMD) [96], utilise the underlying statistics of speech signal and transform the input data to conform to the model assumptions. However, these assumptions restrict their applicability to real-world data [97].

Neural network based approaches for source separation [98, 99, 100] reduce the requirements on input data, profit greatly from data diversity and outperform the conventional signal processing algorithms [101]. A popular source separation method, inspired by image-to-image segmentation [102], employs a U-Net to decompose a music spectrogram into vocal and instrumental components [103]. In another work for the same task [104], multi-band DenseNets were utilised to retrieve lengthy contextual information in order to improve the separation performance. However, when applying these separation approaches to speech signals, the well-known permutation problem can arise [105]. Recent research that circumvent the permutation issue present the deep clustering approach [65] and the deep attractor network (DaNet) [106]. In these two methods, a deep recurrent neural network is trained to generate similar embeddings for time-frequency bins (TF-bins) from the same speaker. Using a clustering approach, the TF-bins are then clustered into distinct speakers based on the learnt embeddings. Tasnet [107] and Conv-Tasnet [67] analyse and reconstruct audio in time-domain using an encoder-decoder framework, hence avoiding the problem that occurs during time-frequency decomposition.

Target Speaker Extraction

As a specialised form of audio source separation, the speech of a target speaker can be extracted from overlapping speech by conditioning a source separation model on the enrolment sample of the speaker [108, 109, 110, 111]. Typically, an additional neural network is used to encode a given audio signalling the target speaker into an embedding representing the acoustic attributes. The source separation model is trained to extract speech components matching the acoustic attributes from the mixture audio . This generalises the model so that it can adapt to target speakers it has never seen during training. The auxiliary network and source separation model can be independently trained [109] or jointly trained [110, 111]. X-Tasnet [112] and SpEx [113] expand the architecture of two successful source separation models, Tasnet and Conv-Tasnet, for speaker extraction, and are capable of working on the time-domain audio waveform. L-SpEx [114] reports improved performance of target speaker extraction using spatial cues provided by their own proposed speaker localiser.

Voice Suppression

Voice suppression is defined as the process of removing all speech from an audio recording while retaining ambient sounds. A similar definition for the separation of musical sources has been established in [115]. The method, however, seeks to disassemble the singing voice and its accompaniments in a musical segment while allowing certain vocal components remaining in the estimated accompaniments. In contrast, we explore a denoising-style model that uses a spectral-mapping approach to estimate the surrounding environments in an audio while discarding the human voice to the greatest possible extent.

Unlike the processing of relatively stable and slowly time-varying environmental sounds, learning the representation of non-stationary and fast time-varying speech signals demands a higher temporal resolution. In addition to RNN, various specialised network modules, particularly those based on predictive coding techniques, such as aggressive predictive coding (APC) [116], masked predictive coding (MPC), and contrastive predictive coding (CPC) [44] have been proposed for capturing transient information in speech signals more effectively.

Multi-task Learning with Audio Enhancement
Using independently designed audio enhancement modules can increase the input quality for subsequent audio models, such as those for ASR [117, 118, 119, 120]. These audio models are sometimes further empowered with data augmentation techniques, such as SpecAugment [121] or additive noise [50], to boost their robustness against disturbances. In practise, the enhancement effect on the cascaded ASR models might degrade due to the unwanted distortions and artefacts introduced in the enhanced audio [122]. To increase the tolerance for these distortions, we can fine-tune the ASR model based on the output audio of the enhancement module [123, 124]. Furthermore, the frontend SE model's parameters can be frozen or trained during the optimisation of the ASR model. In the latter case, the ASR loss is responsible for updating the parameters of the entire combined model, including those of the SE model [123]. However, since the constraint on training the SE model is relaxed, the SE effect may be diminished.

Recent research [125, 126] incorporates the training objectives of SE and ASR into a multi-task learning problem, where the losses of these two modules are compounded for joint optimisation. A dynamic factor is applied to the loss combination to regulate the training focus between the AE and ASR models [127]. This method can be improved by using more advanced deep learning techniques, such as generative adversarial networks (GANs) for SE [128, 129], and self-supervised learning (SSL) for ASR [130]. Also, Joint training has been implemented for speech command recognition or keyword spotting [131, 132]. However, for other audio applications such as SER [50, 133] and ASC [134], an audio enhancement module is often trained separately and then cascaded to the target audio models for noise reduction. Joint training for these audio tasks merits further research.

3.3 N-HANS: Neural Holistic Audio Enhancement System

Using a single neural network architecture, N-HANS integrates audio denoising and source separation functionalities into a publicly available tool. This tool also introduces a novel application, selective noise suppression, which aims to suppress only unwanted sounds while retaining others in order to maintain a natural audio environment and protect particularly vital signals, such as alarms and other auditory warnings. Table 3.3.1 provides an overview of the open-source audio improvement toolkits that are currently available to the general public. The vast majority of these works concentrate solely on a single job, either speech denoising or source separation. Moreover, some of them are developed conditioned on certain acoustic assumptions, resulting in limitations such as processing only stationary noises.

VoiceBox[1] and CtuCopy[2] are two speech processing toolkits that employ traditional signal processing algorithms. The first tool assembles algorithms for a broad variety of audio tasks including denoising, while the second was established for audio feature extraction and

Table 3.3.1: List of the most popular open-source toolkits for audio enhancement. Processing methods: classic signal processing (SP), machine learning (ML) except deep learning, deep learning (DL); Functionalities: denoising (DE), speech separation (SS), selective noise suppression (SNS); and Adaptation ability to speaker (Spk) and speech surrounding environments (Env), are indicated. (*Source*: [34])

	Methods			**Functionalities**			**Adaptation**	
Toolkit	**SP**	**ML**	**DL**	**DE**	**SS**	**SNS**	**Spk**	**Env**
VoiceBox[1]	✓			✓				
CtuCopy[2]	✓			✓				
SETK[3]	✓		✓	✓	✓			
SE Toolkit [135]			✓	✓				
SEDNN [136]			✓	✓				
SEGAN [56]			✓	✓				
openBlissart [140]		✓			✓			
FASST [141]		✓			✓			
GCC-NMF[4]		✓		✓	✓			
Asteroid [138]			✓		✓			
UNTWIST [137]		✓	✓		✓			
N-HANS[1]			✓	✓	✓	✓	✓	✓

speech denoising by combining Wiener filtering theory with spectrum subtraction methods. Both toolkits are dependent on the accurate estimation of noise power, which cannot be assured under non-stationary noise conditions. With the rapid development and increasing use of deep learning technology, neural networks-based denoising toolkits, such as SETK[3], SE Toolkit [135], SEDNN [136], SEGAN [56], and U-Net[58, 57], have been introduced and shown superior effectiveness. However, these methods were primarily intended for speech denoising and must be adapted for source separation. Untwist [137] and Asteroid [138] are two prominent source separation toolkits. Asteroid incorporates numerous neural networks, such as ConvTasnet [67], Deep clustering [65], and Chimera++ [139], with improved source separation capabilities. Speech denoising and source separation methods based on Non-negative Matrix Factorisation (NMF) have also been presented in OpenBlissart [140] and Flexible Audio Source Separation Toolbox (FASST) [141]. FASST considers Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM) for NMF training. GCC-NMF [4] applies the Generalised Cross Correlation (GCC) spatial localisation method to a denoising problem.

Due to the reliance on the diversity of speakers and noise types in the training data, these audio enhancement tools have limited applicability to unseen speakers and environments in real-world circumstances. To overcome this issue, N-HANS exploits auxiliary networks to adapt the audio processing to unseen speakers and speech surroundings.

[1] http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html

[2] http://noel.feld.cvut.cz/speechlab/share/download/ctucopy/ctucopy3.html

[3] https://github.com/funcwj/setk

[4] https://github.com/seanwood/gcc-nmf

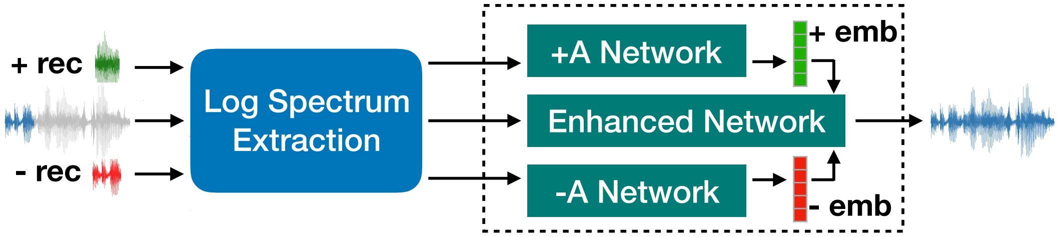


Fig. 3.3.1: System framework of N-HANS [34]. The system process three inputs: a noisy audio signal and two additional recordings: positive (+rec), negative (−rec); The +A Network processes the +rec to produce a positive embedding vector (+emb) that specifies the audio components to be preserved. The −A Network analyses the −rec to acquire the negative embedding vector (−emb) which indicates the components to be suppressed. The enhanced network processes the noisy audio as well as the positive and negative embeddings in order to generate the desired output.

Table 3.3.2: N-HANS overview of inputs and outputs. The input, i. e., the raw input and the positive (+) and negative (−) recordings; as well as the output, are indicated for the three involved tasks: denoising (DE), speech separation (SS), selective noise suppression (SNS). (*Source*: [34]).

Task	DE	SS	SNS
raw input	noisy audio	overlapping sources	noisy audio
+rec	-	target source	noise to preserve
−rec	noise to suppress	interference source	noise to suppress
output	denoised audio	separated source	denoised audio

3.3.1 System Overview & Functionalities

N-HANS, which is integrated with two trained models sharing the same architecture, can handle unseen speakers and noises by supplying its auxiliary sub-networks with extra audio examples that specify the audio components to be preserved and removed. Within these systems, deep fusion mechanism was introduced to inject the context information into the conditional residual network, therefore endowing the system with the capability for speaker- and environment-adaptation (for further information, see Section 3.3.2). This allows it to recover an audio of interest while reducing interference sources with two systems, audio source separation and selective noise suppression, both of which are built on an ±Auxiliary (A) Network (cf. Figure 3.3.1).

For different N-HANS tasks, an overview of input and output information is given in Table 3.3.2. As the inputs, the log magnitude spectra are extracted from the contaminated audio as well as the positive and negative recordings by taking thee logarithmic absolute values of the Short-Time Fourier Transformation (STFT). The +A Network processes the spectrum of the positive recording to generate a positive embedding vector, while the −A Network analyses the spectrum of the negative recording to produce a negative embedding vector. The Enhanced Network emits the denoised or separated audio by processing the

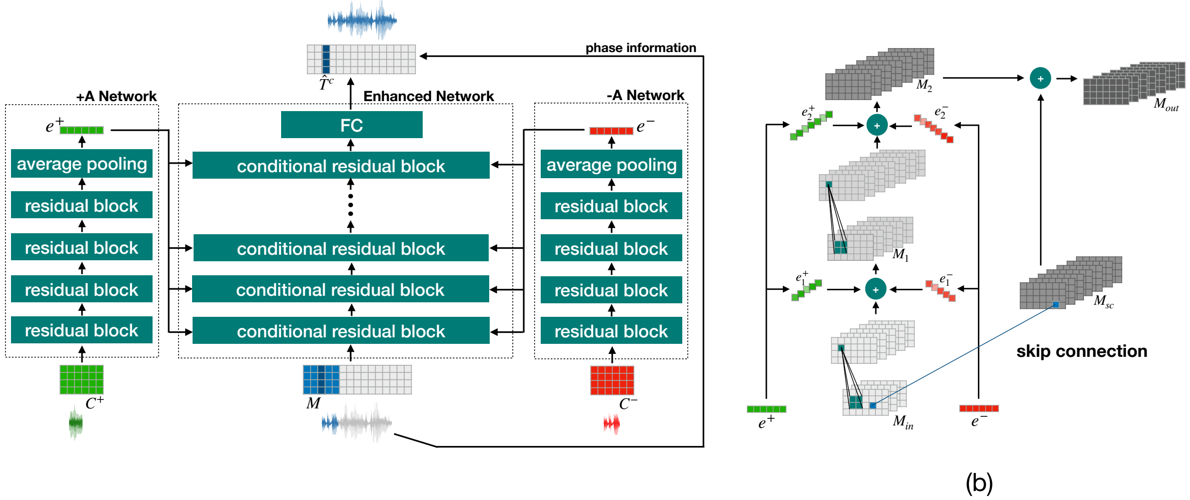


Fig. 3.3.2: (a) Architecture of the ±Auxiliary (A) Networks. The +A and −A Networks process the positive and negative contexts (C$^+$ and C$^-$) via a sequence of 4 residual blocks to produce positive and negative embeddings (e$^+$ and e$^-$). To estimate the *contamination frame* (CF), the Enhanced Network processes the contaminated segment M (noisy or overlapping segment) through a sequence of 8 residual blocks, each additionally conditioned by the e$^+$ and e$^-$. (b) Conditional residual block: the learnt positive and negative embeddings (e^+ and e^-) are injected in the two convolutional layers of the enhanced network. Block's input (M_{in}), output of skip connection path (M_{sc}), first convolutional layer output (M_1), second convolutional layer output (M_2), and block's output (M_{out}) are also indicated. (*Source*: [34])

noisy audio conditioned on the two embedding vectors carrying, respectively, the features of the target and interference audio components. By identifying the positive and negative context, N-HANS makes the system adaptable to unseen audio sources, speakers, and speech backgrounds.

3.3.2 Network Architectures & Training Objectives

The architecture of ±Auxiliary (A) Networks comprises a succession of residual blocks (cf. Figure 3.3.2). Residual network (ResNet) augments CNN with skip-connections across convolutional layers, producing a smoother loss landscape during training, and enabling a substantially deeper architecture [23]. Its effectiveness has been observed in computer vision and audio processing studies [2, 24, 25]. A residual block possesses two signal flow routes, i. e., a primary path and a skip-connection path. The primary path processes the input of the block with two successive convolutional layers, whereas the short-connection path converts the channels of the input using a 1×1 convolution. The block output incorporates the outcomes of both paths. Batch normalisation [142, 143] is applied to each convolutional layer to alleviate the popular ICS problem. In addition, a Rectified Linear Unit (ReLU) [144] is typically used as activation function to limit the output's scale. Both modules are shown to contribute to the convergence of CNN models.

The N-HANS architecture makes use of three subnetworks, each of which is comprised of multiple residual blocks with the specifications given in Table 3.3.3. The +A embedding network learns a *positive embedding* from the positive context. Likewise, a second embed-

Table 3.3.3: N-HANS Model Specifications. Each residual block has its own kernel size, stride, and number (#) of channels. (*Source*: [34])

Auxiliary Networks

Block	Kernel	Stride	#Channels
1	$(8,4)$	$(3,2)$	64
2	$(8,4)$	$(3,2)$	128
3	$(4,4)$	$(1,1)$	256
4	$(4,4)$	$(1,2)$	512

Enhanced Network

Block	Kernel	Stride	#Channels
1	$(4,4)$	$(1,1)$	64
2	$(4,4)$	$(1,1)$	64
3	$(4,4)$	$(2,2)$	128
4	$(4,4)$	$(1,1)$	128
5	$(3,3)$	$(2,2)$	256
6	$(3,3)$	$(1,1)$	256
7	$(3,3)$	$(2,2)$	512
8	$(3,3)$	$(1,1)$	512

ding network with the same architecture (−A Network) generates *negative embedding* from the negative context. Conditioned on the positive and negative embeddings, the enhanced network analyses a contaminated audio segment to estimate the *contamination frame* (CF) approximating the audio components that need to be eliminated in the centre frame of the contaminated segment. To determine the *enhanced frame* (EF)[5], the contamination frame is subtracted from the centre frame of the contaminated segment. The enhanced frame is trained to converge to the target clean frame, i. e., the centre frame of the target segment.

3.3.2.1 Generation of Positive and Negative Embeddings

Due to their distinct functions, the positive and negative embedding networks share an identical structure but have different training parameters. The embedding networks create the embedding vectors representing the positive and negative contexts (C^+ and C^- in Figure 3.3.2(a)) through a sequence of four residual blocks. Inside this processing, the convolution output, i.e., feature maps, are averaged over all locations (time steps and frequency bins) to form a positive and negative embedding vector with a fixed length of 512:

$$e^+ = \mathbf{avg}(f^{+A}(C^+)), \tag{1}$$

$$e^- = \mathbf{avg}(f^{-A}(C^-)), \tag{2}$$

where f^{+A} and f^{-A} denote the sequential processing by the residual blocks in +A and −A Networks. The two embeddings are subsequently injected into the enhanced network to assist with audio denoising, source separation, and selective noise suppression tasks.

[5] EF refers to the *estimated denoised frame* for the tasks of audio denoising and selective noise suppression; For source separation, it refers to the *estimated separated frame*.

3.3.2.2 Enhanced Network with Deep Fusion Mechanism

The enhanced network sequences eight conditional residual blocks, each of which is conditioned on the learnt positive and negative embeddings, to enhance the contaminated segment. As illustrated in Figure 3.3.2(b), in addition to the standard processing of residual block, the learnt positive and negative embeddings are projected, for each convolutional layer, to the length equal to the channel numbers using a trainable dense layer, and subsequently added to every location on the feature maps. Specifically, given the input $M_{in} \in \mathbb{R}^{T \times F \times C_{in}}$, the first convolutional layer outputs

$$M_1 = \mathbf{conv}(M_{in}) + e_1^+ + e_1^-, \tag{3}$$

which has the dimension of $T \times F \times C_1$, and

$$e_1^+ = e^+ W_1^+ + b_1^+, \tag{4}$$

$$e_1^- = e^- W_1^- + b_1^-, \tag{5}$$

denote the projected embedding vectors with the length of C_1. W_1^+, b_1^+, and W_1^-, b_1^- are trainable parameters. Note that the projected embedding vectors are extended to the size of the convolution output by using array broadcasting.

Subsequently, the second convolutional layer further processes M_1, resulting in

$$M_2 = \mathbf{conv}(M_1) + e_2^+ + e_2^-, \tag{6}$$

with the dimension of $T \times F \times C_2$, where

$$e_2^+ = e^+ W_2^+ + b_2^+, \tag{7}$$

$$e_2^- = e^- W_2^- + b_2^- \tag{8}$$

are the projected embedding vectors of the length of C_2. By feeding the positive and negative context information to all the convolutional layers of the enhanced network, the model is able to identify the audio components that are anticipated to be preserved and suppressed from the contaminated segment.

On the skip-connection path, the channels of block input M_{in} are adjusted by means of 1×1 convolution, resulting in

$$M_{sc} = \mathbf{conv}_{1\times 1}(M_{in}), \tag{9}$$

which has the shape of $T \times F \times C_2$. Finally, the block output is computed by adding the outcomes of the primary path and the skip-connection path.

$$M_{out} = M_2 + M_{sc}. \tag{10}$$

The output of the last layer is additionally convolved along the time axis to aggregate the temporal information, flattened to a vector, and then projected to the length of F ($F = 201$ in experiments) through a fully-connected layer, representing the enhanced frame:

$$\hat{S}^c = \mathbf{conv_T}(f^{enh}(M, e^+, e^-))W_o + b_o, \tag{11}$$

where f^{enh} denotes the processing of the enhanced network, and $\mathbf{conv_T}$ stands for the convolution along time direction. W_o, b_o are the learnable parameters of a fully connected layer. Considering the practical application, the learning of unseen speakers may have a negligible effect on the task of voice suppression, we did not consider the use of auxiliary networks in its model.

Training Loss & Optimisation Strategy

For our four audio enhancement tasks, i. e., speech denoising, source separation, selective noise suppression, and voice suppression, the enhanced frame is computed by subtracting the contamination frame (Eq. 11) from the central frame of the contaminated spectrum, resulting in

$$\hat{T}^c = M^c - \hat{S}^c. \tag{12}$$

To minimise the weighted mean squared error (MSE) between the enhanced frame and the actual centre frame of the target spectrum, the network is optimised using stochastic gradient descent (SGD) with a learning rate of 0.1,

$$L = ||(\hat{T}^c(f) - T^c(f)) \times w(f)||^2, \tag{13}$$

where $f \in [1, F]$ stands for the frequency bin in the target frame and

$$w(f) = 2 - \frac{f}{F}. \tag{14}$$

In this way, bigger weights are allocated to the lower frequencies in order to protect the audio components that have the greatest impact on speech intelligence.

The traditional training objective for audio enhancement, such as MSE, is to minimise the difference between the enhanced and clean spectrograms. It is successful when the enhancement performance is assessed using evaluation metrics that measure frequency-domain audio distortions. However, the importance of phase information is neglected while reconstructing the enhanced audio, which has recently been highlighted as a concern for low SNR circumstances, such as when audio are recorded in a far-field condition. A few recent efforts have created time-domain loss functions, obviating this concern. A noteworthy example is the weighted signal-to-distortion ratio (wSDR) [58], which directly minimises the difference between the clean and enhanced waveform, so encouraging the model to circumvent the magnitude and phase decomposition. The primary focus throughout the development and implementation of N-HANS was not the hunt for a more effective loss function, but rather the search for a neural network architecture to unify numerous enhancement objectives. Applying time-domain losses to N-HANS has the potential to further refine its enhancing performance, and is projected to improve the hearing experience in relation to phase perturbations.

3.3.3 Experiments & Evaluation

This section begins with the introduction of the data sets utilised for training our N-HANS models, as well as the general data processing procedure to create model inputs. Then, for each audio enhancement functionality, we detail our experiments and compare their performance to that of other current approaches. These approaches may be developed with other datasets and assessed using alternative performance measures, which will be introduced individually in Sections 3.3.3.3 to 3.3.3.6.

Table 3.3.4: Overview of data sets for N-HANS training. Denoising (DE), speech separation (SS), selective noise suppression (SNS), and voice suppression (VS).

Task	Audio of Interest	Noise or Interference
DE	Librispeech [145]	Audioset [146]
SNS	Librispeech [145], Audioset [146]	Audioset [146]
SS	VoxCeleb 1/2 [147, 148]	VoxCeleb 1/2 [147, 148]
VS	DCASE 2019 challenge	Librispeech[145]

3.3.3.1 Data Description

We perform our study mainly using the standard data sets listed in Table 3.3.4 to train our N-HANS models and assess their effectiveness.

Selective Noise Suppression

To train the model for selective noise suppression and denoising, we synthesised a large and diverse in-the-wild speech data set by mixing up each clean utterance from the LibriSpeech corpus [145] with positive and negative noises, two distinct recordings randomly picked from the AudioSet database [146]. LibriSpeech [145] consists of approximately 1 000 hours of read, clean speech derived from over 8 000 public domain audiobooks, with its own train, development, and test splits. The AudioSet corpus [146] comprises more than two million human-labeled 10-second environmental sound clips extracted from YoutTube videos. After excluding all noise recordings labelled as 'human sounds' according to the provided AudioSet's ontology, we obtained 16 198 samples for the training set, 636 samples for the development set, and 714 samples for the test set. The random selection of the positive and negative noise recordings are from two different categories in AudioSet, in order to maximally create more diverse environmental conditions[6].

By removing the excess signal tails, a positive noise, negative noise, and clean utterance are trimmed to the same length. To create contaminated audio for training, the positive and negative noises are then added to the utterance with two SNRs, i. e., SNR(+) for the positive noise and SNR(−) for the negative noise, randomly selected from $-3, 0, 1, 3, 5, 8$dB. The validation and test sets were generated by combining a clean utterance with environmental sounds using all possible permutations of the SNR^+ and SNR^- mentioned above. During model training, a broader range of SNRs was considered to ensure its robustness to test data. The test and validation sets were created a single time, and were uniform across all experiments.

Source Separation

For source separation, the model is developed based on the merging of the two versions of VoxCeleb [147, 148]. It contains over one million utterances collected from Youtube interviews with more than 7 000 speakers of various nationalities. Each utterance lasts 4 to 12 seconds, resulting in a total of more than 2 000 hours of single-channel audio recordings. Since the training and test partitions of VoxCeleb1 and VoxCeleb2 are from distinct speakers, the training and test sets were created by combining the two corresponding sets from

[6] The data partitioning of AudioSet to reproduce our experimental results can be found in the Github repository of N-HANS: `https://github.com/N-HANS/N-HANS`.

both versions.

For each training iteration, two speakers, a *target speaker* and an *interference speaker*, are selected at random from the data set, and one utterance is taken from each speaker. The two utterances are thus labelled as the *target utterance* and the *interference utterance*, respectively. To make a *contaminated audio*, the two utterances are cut to the same length, and subsequently, the interference utterance is added to the target utterance with a random SNR within the range $-5, 0, 5, 10, 15, 20, 25$dB. For creating the test set, a more constricted SNR in the range between -5dB and 5dB), i. e., either $-5, -3, -1, 0, 1, 3$ or 5dB, were used to mix up the target and interference utterances. The SNR range considered during the creation of the test set is restricted to $-5, -3, -1, 0, 1, 3, 5$dB for two reasons. On the one hand, this can ensue a fair comparison between our method and the past work such as [65]; and on the other hand, it can promote the model's capacity to deal with more challenging real-world scenarios. Again, the validation and test sets for source separation were generated once and used uniformly across all trials.

Unlike the Wall Street Journal (WSJ0) corpus [149] and the TIMIT corpus [150] used in prior work [65, 67, 151], our created dataset encompasses a significantly broader and diversified collection of real-world situations. Therefore, it promotes a more accurate understanding of the model performance in actual use.

Voice Suppression

The objective of voice suppression is to eliminate speech from the background, reversing the roles of speech and its background. We train our voice suppression model using data from the DCASE 2019 SubTask 1A challenge and Edinburgh speech database [152]. The DCASE database comprised 40-hours of stereo environment samples captured with the same device in 10 acoustic scene classes in ten cities. The data was divided into 9 185 segments for training and 4 185 for evaluation, with each segment lasting 10 seconds. Speech recordings from the Edinburgh speech database [152] are used to simulate the presence of human voice in scene recordings. The data collection contains clean spoken utterances of 56 speakers (28 female and 28 male) from Scottish and American accent areas. Approximately 400 utterances are accessible for each speaker.

To account for varying levels of human voice interference, during training, we combined the scene recordings with the spoken utterances using a random SNR within the range [-20, -10 -5 ,0, 5, 10, 15] dB. For testing purpose, we measure the performance at each of these SNRs.

3.3.3.2 Data Processing

As mentioned in Section 3.3.1, the Enhanced Network processes the raw input i. e., original audio file to be enhanced, conditioned on the positive and negative recordings. The additional positive and negative contexts are not required to be included in the contaminated audio.

The general approach of data processing is illustrated in Figure 3.3.3 taking selective noise suppression as an example. To generate the data for training, recordings of speech utterance, positive noise and negative noise are trimmed to the same length by removing signal excess. The creation of the noisy audio is accomplished by mixing up signals of clean speech with noise. Using STFT with a window size of 25 ms and a hop size of 10 ms, the log magnitude spectra are respectively extracted from the noisy audio, positive and negative recordings. Since the sampling frequency of all audio files within the used data

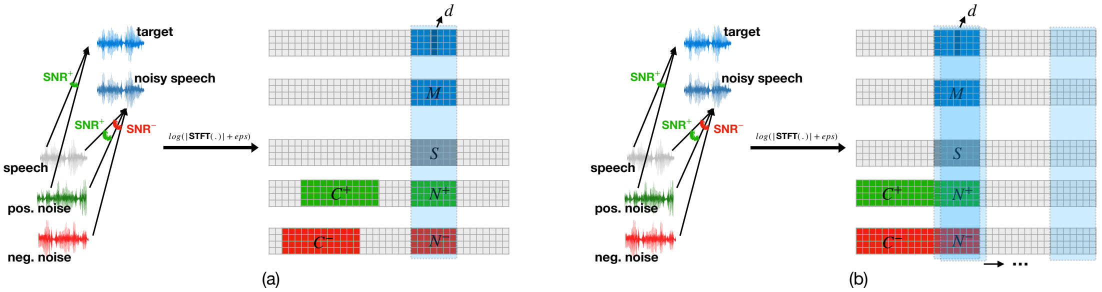


Fig. 3.3.3: Data processing (a) for training, (b) for evaluation. The noisy speech is generated by synthesising clean speech, positive and negative noise with SNR$^+$, SNR$^-$. The target is the composition of the clean speech and the positive noise. $M, S, N^+, N^- \in \mathbb{R}^{35\times 201}$ indicate the noisy segment, speech segment, positive and negative segment, respectively. $C^+, C^- \in \mathbb{R}^{100\times 201}$ stand for the positive and negative context. d represent the target frame, which we attempt to estimate from a noisy segment.

sets is 16 kHz, each segment of 400 sample points is converted into a frame vector of 201 frequencies. We denote the number of frequencies of a frame as F, considering that the N-HANS model may process audio files with a different sampling rate.

The contaminated segment $M \in \mathbb{R}^{N\times F}$ consists of N frames from the log magnitude spectrum of the contaminated audio, whilst the positive and negative contexts, $C_+, C_- \in \mathbb{R}^{L\times F}$, are of L frames retrieved from the log magnitude spectrum of the positive and negative recordings, respectively. In our experiments, we set $L = 200$ greater than $N = 35$ on the assumption that more informative positive and negative contexts can result in better enhancement performance. Therefore, in order to better imply audio content of interest and that to be removed, adequate acoustic information should be supplied to the system. The target segment $T \in \mathbb{R}^{N\times F}$, which has the same length of the contaminated segment, represents the ideal output segment for enhanced audio. For selective noise suppression, the target segment comprises both the speech component and desired positive noise. The centre frames of the contaminated segment and target segment are denoted by M^c and $T^c \in \mathbb{R}^{1\times F}$, respectively.

For inference, the positive and negative contexts are truncated from the beginning of the positive and negative spectra, as seen in Figure 3.3.3(b). Both are consistently employed to assist in the processing of all noisy segments created from the same noise recordings. In this way, each enhanced frame can take use of the same information from positive and negative contexts. The enhanced frames are concatenated into the enhanced spectrum, which is then transformed into the enhanced audio using inverse Short-Time Fourier Transform (iSTFT) with the phase of the contaminated audio.

3.3.3.3 Functionality I: Selective Noise Suppression

Comparison Methods

To the best of our knowledge, selective noise suppression is explored for the first time in N-HANS. In this case, to measure the efficacy of our developed model, we establish a baseline model with the same architecture as N-HANS model, except that it solely conditioned on the negative contexts. Only the negative auxiliary network is used to learn a negative noise embedding specifying the noise to be suppressed. Meanwhile, the specifications of

Table 3.3.5: N-HANS test results for the selective noise suppression task. SNR(+) and SNR(−) represent the Signal-to-Noise ratio (SNR) applied to the positive and negative noises, respectively. For each condition, i. e., a pair of SNR(+) and SNR(−), the following evaluation metrics are given: log spectral distortion (LSD), signal-to-distortion ratio (SDR), perceptual evaluation of speech quality (PESQ), short-time objective intelligibility (STOI), Mel cepstral distortion (MCD), and segmental SNR (SSNR). For comparability, the performance of the baseline model is given in parentheses. (*Source*: [34])

SNR							
(+)	(−)	**LSD**	**SDR**	**PESQ**	**STOI**	**MCD**	**SSNR**
0dB	0dB	0.76 (0.91)	7.72 (6.38)	2.86 (2.70)	0.79 (0.76)	5.36 (5.45)	7.13 (5.51)
	3dB	0.69 (0.83)	9.49 (8.07)	3.09 (2.89)	0.84 (0.81)	4.96 (5.10)	8.55 (6.76)
	5dB	0.65 (0.76)	10.64 (9.35)	3.23 (3.09)	0.87 (0.85)	4.67 (4.65)	9.46 (7.75)
	8dB	0.59 (0.69)	12.12 (11.00)	3.40 (3.28)	0.90 (0.88)	4.29 (4.23)	10.83 (8.88)
3dB	0dB	0.78 (0.92)	7.16 (5.96)	2.78 (2.61)	0.78 (0.75)	5.58 (5.69)	6.56 (5.06)
	3dB	0.73 (0.82)	8.93 (7.81)	2.98 (2.84)	0.83 (0.81)	5.29 (5.25)	7.79 (6.37)
	5dB	0.68 (0.79)	10.06 (8.82)	3.12 (2.98)	0.85 (0.83)	4.97 (5.03)	8.90 (7.19)
	8dB	0.64 (0.72)	11.46 (10.56)	3.29 (3.19)	0.88 (0.87)	4.68 (4.64)	9.93 (8.55)
5dB	0dB	0.81 (0.93)	7.19 (5.80)	2.74 (2.57)	0.78 (0.75)	5.71 (5.87)	6.30 (4.73)
	3dB	0.75 (0.87)	8.68 (7.61)	2.93 (2.76)	0.82 (0.79)	5.44 (5.57)	7.44 (5.91)
	5dB	0.72 (0.82)	9.76 (8.67)	3.06 (2.90)	0.84 (0.82)	5.28 (5.34)	8.27 (6.69)
	8dB	0.65 (0.73)	11.33 (10.36)	3.26 (3.11)	0.88 (0.86)	4.83 (4.86)	9.85 (8.20)
8dB	0dB	0.86 (0.98)	7.01 (5.63)	2.67 (2.48)	0.77 (0.74)	5.99 (6.06)	5.83 (4.04)
	3dB	0.79 (0.89)	8.63 (7.35)	2.86 (2.68)	0.81 (0.79)	5.71 (5.70)	7.07 (5.40)
	5dB	0.74 (0.92)	9.62 (6.38)	2.99 (2.68)	0.84 (0.76)	5.44 (5.54)	7.92 (5.46)
	8dB	0.68 (0.78)	11.30 (10.47)	3.18 (3.02)	0.87 (0.86)	5.09 (5.14)	9.36 (7.77)

the enhanced network and its negative auxiliary network are identical to the N-HANS (cf. Table **??**). We optimised the baseline model for the purpose of selective noise suppression based on exactly the same data. Thus, the only difference between our presented model and the baseline is the use of the positive embedding. Similar methods have proven effective for noise-aware speech enhancement [52, 153].

Evaluation Methods

We assess the performance of our selective noise suppression model using a variety of evaluation metrics that are widely used in prior work [154], including Log Spectral Distortion (LSD), Signal-to-Distortion Ratio (SDR), Perceptual Evaluation of Speech Quality (PESQ), Short-Time Objective Intelligibility (STOI), Mel Cepstral Distortion (MCD), and Segmental Signal-to-Noise Ratio (SSNR), in terms of numerous Signal-to-Noise Ratio (SNR) conditions.

Results Analysis

As the experimental results given in Table 3.3.5, the N-HANS model outperforms the baseline model for all the evaluation metrics considered and SNR combinations in the selective noise suppression task. We attribute the performance gains over the baseline architecture to the introduction of the complementary auxiliary network that captures more information from the positive noise context. The baseline model, on the other hand, is agnostic to the noise sources it should preserve and ends up removing parts of them due to its inability to discern between audio contents that should be preserved and those to be removed. This comparison demonstrates the need of using a positive embedding for more effective selective noise suppression.

In fact, the model performs best for the given task when the speech environment contains more energy for the positive noise than for the negative, i. e., the lower SNR for the positive noise, or higher SNR for the negative noise. As expected, the performance of our model degrades as the strength of negative noise grows (lower SNR(−)) or the power of positive noise reduces (higher (SNR+)). Given the lowest SNR on the positive noise (i. e., 0dB), a higher SNR on the negative noise resulted in improved performance across all the evaluation metrics. This is very likely because the increased the intensity difference between the positive and negative noises provides an extra cue for their discriminate; As a consequence, less negative noise is more easily to be removed by the model while the positive noise is consistently protected.

The fact that SSNR gains are reduced as the SNR(+) rises while the LSD and SDR metrics remain constant or even worsen indicates that our approach performs aggressive denoising, whereby distortion effects dominate the noise reduction gains, leading to worse subjective performance (as measured by STOI and PESQ) for higher SNR(+).

3.3.3.4 Functionality II: Audio Denoising

Supplying silent audio as the positive context to the +A Network of the N-HANS selective noise suppression model, it turns into an environment-aware speech denoising system capable of adapting to unseen environments. Such a denoising model benefits from identifying the speech surroundings as presented in [155, 156].

Comparison Methods
The performance of N-HANS denoising system is compared to that of several state-of-the-art approaches, including SEGAN [56], Wavenet [70], MMSE-GAN [157], and DCUnet-20 [58]. In addition to testing our model with the test set of the same databases used for training, we conduct additional tests using two publicly available databases: The Diverse Environments Multichannel Acoustic Noise Database (DEMAND) [158], and the Voice Bank corpus [159], both of which are employed in the comparison methods. The DEMAND database provides recordings of real-world noise in a variety of settings, and the Voice Bank corpus collected more than 300 hours of English recordings from approximately 500 healthy speakers.

Evaluation Methods
First, we test the N-HANS denoising model on the test sets prepared using LibriSpeech and AudioSet in terms of the same performance measures for selective noise suppression. To compare with the other speech enhancement methods, we consider PESQ, SSNR, and three other evaluation metrics defined in [160], i. e., CSIG, CBAK and COVL, indicating the Mean Opinion Score (MOS) predictor of signal distortion, background-noise intrusiveness, and the overall signal quality, repectively.

Results Analysis
Our N-HANS denoising model is capable of producing audio output with comparable LSD, SDR, and MCD performance to previous speech enhancement systems [154, 161], accord-

Table 3.3.6: Test results for the speech denoising task with N-HANS trained on the LibriSpeech and AudioSet corpora considering the Evaluation Metrics: LSD, SDR, PESQ, STOI, MCD, and SSNR. (*Source*: [34])

SNR	**LSD**	**SDR**	**PESQ**	**STOI**	**MCD**	**SSNR**
0dB	1.17	7.02	2.49	0.81	6.79	4.06
3dB	1.10	8.72	2.70	0.84	6.51	5.10
5dB	1.05	9.60	2.84	0.86	6.40	5.90
10dB	0.93	11.86	3.12	0.90	5.98	7.80
15dB	0.84	13.35	3.34	0.92	5.49	9.58

Table 3.3.7: Test results for the speech denoising task with SEGAN, Wavenet, MMSE-GAN, DCUnet-20, and N-HANS considering the Evaluation Metrics: CSIG, CBAK, COVL, PESQ, and SSNR (cf. the caption of Table 3.3.5). For N-HANS, results are given considering the Librispeech and AudioSet corpora (Train 1), as well as Voice Bank and DEMAND (Train 2), for training. Note that the other evalauted methods are trained with Voice Bank and DEMAND, thus, results for Train 2 enable a fairer comparison. (*Source*: [34])

	CSIG	**CBAK**	**COVL**	**PESQ**	**SSNR**
SEGAN	3.48	2.94	2.80	2.16	7.73
Wavenet	3.62	3.23	2.98	–	–
MMSE-GAN	3.80	3.12	3.14	2.53	–
DCUnet-20	4.24	4.00	3.69	3.13	15.95
N-HANS (Train 1)	3.60	2.84	2.83	2.05	6.42
N-HANS (Train 2)	4.00	3.18	3.23	2.44	8.24

ing to an evaluation on the synthesised data using the LibriSpeech and AudioSet corpora (cf. Table 3.3.6). Moreover, our system yielded to high STOI values for all the examined SNR conditions, showing strong speech intelligibility even for the lower SNR.

To ensure a more equitable comparison with different SE methods, the data of DEMAND and Voice Bank corpus are partitioned in accordance to Choi et al. [58]. Since the functionality of speech denoising is a byproduct of the selective noise suppression system and the model is trained with LibriSpeech and AudioSet data, the testing performance has not yet been optimised for the Voice Bank and DEMAND corpora. In spite of this, the N-HANS denoising model is able to achieve comparable results w. r. t. state-of-the art approaches (cf. Table 3.3.7). Our method performs slightly better than SEGAN in terms of CSIG and COVL, indicating superior signal quality and less distortions. Regrading the remaining three performance measures, however, N-HANS cannot outperform SEGAN in reducing noise signals selected from DEMAND corpus.

Transfer learning can be used to adjust the denoising model to the DEMAND and the Voice Bank corpus, hence improving its performance on the test set for all of the evaluation metrics considered. Despite the overall gains, our denoising model trails behind of the model using a DCUnet-20 architecture, which is tailored for boosting the hearing experience using the loss function of wSDR [58]. Negative aspects of this training objective include the necessary duration of the model input signal. Indeed, this requirement might affect the real-time factore (RTF) of the model for inference. Though applying wSDR and employing lengthier input has the potential to further improve the evaluation performance

Table 3.3.8: Test results for the speech separation task considering the evaluation metrics signal-to-distortion ratio (SDR), signal-to-artifacts ratio (SAR), and signal-to-interference ratio (SIR), for the baseline with Deep Clustering (DC), Conv-Tasnet, and N-HANS methods. Results for combining female (f) and male (m) speakers are given, followed by their overall average (all). (*Source*: [34])

	SDR				**SAR**				**SIR**			
Method	**f+f**	**m+m**	**f+m**	**all**	**f+f**	**m+m**	**f+m**	**all**	**f+f**	**m+m**	**f+m**	**all**
`DC`	3.05	2.52	4.33	3.48	5.73	5.32	6.59	6.00	9.46	9.05	11.31	10.21
`Conv-Tasnet`	4.18	6.16	6.76	6.38	8.29	9.10	10.46	9.94	8.12	9.75	10.83	10.19
N-HANS	**7.62**	**8.92**	**10.58**	**9.47**	**8.61**	**9.71**	**11.04**	**10.15**	**17.13**	**18.93**	**22.48**	**20.21**

of N-HANS, we persist to use wMSE (cf. Eq.13) to guarantee real-time processing for more realistic applications.

3.3.3.5 Functionality III: Source Separation

Comparison Methods

To evaluate the efficacy of N-HANS for the task of source separation, we implemented two baseline source separation algorithms, Deep Clustering (DC) [149, 150] and Conv-Tasnet [67], using our created data based on VoxCeleb. Both approaches adhere to the conventional problem formulation for source separation and are not dependent on additional context recordings.

The deep clustering model [149] exploits a deep recurrent neural network with two bidirectional LSTM layers to process the spectrogram of an audio of overlapping speech. The model is trained to generate similar embeddings for TF-bins originating from the same speaker. Subsequently, based on these embeddings, a clustering technique is used to assign the TF-bins to different speakers . The Conv-Tasnet [67] uses an encoder-decoder framework for end-to-end time-domain speech separation, wherein the encoder, a temporal convolutional network (TCN) consisting of stacked 1-D dilated convolutional blocks, is trained to learn a speech representation optimised for separating speakers, and the linear decoder is used to transform the speech representation back to waveform.

Evaluation Metrics

Three objective evaluation metrics, signal-to-distortion ratio (SDR), signal-to-artefacts ratio (SAR), and signal-to-interference ratio (SIR) have been implemented in the BSSEval toolbox [162], and are well-known to be effective in evaluating source separation models [163]. As their names imply, SDR, SAR and SIR assess the composition of the target speech in terms of the distortion, artefacts and interference left in the processed audio caused by the interfering speaker.

Results Analysis

Our N-HANS speech separation system is compared to the baselines for two female speakers (f+f), two male speakers (m+m), and speakers of different genders (f+m). In addition, we include the overall results for both speakers in Table 3.3.8 (all).

The testing results indicate that N-HANS can significantly outperforms the DC baseline in a two-tailed t-test, yielding a $p < .0004$ for the SDR, SAR, and SIR measures. Concerning the Conv-Tasnet baseline, N-HANS also presents a significant improvement in terms of SDR and SIR: $p < .008$. Although the performance of our N-HANS source separation system can surpass Conv-Tasnet regarding SAR, the improvement is not statistically significant: $p = .558$. Intriguingly, although the baseline methods achieved strong separation results on the WSJ0 and the TIMIT corpus [149, 150], their performance worsened when confronted with the VoxCeleb corpus, which is more complicated than the prior ones. Indeed, the higher performance of N-HANS on this more challenging dataset increases our confidence in the robustness of applying the presented system to real-world conditions.

An further insight arising from Table 3.3.8 is that all three separation models perform better on the speakers of different genders than the speakers of the same gender. This is because speech from speakers of the same gender share similar acoustic features, making it more difficult to separate the mixed spectrum. The results obtained from speakers of the same gender to those acquired from speakers of different genders reveal a larger average performance difference between N-HANS and the baseline methods for the three evaluation criteria. This discovery leads us to the conclusion that, particularly in challenging cases, conditioning a source separation model on additional contexts can successfully inject useful information that improves the separation process. Moreover, our N-HANS source separation method circumvents the notorious problem of label permutation [164]—a research question that has consumed a substantial amount of human efforts [150]. Some recent studies [149, 165, 164] have sought to overcome this issue. Our solution tackles this issue by learning the extra target and interference recordings. As a consequence, the enhanced network is able to determine the speaker labels, i. e., 'target' or 'interference', enabling the separation of a mixture speech without the label permutation problem.

3.3.3.6 Functionality IV: Voice Suppression

Acoustic Scene Classification Models

The ultimate aim of our voice suppression system is to improve the speech robustness of ASC models. For this, we consider two ASC architectures, i. e., the official 2019 DCASE baseline and attentive atrous CNN [166]. The specifications of these two models appear in Table 3.3.9 and Table 3.3.10.

The DCASE baseline model is a two-layer CNN in which convolution, batch normalisation, ReLU activation, dropout and max-pooling are sequentially applied to each layer. Two dense layers are utilised to project the convolution outputs onto the classes. As model input, we extract log mel spectra from the audio waveform using a frame size of 40 ms and hop size of 20 ms, yielding a 40×500 Mel-band spectrum. On the official test set of the 2019 DCASE challenge, the baseline model achieves a classification accuracy of 62.20 %.

The second CNN model employs atrous convolution and spatial attention mechanism, which has been shown to be effective for the ASC tasks [166]. An atrous CNN incorporates dilation settings, a tweak that controls the spacing between the convolutional kernel points [167], to replace the need of pooling layers, hence expanding the receptive field for each convolutional layer in comparison to the normal CNN design. Our attentive atrous CNN consists of four atrous convolutional layers, 2D attention values are learnt and assigned to each pixel in the feature maps. The attentive feature maps are then averaged across all locations (time steps and frequency bins) and projected onto the scene labels using a fully-connected layer. The classification accuracy can reach 69.0 % on the test set of the 2018 DCASE challenge and 77.51 % for the same challenge in 2019.

Table 3.3.9: Specifications of DCASE2019 baseline model. (*Source*: [134])

Block	**Kernel**	**#Ch_input**	**#Ch_out**
conv1	$(7,7)$	2	32
pool1	$(5,5)$	32	32
conv2	$(7,7)$	32	64
pool2	$(4,10)$	64	64
flatten	–	–	128
fc1	–	128	100
fc2	–	100	10

Table 3.3.10: Specifications of attentive atrous CNN. (*Source*: [134])

Block	**Kernel**	**Dilation**	**#Ch_input**	**#Ch_out**
atrous conv1	$(5,5)$	1	2	64
atrous conv2	$(5,5)$	2	64	128
atrous conv3	$(5,5)$	4	128	256
atrous conv4	$(5,5)$	8	256	512
attention	$(1,1)$	–	512	512
global average	–	–	–	512
fc	–	–	512	10

Evaluation Methods

Data augmentation can be utilised as a straightforward method to strengthen the ASC classifiers' generalisability in the presence of speech. To this end, both classifiers are trained using scene recordings that are contaminated with speech samples from the Edinburgh database. This solution is evaluated under two conditions: *matched-SNR* and *multi-SNR*. The first case augments the training data with the same SNR as for the test data, but the second one is more realistic, i. e., instances of the training data are combined with a random SNR selected to be one of $-10, -5, 0, 5, 10, 20$ or 30 dB.

Results Analysis

First of all, we need to explore the robustness of the baseline ASC models with respect to human speech. For this purpose, we test the baseline models, which are trained on the clean training data from the 2019 DCASE challenge, with the test data from the same database that has been corrupted with speech noise from the Edinburgh database. The testing results in the "Noisy" column of Table 5.3.3) reveal that the performance of both ASC models degrades when the relative volume of the human voice to the acoustic scene grows, i. e., under stronger SNR conditions. We observe that under conditions of high SNR, the noise has negligible influence on the ASC classifiers. However, once the SNR falls to -10 dB, , i. e., the speech is 10 dB more energetic than the surrounding environment, classification accuracy declines to near chance levels, rendering both models utterly useless for the ASC task. Even at the reasonably high SNR of 10 dB, the performance declines by

Table 3.3.11: Test results for the voice suppression task. The testing accuracy[%] of two ASC models are given regarding Signal-to-Noise Ratio (SNR). (*Source*: [134])

	2019 DCASE Baseline			
SNR	**Noisy**	**Multi-SNR**	**Matched SNR**	**Denoised**
Clean	62.20	–	–	–
30 dB	61.60	41.46	59.89	61.67
20 dB	59.40	42.15	59.90	60.81
10 dB	47.38	44.87	59.42	57.35
5 dB	36.27	46.81	60.22	56.42
0 dB	25.19	49.68	58.97	57.08
-5 dB	17.11	52.21	56.27	57.71
-10 dB	14.27	53.41	57.37	57.71

	Attentive Atrous Model			
SNR	**Noisy**	**Multi-SNR**	**Matched SNR**	**Denoised**
Clean	77.51	–	–	–
30 dB	76.58	58.78	65.93	77.16
20 dB	71.95	59.93	62.22	67.00
10 dB	54.84	60.86	61.51	62.08
5 dB	41.17	61.15	63.23	60.45
0 dB	28.29	61.74	58.87	61.91
-5 dB	21.48	61.31	59.73	63.25
-10 dB	17.99	60.05	58.23	61.95

15 % and 23 %, respectively. Indeed, both models appear to be able to handle minor audio disturbances, since their performance does not degrade much for SNRs up to 20 dB. This inspires us to apply a voice suppression system to reduce the speech noise content in the scene recordings to a level that an ASC model can manage.

Moreover, the results in Multi-SNR and Matched SNR columns of Table 5.3.3 reveal that data augmentation can recover most of the lost accuracy owing to the mismatch between training and testing conditions. The baseline model performs better under the conditions of all the matched SNR cases than under the multi-SNR conditions. Besides, we find the attentive atrous CNN can perform better with multi-SNR settings than with matched SNR under the low SNR situations, suggesting that training with several distinct SNRs provides a generalisation advantage.

In general, our developed N-HANS voice suppression model can successfully support the ASC classifiers considered in obtaining higher classification accuracy for the large SNR cases (cf.. the "Denoised" column of Table 5.3.3). When only little quantity of human speech appear in the recordings of noisy scenes, such as for SNRs exceeding 20 dB, the original ASC classifiers function similarly as for the clean recordings. Using voice suppression as forefront

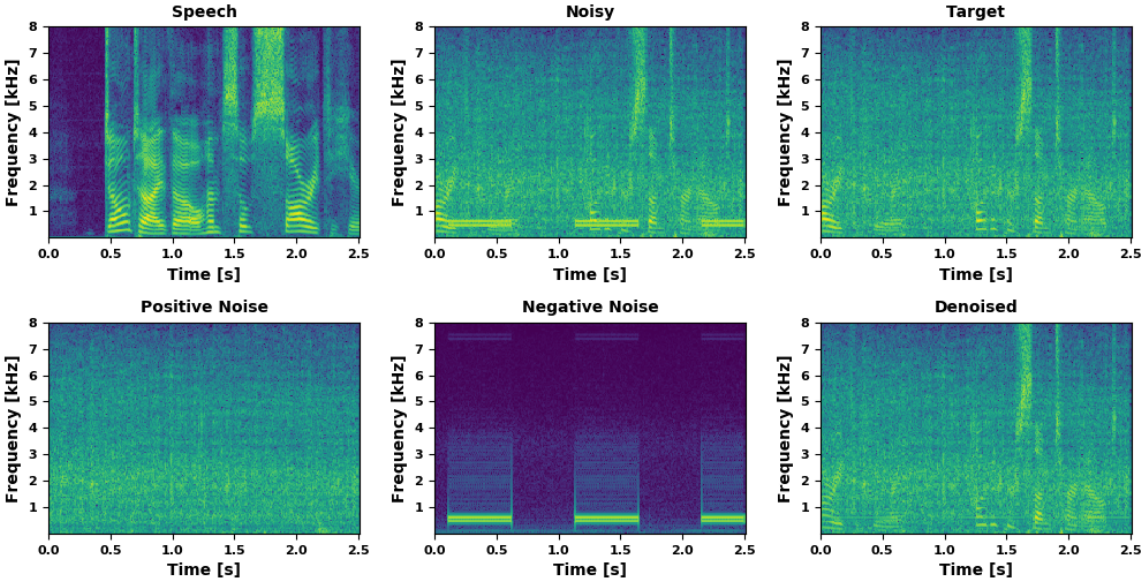


Fig. 3.3.4: Spectrograms illustrating the audio components involved in the N-HANS selective noise suppression system, i. e., the clean spoken utterance (speech), the contaminated audio (noisy), the ideal result (target), the negative and positive noises, and the achieved outcome (denoised). Positive noise: train; Negative noise: telephone busy signal. (*Source*: [34])

processing, the baseline model attains average performance across all SNRs, resulting in a classification accuracy of around 60 %. Nonetheless, the attentive atrous CNN performs less stable across these SNRs. As the speech level rises, the classification performance declines. Using voice suppression, however, can mitigate this impact and boost the scene classification performance for the contaminated recordings by a great margin.

The ASC classifiers are incapable of achieving the performance of clean scene recordings despite the use of voice suppression as a front-end. Although we aim to suppress the human voice in noisy scene recordings to its greatest possible extent, in fact some residual speech components remain in the processed scene audio. The speech residue sometimes turns into fizzer noises that are not always audible, but are deemed detriment to ASC. Moreover, the voice suppression system seeks to eradicate all speech components in scene recordings, and thus maybe it is too aggressive towards the scene context itself, resulting in an unwanted loss of environmental information if the scene recordings include any useful voice.

3.3.4 Performance Visualisation

We end this section with some selected examples of the performance of our four N-HANS audio processing functionalities, selective noise suppression in Figure 3.3.4, speech denoising in Figure 3.3.5, speech source separation in Figure 3.3.6, and voice suppression Figure 3.4.7, respectively. Further examples for the performance visualisation can be found in original articles [34, 134].

Selective Noise Suppression

Figure 3.3.4 depicts the procedure of N-HANS selective noise suppression. Provided are the spectra of a sample of the clean spoken utterance, the noisy audio (mixture of the

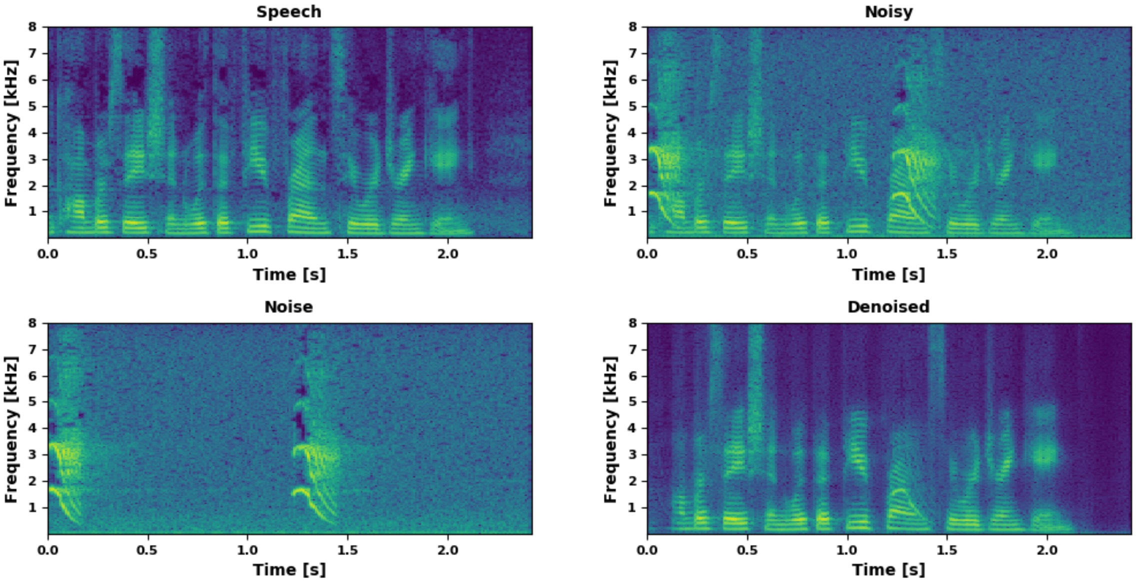


Fig. 3.3.5: Spectrograms illustrating the audio components involved in the N-HANS denoising system, i. e., the clean spoken utterance (speech), the contaminated audio (noisy), the interfering noise, and the achieved outcome (denoised). Bird song. (*Source*: [34])

clean utterance, positive and negative noises), the target (mixture of the clean utterance and the positive noise), the positive and negative noises, and the denoised sample (the system output). The objective of N-HANS selective noise suppression system is to remove the negative noise from the noisy spectrum. Therefore, it is anticipated that the output should be a near estimate to the target spectrum which comprises of speech and positive noise. In this example, we can find that our model is able to recover speech signals under the heavy noise scenario, as the speech components obscured by the negative noise in the noisy spectrum resurface in the denoised output. The successful selective suppression for non-stationary noise shown in this example should reduce the concerns over its capability to handle more stable wide-band stationary noise. Because these kinds of noise contain more consistent noise context, and hence supplying the system affluent indication of the noise to be suppressed.

Denoising

The effect of the N-HANS denoising system is shown in Figure 3.3.5. Provided are the spectra of an example of the clean utterance, the noisy utterance, the background noise, and the denoised sample (the system output). In this example, an utterance is submerged into a severe industrial noise at an SNR of 0dB; nonetheless, the system can still effectively remove the majority of noise components, hence improving the speech quality. The denoising model is capable of removing the noise based on the identification of additional noise contexts, such as those non-continuous noises characterised by isolated impulses in this example. Note that when the speech environment includes noise with similar acoustic properties to human voice, the system may distort the estimated speech spectrum to suppress the noise as much as possible. Yet, these distortions are not that disruptive to normal human hearing perception.

Speech Separation

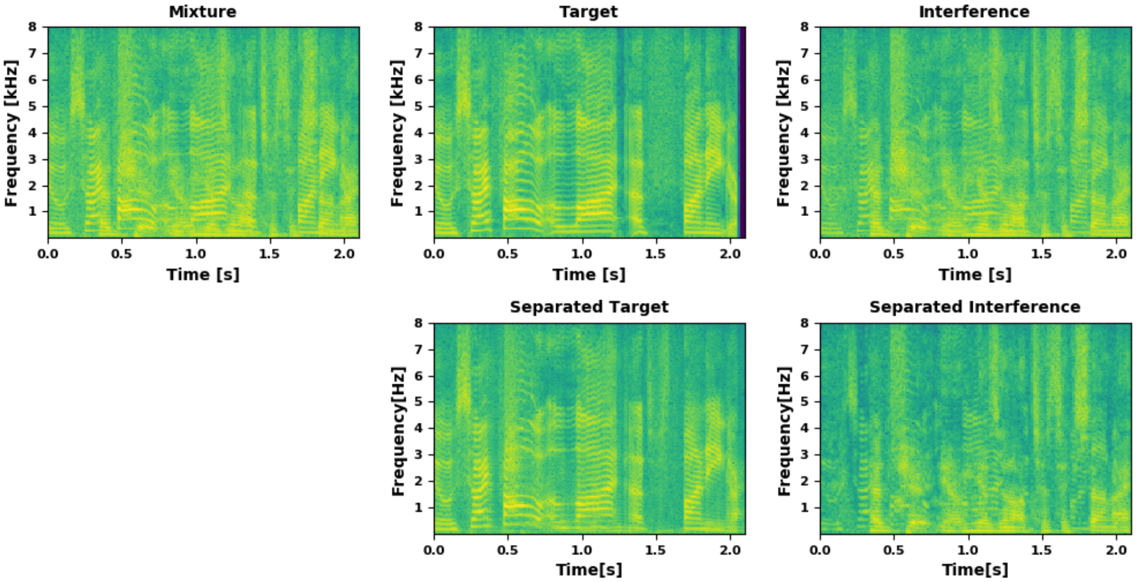


Fig. 3.3.6: Spectrograms illustrating the audio components involved in the N-HANS source separation system, i. e., the mixture between the two speakers, and the target and interference speakers before (above) and after (below) to be separated by the system. Target speaker: id04656 (female); Interference speaker: id04232 (male). (*Source*: [34])

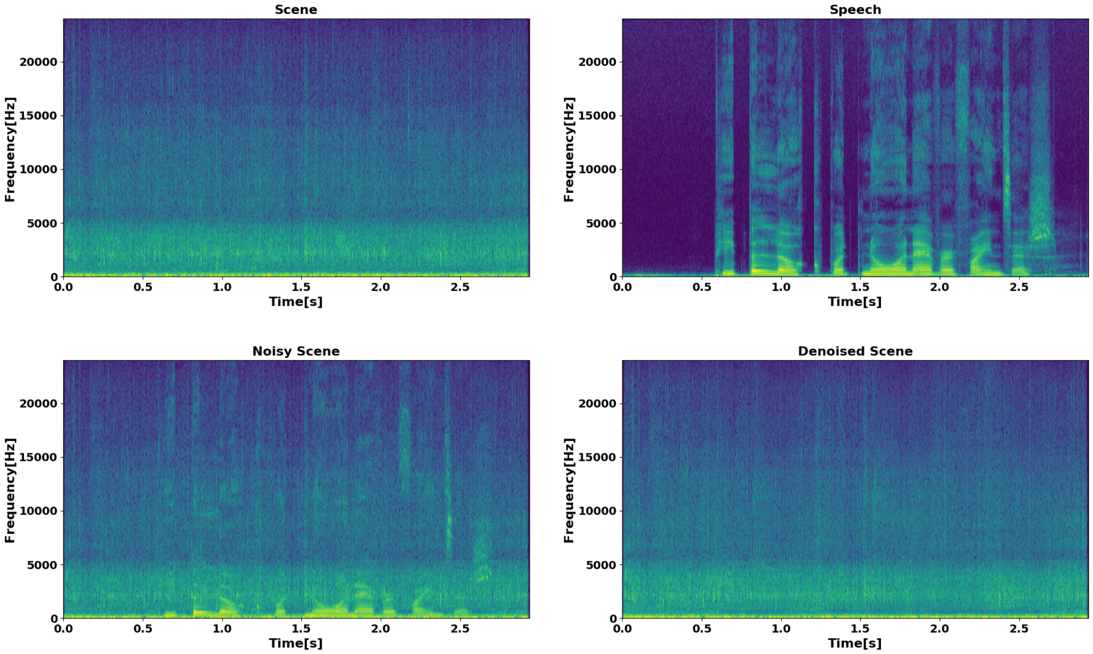


Fig. 3.3.7: Spectrograms illustrating the audio components involved in the voice suppression system, i. e., clean scene audio (scene), clean spoken utterance (speech), the contaminated audio (noisy scene), and the achieved outcome (denoised scene). (*Source*: [134])

Similarly, the example in Figure 3.3.6 depicts an overlap of two utterances from two speakers of different genders. This mixture signal is processed using N-HANS source separation model, which can successfully produce the separated target and interference speech. Please refer to [34] for For the treatment of two speakers of the same gender, . Although the target speech is notably distorted at 0.8 s and 1.5 s, the system is still powerful enough to compress the interfering sounds to a great extent. Another interesting observation arising from the visualisation is that despite the target utterance's high resolution at the low-frequency range (below 1 kHz), which is smeared by the interference speech seen in the mixture spectrum, the system can jointly estimate the amounts of speech components in each time-frequency bin, resulting in the recovered target speech with high clarity.

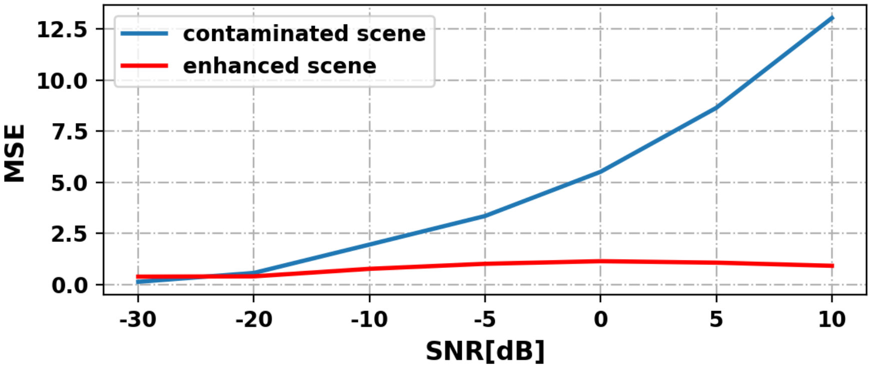


Fig. 3.3.8: Deviation between the extracted log mel-band energies from denoised scene audio and clean scene over different SNR. (*Source*: [134])

Voice Suppression

Figure 3.3.7 illustrates the efficiency of the N-HANS voice suppression system on a contaminated scene recording by a spoken utterance. Using our method, the speech components are explicitly suppressed from the recording of the noisy scene, while the environmental sounds are well protected.

To illustrate the effect of voice suppression, we compare the deviation of the log Mel-band energies extracted from the enhanced scene audios from that extracted from the clean scene audios in Figure 3.3.5. The deviation is quantified in terms of the average mean square error (MSE) between the two feature sets. The blue curve represents the difference between the noisy and clean scene signals, while the red curve stands for the deviation between the enhanced and clean scene signals. The presence of human voice has less of an effect on the feature sets derived from the enhanced scene recordings, particularly for the original noisy scenes with low SNRs.

3.3.5 Section Summary

In this part, we presented the N-HANS audio enhancement toolkit, which focuses on employing a single neural network architecture to manage multiple enhancement goals corresponding to diverse practical circumstances. Based on the concept of auxiliary network, our overall neural network makes use of the propose deep fusion approach to allow the enhancement processing conditioned on extra audio samples that specify the audio contents to be retained or eliminated. The success of using audio enhancement method to improve the performance of the following audio application, specifically the use of voice suppression for the speech robustness of ASC models, inspired us to generalise this framework to a broader range of computer audition tasks and explore solutions to optimise the audio enhancement for a target audio application, which will be discussed in the next section.

3.4 Audio Enhancement for Computer Audition

Audio enhancement has the objective to separate the audio of interest from background noise and interfering sounds, and it is widely employed at the front-end of a computer audition (CA) model to ensure the quality of the record audio. Within this framework, we rely on the enhancement processing to manage the uncertainty in audio recordings that may contain ambient noise from daily life and speech interference, etc. By doing so to improve the reliability and robustness of the subsequent audio applications such as ASR [168], SCR, SER, ASC. However, to fulfil the demands on audio quality and intelligibility for numerous audio tasks, an audio enhancement model, which is often developed independently, should be pretty versatile. Integration of such a generalised audio enhancement model with a subsequent CA model may not result in the optimal performance of the CA task. This is attributable to two factors: first, the AE model is not optimised towards its subsequent CA tasks since the loss function used has no association with the performance of the intended applications. Consequently, due to its decisive influence, the prudent selection or specific design of an AE training loss is needed, but this also requires adequate understanding of the CA applications; and second, the CA models process the enhanced output from the AE system, which may contain introduced distortions. To reduce noise to its utmost extent, the AE model may damage the audio of interest aggressively, which might affect the performance of the CA applications. To alleviate this problem, the CA models must adapt to the AE outputs to lessen the mismatch at the interface between the AE and CA models. In addition, we make use of multi-task learning to jointly optimise the AE and its subsequent CA models in order to guide the optimisation of an AE model towards the CA applications.

The successful uses of neural networks in the realm of computer vision (CV) have promoted the development of various feasible computer audition (CA) solutions. Numerous CNN architectures have been shown to be effective for audio processing, including speech enhancement. For instance, the N-HANS toolkit described in Section 3.3 is built on ResNets [2], which have an advantage in CNN model depth yet can be trained with stable convergence. Recently released U-Net [15] is an alternative CNN architecture that has proven useful for speech enhancement. U-Net was first proposed as a solution to biomedical image segmentation, but it has been expanded for numerous CA and CV problems. An U-Net is constructed based on an auto-encoder structure, with the inclusion of skip-connections to transmit information from each encoder layer to its respective decoder layer. This model architecture is appropriate for the task of speech enhancement. In addition to the original U-Net, its two variant forms, i.e., complex U-Net and wave U-Net, which circumvent the primary shortcoming of the original design, the absence of processing phase information, are also investigated. We begin by comparing the different U-Net architectures for audio enhancement in terms of some assessment measures that quantify audio distortions in enhanced audio. Subsequently, the model producing the fewest distortions is chosen for further research to enhance the audio quality for the following CA applications. We present two joint optimisation approaches that optimise the audio enhancement model for our four representative CA applications, namely ASR, SCR, SER and ASC. These applications are chosen because they span speech and non-speech, English and non-English (Italian), simple and complex classification problems.

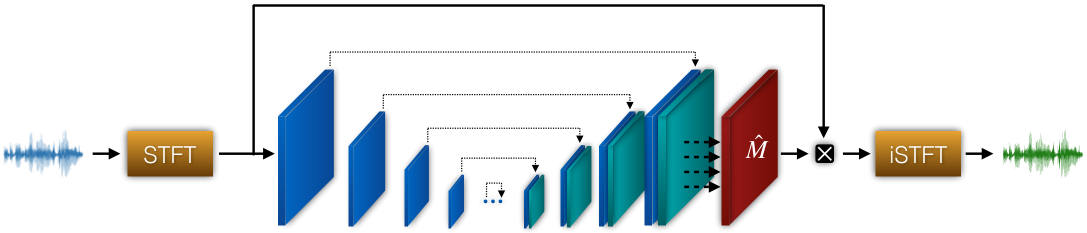


Fig. 3.4.1: U-Net architecture for speech enhancement. The noisy audio is converted to spectrogram using STFT, and then a ratio mask is estimated based on the spectrogram using U-Net, an auto-encoder architecture with feedforward connections between encoder and decoder layers. The mask is applied to original spectrogram to predict the clean spectrogram, before applying iSTFT to produce estimated clean speech.

3.4.1 Network Architectures & Training Objectives

This section describes the neural network-based models that served as the foundation for the audio enhancement and our four CA applications. These models provide cutting-edge outcomes with simple network architectures.

3.4.1.1 U-shaped Audio Enhancement Models

We investigate three architectures of U-shaped neural networks, including the original U-Net, Complex U-Net and Wave U-Net, in an attempt to analyse their potential for enhancing audio quality for our CA applications under consideration.

Audio Enhancement U-Net

Audio enhancement U-Net [169], as seen in Figure 3.4.1, takes the time-frequency representations of an audio signal as input. Only the spectrogram (magnitude spectrum) is used, while the decomposed phase spectrum is left unaltered. The network has an auto-encoder architecture with feed-forward layers that stack each encoder layer with its mirrored decoder layer. The encoder analyses the spectrogram of a noisy audio input, and decompose the audio of interest, for example speech, from the noise components into separate feature maps. The decomposition ability increases with the encoder depth. Then, the decoder recombines necessary feature maps to reconstruct the enhanced audio. Similar to ResNet, skip-connections can facilitate the retrieval of more complete information from the noisy input for the reconstruction of the desired audio.

Given a clean sample x, a spectrogram Y is generated from the contaminated audio y. The U-Net aims to estimate a ratio mask Mask$(\cdot)$, which is used to filter the original noisy audio to produce the enhanced spectrogram:

$$\hat{X} = Y \cdot \text{Mask}(Y). \tag{15}$$

Using inverse STFT, the enhanced audio $\hat{x}$ can be reconstructed with the phase information of the noisy input. The model parameters are optimised by minimising the weighted SDR (wSDR) loss of the original and the estimated clean speech and noise [38].:

$$L_{\text{SE}}(x, \hat{x}) = \alpha L_{\text{SDR}}(x, \hat{x}) + (1 - \alpha) L_{\text{SDR}}(n, \hat{n}), \tag{16}$$

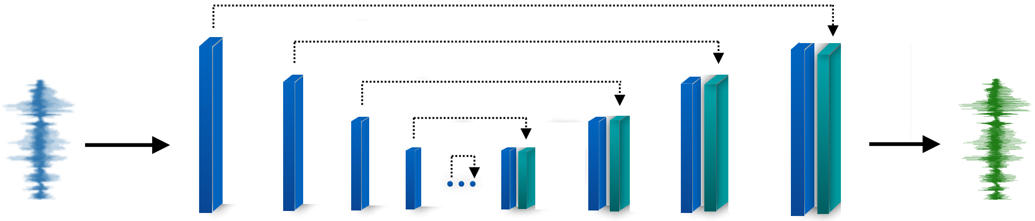


Fig. 3.4.2: Wave U-Net architecture for speech enhancement. The noisy audio is processed by a WaveNet, 1-D convolution neural network, as encoder, and another WaveNet as decoder. The output of each encoder layer is fed to the corresponding decoder layer.

where

$$n = y - x \quad \text{and} \quad \hat{n} = y - \hat{x}$$

represent the actual and estimated noise signal, and

$$L_{\text{SDR}}(x, \hat{x}) = -\frac{< x, \hat{x} >}{||x|| \cdot ||\hat{x}||}, \tag{17}$$

where $< x, \hat{x} >$ indicates the inner product of the actual clean signal and enhanced output, and

$$\alpha = \frac{||x||^2}{||x||^2 + ||n||^2} \tag{18}$$

is a hyper-parameter used to weight the importance of the audio of interest and noise during model optimisation.

Audio Enhancement Complex U-Net

Since it has been shown that phase information is crucial to the quality of the enhanced audio under low SNR conditions [58], the original U-Net, which does not process phase components, may result in a suboptimal solution. Attempts have been made to estimate or rebuild the phase spectrum of clean audio in order to remedy the deficiency of the conventional paradigm of speech enhancement problem [78, 77, 76]. Alternately, we can divide the noisy audio spectrogram into real- and imaginary-parts, and implement a Complex U-Net [38] which builds a parallel structure comprising two U-Nets, one of which handles the real part of the noisy spectrogram, and the other the imaginary part. The model generates two masks that are individually applied to the real and the imaginary parts of the noisy spectrogram to estimate the propositions of clean components. The model can be optimised using wSDR in the same manner as AE U-Net.

Audio Enhancement Wave U-Net

Another strategy for handling phase information during speech enhancement processing is to directly operate on the time-domain waveform, so avoiding the decomposition of magnitude and phase components of the audio spectrogram. Wave U-Net substitutes the two-dimensional convolutions of U-Net with one-dimensional convolutions (cf. Fig. 3.4.2). It learns a mapping function that, when applied to a noisy audio waveform, produces the enhanced waveform which is ideally identical to the clean audio. During training, the mean square error (MSE) between the enhanced output and clean audio is minimised to optimise the model. Besides, the model may be optimised with a training objective inspired by GANs as described in [56]. Similar ideas have been applied in other studies [84, 85, 86, 87, 88, 89], along with the tricks that result in more efficient GAN training [170].

3.4.1.2 Computer Audition Applications & Models

We detail the CA applications under consideration, as well as their corresponding model architectures, which are used as the downstream audio tasks of the audio enhancement model to assess our methods for optimising an AE system towards CA applications.

Automatic Speech Recognition

Automatic Speech Recognition (ASR), a problem at the intersection of computer science and computational linguistics, is the technology that converts spoken language into the corresponding texts by machine. It is a crucial component for the construction of devices that enable human-computer interaction (HCI), such as voice assistants can recognise human voice commands [171]. In the era of deep learning, ASR technology has progressed significantly, and current neural network-based ASR systems are approaching human recognition capabilities, when the input speech captured by a close-talk microphone [172]. A typical ASR system contains both an acoustic model and a language model. The acoustic model learns the speech structure and converts it into probabilities over alphabetic letters. The language model transforms these probabilities into words of coherent language. The acoustic models of cutting-edge ASR algorithms are created using self-supervised learning, a deep learning methodology that aims at discovering general representations from large-scale data without human labelling. It is expected that the learnt representations are effective not only for ASR [173, 174, 175], but also for various downstream tasks [47, 176].

The ASR model implemented does not dependent on SSL training, allowing the use of our joint optimisation approaches detailed in Section 5. The architecture of the model (cf.. Figure 3.4.3) is similar to that of Deep Speech 2 [177], which is a Convolutional Recurrent Neural Network (CRNN), i.e., an RNN constructed on top of a CNN. Specifically, the CNN module is comprised of three residual blocks, each of which contains two convolutional blocks comprising the sequence of layer normalisation, Gaussian Error Linear Unit (GeLU) as activation function, Dropout, and convolution modules. Unlike the CNN used in Deep Speech 2, our additional usage of skip-connections can provide a more stable convergence [23, 178]. The CNN output is then fed into bidirectional Gate Recurrent Units (GRUs) capable of capturing the temporal dynamics of data to generate speech representations . The representations are finally transferred to the character indices, and optimised using connectionist temporal classification (CTC) loss [179]. During inference, to improve the quality of the recognised text, the output of the acoustic model is decoded using beam search with a 3-gram ARPA language model[7].

In practical applications, the ASR performance may be hindered by noise environments, particularly when the noise is intense and obscures the intended speech. Under conditions of low SNR, machine ASR performance remains inferior to that of humans [180]. A considerable number of studies have been conducted to increase the robustness of ASR models against noise, including the use of data augmentation techniques that introduce data deformations in small partial loss of temporal or frequency information [121], or incorporate additive noise to speech input [181, 182]. Alternately, a teacher network, which is developed

[7] The language model is available at https://www.openslr.org/11/.

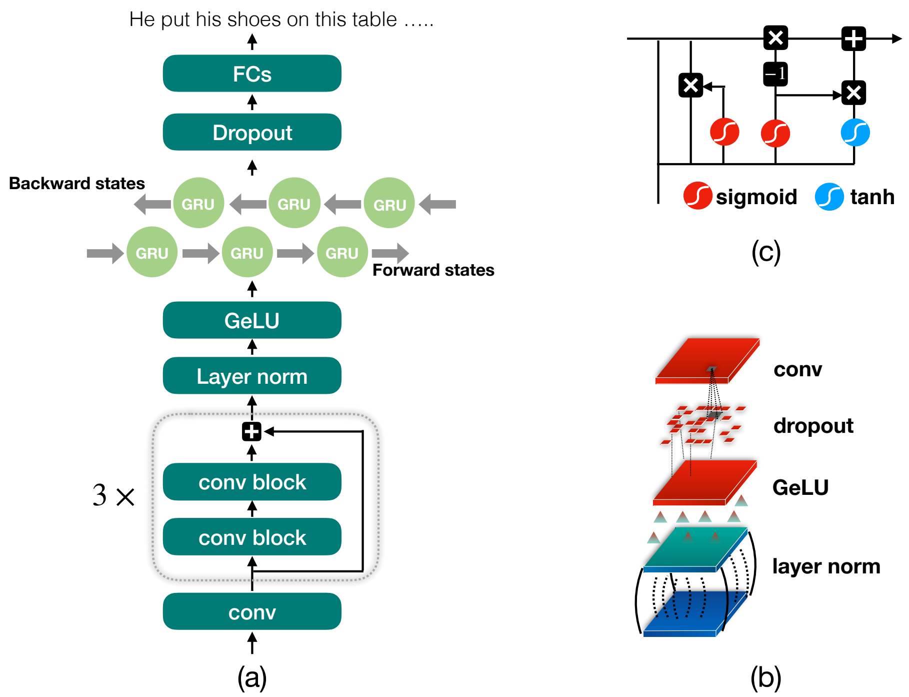


Fig. 3.4.3: Acoustic model of the used ASR system. (a) The architecture consists of residual networks and BiGRU in sequence, fully-connected layers are used to predict the characters in alphabet. The acoustic model is optimised using CTC Loss. (b) The structure of convolutional (conv) block used in the acoustic model. (c) The signal flow of Gated Recurrent Unit (GRU).

based on clean audio, can be used to train a student network to progressively adjust to noisy input [183, 184].

Speech Command Recognition

Keyword spotting systems are meant to identify the presence of speech commands in audio, and are thus commonly deployed on edge devices as the voice interface to activate further functions, which may cost much energy and time, and should therefore be implemented on cloud-severs. Since SCR modes [185] are implemented on hardware of edge devices, they should have a modest model size for memory efficiency, and minimal computation requirements for improved battery life. Unlike for ASR, a SCR model process limited numbers of spoken words as opposed to sentences that may include several words, hence reducing the recognition task to a simple classification problem. As a consequence, it requires no specific loss function for model training and an additional language model for decoding.

The SCR model implemented in this study is a modified M5 CNN [186] that is able to predict the 35 types of speech commands mentioned in [185]. The model is a 4-layer, one-dimensional CNN that directly processes time-domain audio waveforms. Each CNN layer comprises of convolution, batch normalisation, ReLU, and max-pooling. The averaged CNN output is linearly projected onto the speech command classes. As the model processes raw waveform directly, its parameters are meticulously chosen in order to ensure

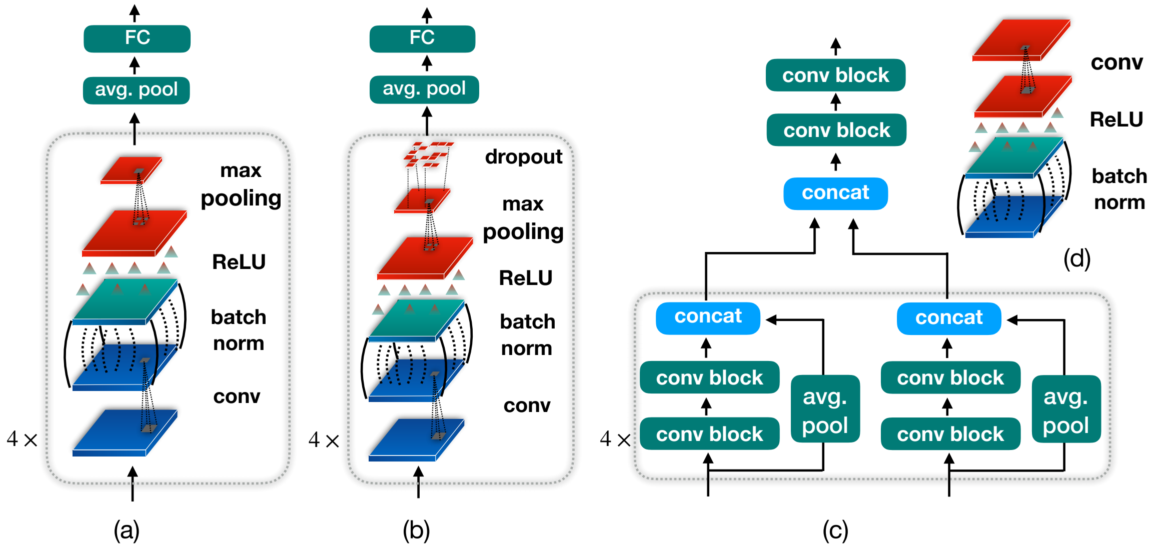


Fig. 3.4.4: (a) Speech command recognition model. (b) Speech emotion recognition model. (c) Acoustic scene classification model, in which the convolutional (conv) block is shown in (d).

that the audio processing is performed with a suitable CNN respective field.

Speech Emotion Recognition

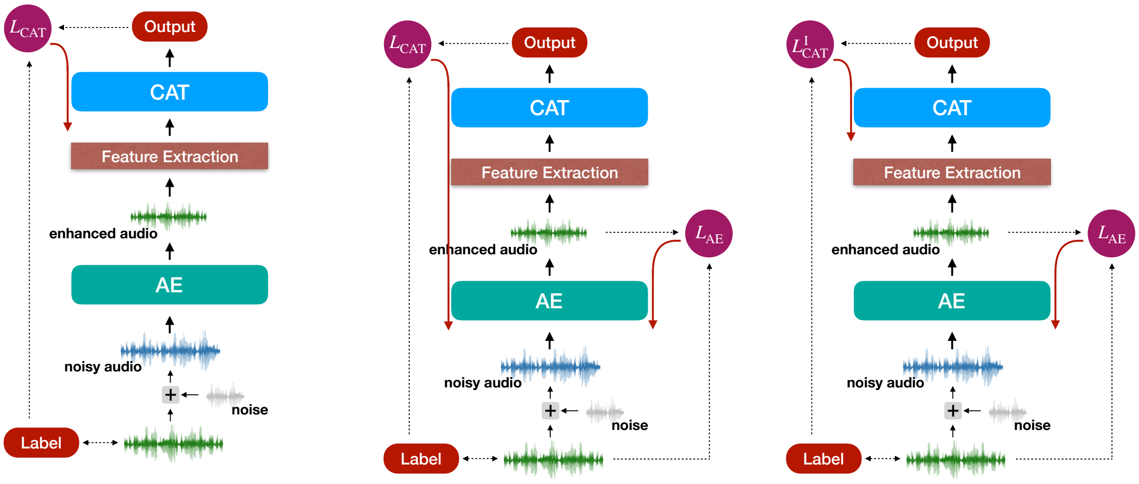
SER is an essential technology for the effective development of Human-Computer Interaction (HCI) applications [187]. Typically, SER algorithm development is formulated as a classification (of 'basic' emotions) or a regression problem (of emotional dimensions) [187]. The study field has achieved great progress as a result of the development of deep learning algorithms; yet, model robustness remains a serious barrier. It has been shown that SER models are particularly susceptible to external noise, since it is beyond the control of the application developer; and thus audio enhancement is required to overcome this issue.

Specifically, we use a 4-layer two-dimensional CNN, where each layer consists of a sequence of convolution, batch normalisation, ReLU activation, max-pooling, and dropout. Its input is a Mel spectrogram computed with 32 Mel-scale filters, a window length of 20 ms, and a step size of 10 ms. The CNN output is projected onto emotion classes using a dense layer.

Acoustic Scene Classification

As the last application for evaluating our joint optimisation methods, we again consider the application of voice suppression for ASC models to suppress human voice in scene recordings. For the ASC model, we implement Dual-ResNet [188] which was awarded as the best reproducible system for the first task of the 2020 DCASE challenge [189].

The model has two separate paths for independent analysis of the low- and high-frequency bands. The outputs of these two paths are concatenated using late fusion before going through two 1 × 1 convolutional layers to reduce the channels to the number of desired classes. The low- and high-frequency paths share the same architecture, a residual network consists of eight convolutional blocks, each of which is a sequence of batch normalisation,



Cold cascade + data augmentation

Multi-task learning

Iterative optimisation

Fig. 3.4.5: Diagrams showing the methodologies used. The red arrows demonstrate the back-propagation through the network modules with respect to the losses L of the AE and the CAT.

ReLU activation, and a convolution processing (cf.. Figure 3.4.4(d)).

3.4.2 Systematic Combination & Training Paradigms

To combine the audio enhancement system and a CA model into a sequence, as well as to enable end-to-end learning for training the entire system, we make a minor but crucial change to the U-Net specifications by setting the max-pooling along the time-axis to 1, while leaving the pooling along frequency-axis unchanged. By doing this, the audio enhancement model can handle audio signals of varying lengths for applications like ASR. Consequently, the AE system is flexible in cascading with any subsequent audio models. Intermediate features, such as Mel-Frequency Cepstral Coefficients (MFCCs) as ASR model input, are extracted from the enhanced waveform or the AE outcome.

We investigate two joint optimisation approaches for training an AE and CA models, namely multi-task learning and iterative optimisation, both of which seek to improve the mutual promotion between the AE and CA models.

3.4.2.1 Multi-task Learning

The first approach utilises a multi-task learning framework that combines the losses of the audio enhancement system and a CA model. The total loss is expressed as

$$L = L_{\mathrm{AE}} + L_{\mathrm{CA}}. \tag{19}$$

Minimising the total loss entails optimising both models simultaneously, since the losses from them are equally weighted. Unlike the typical MTL problem, the alignment and connection of the two models are distinct. The AE loss is derived from an intermediate system layer, although the two models are each viewed as a single entity. Consequently, minimising the AE loss has no impact on the parameters of the CA model, but the CA loss back-propagates through the AE model. Therefore, even while the AE and CA losses function as mutual regularisation terms, they also create a bias towards the updating of the AE parameters. Similar effects have been reported in supervised auto-encoder research [190].

3.4.2.2 Iterative Optimisation

Iterative optimisation, as its name suggests, iteratively trains the AE and CA models. The technique is primarily motivated by a joint view of the two models. First, the CA model should continuously be tuned to the AE model's output, which may contain residual noise, introduced speech distortions, and artefacts, amongst others. Second, the performance of the CA models can be used to enhance the training process of the AE model, allowing the optimisation to concentrate on samples that pose particular challenges to the CA tasks. By doing so, we aim for the optimum performance of the entire neural system, which includes the front-end audio enhancement and the subsequent CA applications.

To implement the iterative optimisation, given a batch of samples $x = [x_1, x_2, ..., x_i, ..., x_N]$, we weigh the AE loss by the normalised CA loss:

$$L^{\mathrm{I}}_{AE}(x, \hat{x}) = \frac{1}{N} \sum_{i=1}^{N} w_i L_{AE}(x_i, \hat{x}_i), \tag{20}$$

where

$$w_i = L_{CA}(t_i, \hat{t}_i) \tag{21}$$

indicates the importance weights, which sum up to 1. These sample-level weights aid in training the AE model to be biased towards relatively more difficult data, for example, those contaminated by more intense noise.

The CA model should be trained using data from the AE system as opposed to the clean signal, to prevent the performance gap between the AE and CA models induced by a cold cascade.

As long as the AE model is optimised, a more robust CA model must be adapted to the enhanced audio, and a more robust CA model can further assist the AE model's optimisation by updating the sample difficulties. Therefore, we alternate between the optimisation of the CA and AE models, i.e., training the CA model based on the AE output while freezing the parameters of the AE system, and training the AE system with the indications from the CA outcomes. The execution of both optimisation steps iteratively can eventually approach to an optimum solution.

3.4.2.3 Comparison Methods

To assess the efficacy of our proposed joint optimisation methods, we compare them with the methods outlined below over different levels of noise intensity:

- **Baseline**: CA models only and they are not trained on noisy data. Since these models are not optimised for noise robustness, we anticipate a considerable performance decrease when confronted with noisy data.

- **Data Augmentation (DA)**: CA models only, and they are trained on synthesised noisy data. We intentionally introduce noise into the clean audio recordings with different SNR ratios. Considering that data augmentation is a common machine learning technique for improving robustness, the CA models should perform better on the noisy test data.

- **Cold Cascade**: The CA models exploit a front-end AE component. However, the AE and CA models are independently optimised. To do this, the AE model is trained to reach a satisfactory enhancement performance, and then the CA model is trained with clean data and stacked on top of the AE model.
- **Cold Cascade + Data Augmentation (DA)**: Both the AE and CA models are trained with synthesised data. The AE model is trained to achieve a satisfactory performance. Subsequently, the CA model is optimised based on the enhanced output from the AE model. Due to the models' exposure to noisy data and the incorporation of an AE component, this method should exhibit promising noise robustness.

To evaluate ASR performance using the CHiME-4 challenge pipelines[8], a classic GMM-HMM model and a DNN-HMM-sMBR model employing an RNN language model for rescoring, we distinguish two comparison methods of the cold cascade style using distinct training data, both methods only train the SE component:

- **Cold Cascade 1**: The SE model is trained using the data synthesised from the LibriSpeech and AudioSet corpora.
- **Cold Cascade 2**: The SE is trained using CHiME-4 training set.

3.4.3 Experiments & Evaluation

We begin by comparing different U-shaped AE neural networks presented in Section 3.4.1. The model with the best performance, which produces the least audio distortions in enhanced audio, is chosen as the front-end enhancement module of our CA tasks to evaluate the effectiveness of our proposed joint optimisation approaches. Following is a description of the data used in this study and an introduction to the data processing. The training parameters for these models are detailed in Section 3.4.3.2.

3.4.3.1 Data Description & Processing

Audio Enhancement
The performance assessment of the U-shaped AE models is based on Edinburgh noisy speech database [152], a corpus including both clean and noisy speech in parallel. The speech data was collected from 56 speakers, 28 males and 28 females, from distinct accent areas (Scotland and United States). Each speaker contributed around 400 utterances. The noisy data was created using two artificially generated noise and eight real noise recordings from the Demand database [158]. The chosen noises represent a variety of real-world situations, including domestic noise, office noise, public space noise, transportation noise and street noise. SNR values of 0, 5, 10, 15 dBwere considered to synthesise the training data. The test set employs slightly higher SNR values of 2.5, 7.5, 12.5, or 17.5 dB, to synthesise speech recordings from a male and a female speaker with five kinds of noises from the Demand

[8] CHiME-4 ASR pipelines are available at `http://spandh.dcs.shef.ac.uk/chime_challenge/CHiME4/software.html`

database. All data is sampled at a frequency of 48 kHz, while the audio recordings are downsampled to 16 kHz for computing efficiency.

Automatic Speech Recognition

The performance of ASR is first evaluated using synthesised audio data based on Librispeech and AudioSet, with SNR values in the range of $0, 5, 10, 15, 20, 25$dB. Additionally, we base our study on a standard benchmark released in the fourth CHiME challenge, which establishes a target for distant-talking automatic speech recognition using the Wall Street Journal (WSJ0) corpus. The simulated training set is generated by artificially mixing clean speech with noisy backgrounds, resulting in 35 690 utterances from 83 speakers in four different noisy circumstances. The test set comprises simulated recordings and utterances recorded by 4 other speakers in actual noisy environments.

Further, Mel-Frequency Cepstral Coefficients (MFCCs) are taken from the audio as input features to the ASR model, the number of Mel-band filters is set to 40. The MFCCs are generated by calculating the Short-Term Fourier Transform (STFT) of an audio sample, and then mapping its powers onto the Mel scale using triangular overlapping windows. Applying discrete cosine transform on the logarithmic values of the powers at each of the Mel bands yields the in final MFCCs. Note that in our two joint optimisation methods, the MFCCs are retrieved from the SE output.

Additionally, we develop the models using the more difficult and realistic data from the CHiME-4 challenge, and then evaluate their performance with the given ASR pipelines.

Speech Command Recognition

The Speech Commands dataset contains 105,829 one-second audio clips of 35 words, including the numerals zero through nine, fourteen words used as commands in IoT and robotics applications, and other spoken words that cover a variety of phonemes. It also contains recordings of only background noise or non-command audio, with the expectation that the tested keyword spotting systems are able to distinguish the audio of commands from the audio of none with the lowest possible false positives.

To evaluate the robustness of a SCR model, noise recordings from AudioSet are selected, trimmed to a length of one second, and then added to speech commands with SNR values within the pale of $0, 5, 10, 15, 20, 25$dB. This results in a large-scale audio collection of speech commands in noisy conditions. Since our implemented model for SCR directly handles the raw audio, no further data processing is needed.

Speech Emotion Recognition

The DEMoS database encompasses 9,365 emotional and 322 neutral audio samples collected from 68 native speakers (23 females and 45 males; mean age 23.7 years, std 4.3 years). Seven emotions, namely anger, sadness, happiness, fear, surprise, fear, and guilt, were evoked by listening to music, watching pictures or movies, pronouncing or reading emotionally sentences and recalling personal memories. All recordings are sampled at 44.1 kHz.

We used all of the emotional samples from DEMoS, and the data partitioning for training, development and testing is the same as in [191], which ensures a gender- and class-balanced speaker-independent split. To maintain consistent with other CA applications, DEMoS samples are downsampled to 16 kHz, which according to [191], will not lead to much information loss . Similar to the study of ASR, we simulate background noise along by adding environmental recordings from AudioSet to speech utterances.

We extract Mel spectrogram from the audio samples using STFT with a window length of 20 ms and a step size of 10 ms, as well as 32 Mel-scale filters.

Acoustic Scene Classification

To evaluate our approach on the ASC task, we use the DCASE 2021 Challenge dataset [192]. This is accomplished by incorporating speech samples from LibriSpeech into the soundscape recordings from the DCASE 2021 challenge in order to create the noisy scene audio. The SNR range is expanded to $-25, -20, -15, -10, -5, 0, 5, 10$ dBto account for a wider range of real-world circumstances. A lower SNR implies that the scene dominates the soundscape, while a higher SNR indicates that speech interference is more prominent. The ASC model input is a log Mel spectrogram derived from the audio using STFT with a window length of 64 ms and a hop size of 16 ms. Meanwhile, the number of Mel bands is set to 128.

3.4.3.2 Training Settings

From our empirical experience, a batch size of 16 is optimal for training a U-Net for audio enhancement. Thus, the batch size remains constant throughout the experiments presented in this section. CTC loss is utilised to optimise the ASR model, while cross-entropy loss between the predicted and ground-truth labels is used for the other three CA classification tasks. These models are all optimised using an Adam optimiser. Weight decay is additionally applied to the training of the SCR and ASC models for the L2 regularisation effect. The AE, ASR and ASC models are optimised using a learning rate of 0.0001, and the SER model is trained with a learning rate of 0.001 . For SCR optimisation, the initial learning rate reduces from 0.01 to 0.001 after 20 epochs. Additionally, for CA applications like ASR and SER that handle audio of varying lengths as model input, the recordings are padded to the length of the longest sample within a batch for training.

3.4.3.3 Choice of U-shaped AE Models

Using the Edinburgh noisy speech database, we will first determine the most appropriate AE model among the U-shaped neural networks, i. e., the original U-Net, complex U-Net and wave U-Net. The selection criterion is associated with audio distortions in the AE model's output, which can be represented by the following evaluation metrics.

Evaluation Metrics

We consider Cepstral Distortion (CD), SDR, STOI and LSD as performance metrics. These evaluation metrics were chosen to complement [38] as they better reflect improvements in machine understanding and are thus deemed suitable for assessing AE systems tailored for further audio applications.

Results Analysis

The standard U-Net, whose specifications can be found in the appendix, surpasses the other two alternative architectures, i. e., Complex U-Net and Wave U-Net, in terms of all the metrics tested, as shown in Table 3.4.1. In particular, an SDR of 18.16, which is 8.35 higher than the original noisy audio, indicates a substantial increase in audio quality w. r. t. signal distortions. Moreover, according to the STOI results, the U-Net model seems to have no detrimental effect on speech intelligibility, but the other two AE models do. Therefore, the AE U-Net will be used for further testing of our proposed joint optimisation approaches.

Table 3.4.1: Testing results of different U-shaped neural networks for speech enhancement using Edinburgh noisy speech database.

Methods	CD	SDR	STOI	LSD
original audio	7.03	9.81	**0.93**	6.29
U-Net	**6.90**	**18.16**	**0.93**	**5.76**
Complex U-Net	6.91	17.89	0.89	5.89
Wave U-Net	7.16	13.21	0.89	5.97

Table 3.4.2: Testing results, WER [%], using Librispeech and the AudioSet corpus. DA stands for the method using only data augmentation. MTL represents the proposed multi-task learning solution.

Methods	Inf	25dB	20dB	15dB	10dB	5dB	0dB	average
original ASR	7.84	10.74	13.53	19.87	31.97	49.72	68.46	32.38
DA	-	9.58	10.18	11.17	14.50	21.05	35.46	16.99
Cold Cascade	-	9.53	10.84	13.31	18.16	28.07	43.78	20.62
Cold Cascade + DA	–	8.15	8.76	10.03	13.30	20.89	34.67	15.97
MTL	-	**8.03**	**8.69**	**9.91**	12.93	19.45	32.64	15.27
iterative optimisation	-	8.35	8.79	10.00	**12.71**	**19.27**	**31.93**	**15.18**

3.4.3.4 Application I: Automatic Speech Recognition

We evaluate the proposed MLT and iterative optimisation methods using ASR as the first application of interest. The testing is undertaken first with the artificially generated noisy speech data using Librispeech and AudioSet at different SNR levels, and then using CHiME-4 benchmark data.

Evaluation Metrics

Character Error Rate (CER) and Word Error Rate (WER) are two standard performance assessment measures for ASR systems. The first one estimates the proportion of alphabetic letters in an utterance that are erroneously classified, whereas the second one calculates the same percentage with respect to the words in speech. We choose WER as our performance metric in order to make a fair comparison to prior work.

Results Analysis

Testing on Synthesised Noisy Speech

Training and testing the implemented ASR model using clean speech recordings from LibriSpeech yields to a WER of 7.84 %, approaching the performance reported in the previous work [177] and thus adequate for evaluating our suggested joint optimisation approaches, albeit being lower than some current state-of-the-art methods [173, 175]. Adding noise to

Table 3.4.3: Testing results, WER [%], using CHiME-4 challenge set. DA stands for the method using only data augmentation. MTL represents the proposed multi-task learning solution.

	GMM-HMM		**DNN-HMM**	
Methods	**simu**	**real**	**simu**	**real**
original ASR	24.46	22.19	12.96	11.56
Cold Cascade 1	18.48	18.06	12.54	11.14
Cold Cascade 2	16.06	14.59	11.15	9.50
MTL	15.04	12.76	9.88	8.73
iterative optimisation	**14.08**	**12.53**	**9.45**	**8.12**

the clean test audio recordings reduces the accuracy of speech recognition (cf. Table 3.4.2). As the SNR declines to 5dB, almost half of the words in each utterances are misidentified, resulting in a WER of 49.72 %. The ASR performance degrades at lower SNR values like 0dB.

Applying data augmentation to the training of the ASR model improves the WERs under all SNR conditions. It outperforms the use of an independently trained SE model at front-end of the ASR model trained on clean speech, especially at relatively lower SNRs. Applying data augmentation to the training of the ASR model improves the WERs under all SNR conditions. It outperforms the use of an independently trained SE model at front-end of the ASR model trained on clean speech, especially at relatively lower SNRs. For example, data augmentation reduces WER by 28.67 % at the SNR of 5dB, and 33.00 % at the SNR of 0dB. Additionally, speech enhancement can boost the WERs further, yielding in an average WER of 15.97 % for all SNR levels evaluated.

Our two suggested optimisation methods can further reduce WERs for all SNR scenarios. With a WER decrease of 2.03 % for 0dB SNR and 2.74 % for 5dB SNR, the MTL method provides considerable performance gains for low SNR levels. The iterative optimisation strategy improves the ASR performance in low SNR situations while maintaining a similar recognition performance at other SNR levels. It reaches an average WER of 15.18 % across all SNR levels considered. In general, at low SNR levels, the two presented joint optimisation techniques that strengthen the interaction between the SE and ASR models outperform those that rely on separate training.

Testing on CHiME-4 Challenge Data

We also apply both joint optimisation approaches to the training of the SE and ASR models using CHiME-4 training dataset, however we only assess the performance of the SE system in combination with the CHiME-4 ASR pipelines.

Using the approach of Cold Cascade 1, the GMM-HMM ASR model can obtain WER reductions of 5.98 % and 4.13 % for the simulated and real recordings, respectively (cf.. Table 3.4.3). The DNN-HMM-sMBR model performs slightly better than the other model on both the simulated and real data, presenting a minor decrease in WER when the front-end SE model is exploited. Adapting the SE model to the CHiME-4 training data set

Table 3.4.4: SCR testing results, (Acc)uray[%], using Speech Commands data set and the AudioSet corpus. DA stands for the method using only data augmentation. MTL represents the proposed multi-task learning solution.

Methods	Inf	25dB	20dB	15dB	10dB	5dB	0dB	average
original SCR	85.07	83.37	81.35	76.87	67.57	51.52	33.12	65.63
DA	-	82.69	82.07	80.09	77.53	71.66	58.26	75.38
Cold Cascade	-	84.34	83.38	80.85	75.54	66.06	51.92	73.68
Cold Cascade + DA	–	82.65	82.31	81.64	79.31	74.22	64.93	77.51
MTL	-	**85.53**	**84.21**	82.12	80.04	76.54	67.18	79.27
iterative optimisation	-	85.35	83.93	**82.37**	**81.56**	**77.41**	**69.18**	**79.97**

considerably enhances the speech recognition performance of both ASR pipelines, and both simulated and real-world recordings.

In addition, the joint optimisation approaches result in additional WER improvements (cf. Table 3.4.3). Using iterative optimisation, a more effective SE model is connected to the CHiME-4 ASR pipelines, resulting in a WER of 9.45 % on the simulated recordings and 8.12 % on the real recordings for the DNN-HMM-sMBR-based ASR models with language models for rescoring. This improvement can be attributed to the specialised training of the SE model towards the samples that are more critical to the ASR performance. Despite the fact that ASR systems may encounter unique challenges in speech recognition, their performance degrades for contaminated audio samples with similar causes, such as the same type of speech disturbance.

3.4.3.5 Application II: Speech Command Recognition

Evaluation Metrics

Although the amount of utterances for each speech command recorded in [185] are not absolutely equal, the distribution of classes in its standard test partition[9] are sufficiently balanced, allowing the use of classification accuracy as the assessment measure for the task at hand.

Results Analysis

The classification accuracy of the SCR model trained on the original data of Speech Commands reaches 85.07 % (cf.. Table 3.4.4). As the noise intensity increases, the SNR decreases, and the classification performance diminishes. When the SNR falls to 5dB, more than fifty percent of the spoken keywords are misclassified. The average accuracy across all examined SNRs rises from 65.63 % to 75.38 %, when the model is trained with the contaminated data. This pertains particularly to the situations with a low SNR. The additional use of speech enhancement at its front-end leads to a 2.13 % improvement in average accuracy.

Our two joint optimisation approaches attain an average classification accuracy of 79.27 % and 79.97 %, outperforming the baseline methods. In scenarios with a relatively lower SNR, the iterative optimisation strategy performs better than the MTL method. Moreover, when

[9] `http://download.tensorflow.org/data/speech_commands_test_set_v0.02.tar.gz`

Table 3.4.5: SER testing results, Unweighted Average Recalls (UAR)[%], using DEMOS and the AudioSet corpus. DA stands for the method using only data augmentation. MTL represents the proposed multi-task learning solution.

Methods	Inf	25dB	20dB	15dB	10dB	5dB	0dB	average
original SER	81.32	81.18	79.95	78.98	73.70	59.82	40.32	68.99
DA	-	79.53	79.46	79.05	78.30	75.69	68.06	76.68
Cold Cascade	-	80.45	79.59	79.45	77.86	69.93	54.34	73.60
Cold Cascade + DA	–	77.59	77.27	77.07	77.54	74.52	69.02	75.49
MTL	-	81.30	80.67	80.31	79.93	77.29	75.44	79.16
iterative optimisation	-	**81.31**	**80.76**	**80.35**	**79.95**	**78.09**	**76.91**	**79.56**

adding noise to the spoken commands at the SNR level of 25dB, both of our methods manage to achieve higher classification results than the models trained with the original clean data, which is likely owing to the effect of data augmentation using additive noise.

3.4.3.6 Application III: Speech Emotion Recognition

Evaluation Metrics
Considering the imbalanced class distribution in the test set, we assess the trained models using Unweighted Average Recall (UAR), i. e., the unweighted average of the class-specific recalls.

Results Analysis
When training the SER model using the clean Italian speech samples, the UAR on the clean test set approaches 81.32 %. With an average UAR of 68.99 %, the model presents a degree of robustness to additive noise. However, when the SNR is below 15dB, the SER classification accuracy reduces considerably (cf.. Table 3.4.5).

Data augmentation, which involves noise into the training audio, boosts the mode robustness to strong noise, for example in cases when the SNR is below 10dB. However, introducing a separately trained SE model to the SER task can hardly improve or even degrade the classification results.

Both of our joint optimisation approaches indicate improved SER performance, with UARs of 79.16 % and 79.56 %, respectively, surpassing the best-performing baseline model which yields a UAR of 76.68 %. This improvement is in part the consequence of closing the language gap. The original SE model is trained on English data; thus it needs to be optimised for the Italian-SER model for better performance.

3.4.3.7 Application IV: Acoustic scene classification

We train a U-Net as a voice suppression system capable of enhancing the speech robustness of an ASC model.

Results Analysis
According to Table 3.4.6, training the ASC model using the original recordings of acous-

tic scene yields a testing accuracy of 77.81 % for classification. Similar to the previous speech tasks, the classification performance falls as the intensity of voice interference in the testing scene recordings rises. Consequently, an overall average accuracy of 57.48 % is attained across all SNR cases evaluated. For the SNR values of −25 and −20 dB, applying data augmentation or voice suppression might negatively impact the ASC performance. Nonetheless, the benefit of data augmentation begins at the SNR of −15 dB, yielding in an average accuracy gain of 10.68 %. On the other hand, the effectiveness of voice suppression starts at the SNR of −20 dB. However, using voice suppression at the front-end of the ASC model trained with clean scene audio or contaminated audio cannot result in better average accuracy.

Table 3.4.6: ASC testing results, (Acc)uracy[%], using DCASE2021 and Librispeech corpus. DA stands for the method using only data augmentation. MTL represents the proposed multi-task learning solution.

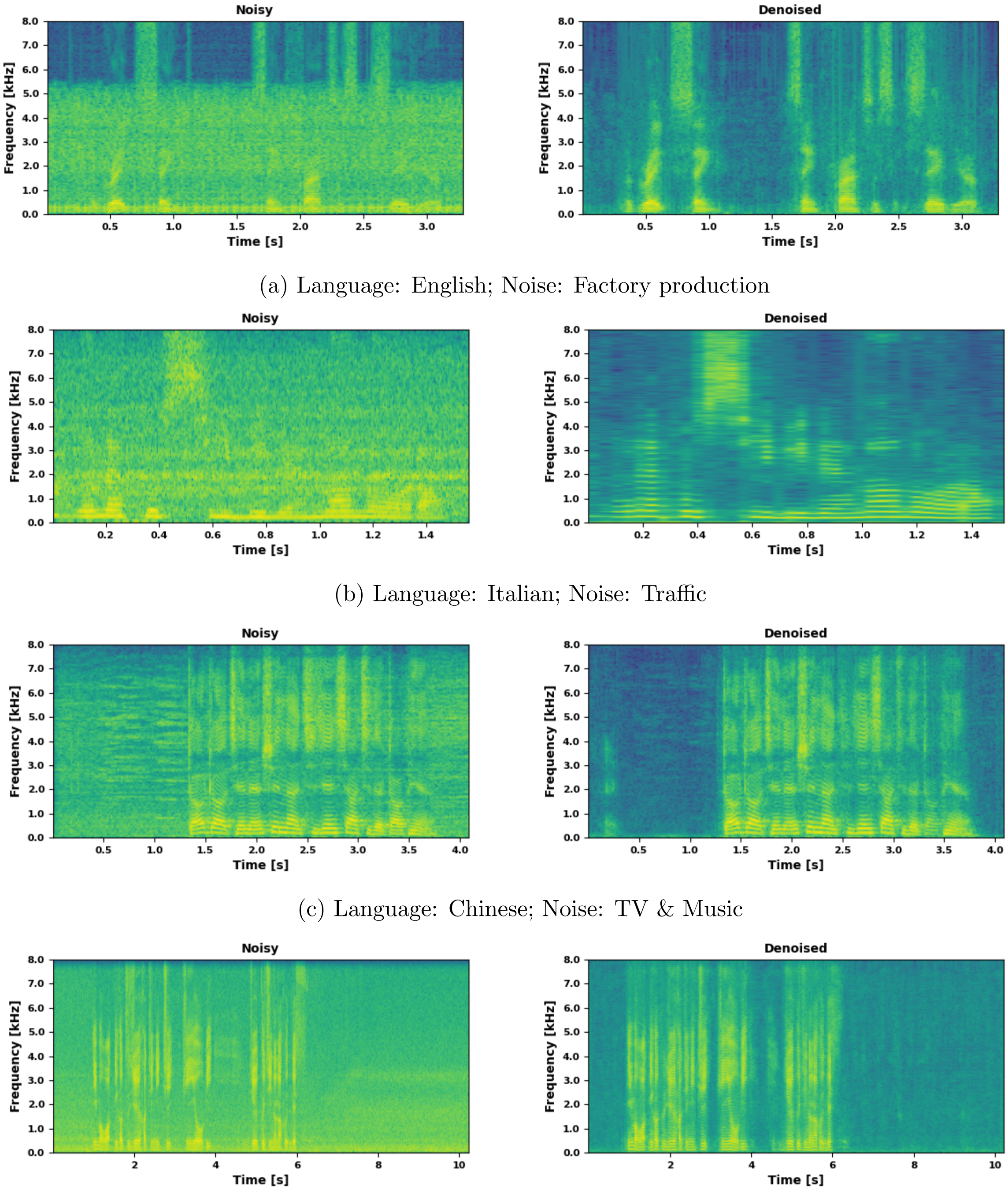
Methods	Inf	−25dB	−20dB	−15dB	−10dB	−5dB	0dB	5dB	10dB	average
original ASC	77.81	75.45	72.92	69.05	65.02	60.14	51.19	39.12	26.91	57.48
DA	-	70.51	71.44	71.50	70.84	69.08	68.86	63.73	59.31	68.16
Cold Cascade	-	73.83	73.11	71.14	67.60	62.58	56.87	50.53	44.53	62.53
Cold Cascade + DA	–	72.92	72.84	72.65	71.71	69.27	63.59	60.16	59.23	67.80
MTL	-	**74.31**	**73.97**	73.01	72.48	71.52	70.10	65.79	61.34	70.32
iterative optimisation	-	74.26	73.50	**73.12**	**72.71**	**72.09**	**71.43**	**66.81**	**63.19**	**70.89**

Training the voice suppression and ASC models using a multi-task learning approach is shown to be superior to iterative optimisation in the case of low SNR, i. e., the presence of a tiny amount of speech in acoustic scene recordings. However, the iterative optimisation outperforms the MTL approach for the worse SNR levels. The average classification accuracy for the two techniques is 70.32 % and 70.89 %, respectively. Both techniques successfully boost the system's robustness across all SNR situations.

3.4.4 Performance visualisation

To provide a more intuitive understanding of the audio enhancement performance of our approaches, for both the tasks of speech enhancement and voice suppression, we visualise the input and output of the trained U-Nets. First, we illustrate the denoising performance on English utterances, the same language used to train the SE model, and then on three additional languages: Italian , Chinese, Japanese, with Chinese and Japanese being actual recordings. Finally, we demonstrate how effective our voice suppression U-Net is removing English speech from environmental backgrounds.

Speech Enhancement
Fig. 3.4.6a displays the performance of our SE U-Net on a testing noisy speech sample. Even when the speech signal is obscured by a severe industrial noise, our model is able to recover voice from the noisy recording while keeping the low-frequency speech components. The similar effect is seen when extracting Italian from the traffic noise (cf. Figure 3.4.6b), and the well-preserved harmonic structure of the speech results in a pleasant audio quality.



(a) Language: English; Noise: Factory production

(b) Language: Italian; Noise: Traffic

(c) Language: Chinese; Noise: TV & Music

(d) Language: Japanese; Noise: Street traffic

Fig. 3.4.6: Spectrograms illustrating the performance of speech enhancement U-Net for English, Italian, Chinese and Japanese in different kinds of real-life environments. The SE model processes the the contaminated audio (noisy) and output the denoised speech.

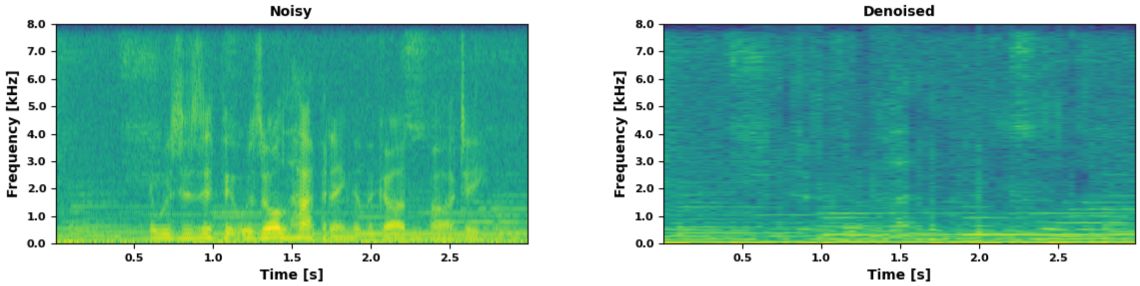


Fig. 3.4.7: Spectrograms illustrating the effect of training an U-Net for voice suppression. The model targets at removing the speech components in the acoustic scene recording to the maximum possible extent.

The SE U-Net trained on noisy English speech data has also been examined for recovering Chinese and Japanese speech in noisy environments. These eastern languages are believed to be fairly dissimilar to English. Nonetheless, the enhancement model retains the ability to improve audio quality by eliminating irrelevant background sounds. Therefore, we conclude that the SE U-Nets trained using our proposed approaches have adequate cross-language processing capabilities, alleviating the difficulty associated with transferring an SE model to a particular language for real-world applications.

Voice Suppression

The purpose of voice suppression is to protect acoustic scene recordings from spoken interference. As seen in Figure 3.4.7, our trained U-Net for voice suppression can successfully suppress the speech components present in the scene audio. Even with some residual speech that is still audible, the enhanced scene audio is of higher audio quality and can improve the following ASC performance.

3.4.5 Section Summary

The emphasis of this section was on single-channel audio enhancement tailored for specific CA applications under low SNR circumstances. Specifically, we considered three CA tasks taking speech as the signal of interest and one with the background soundscapes as the target audio. Instead of a separate training paradigm for audio enhancement and CA models, we presented the multi-task learning and iterative optimisation methods that strengthen the interaction between the two models during training. The testing results reveal considerable improvements determined by the respective assessment criteria for each CA task, particularly for low SNRs. Our suggested solutions allow the customisation of an audio enhancement front-end to the specific CA problem that needs denoising, resulting in substantial improvements over generic enhancement models trained with out-of-domain data. Inspired by the gains, additional effort should be put on the coupling between the two models, and using recent discoveries in self-supervised learning that have been shown to improve audio enhancement.

3.5 Chapter Summary

In this chapter, we presented a deep learning framework that integrates multiple audio enhancement functionalities, including audio denoising, source separation, and selective noise suppression. The framework utilises additional auxiliary networks to encode extra audio samples that indicate the components to be preserved and to be discarded. We also explored the multi-task learning framework aimed to use audio enhancement to improve the succeeding audio applications, i. e., ASR, SCR, SER and ASC, in terms of their robustness and reliability in everyday noisy environments. Our proposed iterative training scheme can improve the global performance of the audio applications in real-world situations, especially when the audio recordings contain intense noise or interference. During the study of this joint training method, we optimised an audio enhancement system with respect to target audio application, respectively. A straightforward approach would be to train such an audio enhancement system with multiple subsequent audio applications. The simultaneous optimisation of an audio enhancement system and these applications models has the potential to further improve both the enhancement efficacy and the application performance. Due to the demand for multi-label data, this idea has not been implemented in this study.

CHAPTER 4

Detection of Coronavirus Disease 2019 (COVID-19)

4.1 Introduction

Since the outbreak of *Coronavirus Disease 2019* (COVID-19) at the end of December 2019, the epidemic has spread globally and affected every aspect of human life. As of the time of writing, more than 523 million positive cases of COVID-19 have been confirmed, including 6.27 millions deaths, according to the World Health Organization (WHO) [10]. The most prevalent signs of COVID-19 infection are fever, cough, fatigue, and a loss of taste or smell. Less common symptoms include sore throat, pains and aches, headache, red or irritated eyes, and diarrhoea, etc. The infection can cause more severe results, including immobility, difficulty in breathing and speaking, chest pain, and even death. Patients are suggested to seek emergency medical care in this situation. Initially, elderly COVID-19 patients were more likely to experience a serious or fatal illness [193]. Numerous mutant strains, including Delta and Omicron, have increased transmissibility, decreased neutralisation by antibodies generated from previous infection or vaccination, impaired efficacy of past treatments or vaccinations, and the ability to attack individuals of all ages. Moreover, recent evidence indicates that COVID-19 is likely responsible for the recent surge of the acute hepatitis of unknown origin in young children. [194].

Currently available approaches for diagnosing COVID-19 infection include CT-scan, PCR test, and rapid test, amongst others [195]; however, they require expensive medical equipment or public expenditures. Presently, rapid point-of-care testing can be completed within minutes at medical facilities, allowing medical teams to rapidly determine the cause of symptoms and manage the isolation and treatment. In actuality, however, COVID-19 patients undergo rapid testing only after the onset of symptoms, resulting in delayed treatment and isolation. The urgent need for a cost-efficient solution that can continually monitor a person's health, and provide immediate alerts upon the detection of a potential COVID-19 infection prompts us to utilise AI technology.

More importantly, measures must be taken to prevent the spread of COVID-19 virus, such as maintaining a safe distance from others, wearing a properly fitted mask in public domains, routinely washing hands, and covering mouth and nose when coughing or sneezing, to mention a few. Vaccination is an effective method of preventing infection or mitigating the disease severity. According to a study based on the data collected from Israel, Sweden the United States of America, and the United Kingdom [196], it can lower the likelihood of hospitalisation by over 80 percent. However, the vaccinations have been shown to be less effective against coronavirus variants such as Omicron, and some adverse reactions to the COVID-19 vaccine have been documented. Another policy for combating the spread of COVID-19 is implementing the mobility restrictions. Despite its evident effectiveness,

[10] https://covid19.who.int/

lockdown and social isolation are detrimental to our everyday lives, as they can result in financial challenges for the human community, cause health issues including to mental health damage, and negative emotional responses [197].

Thanks to the unremitting efforts of medical experts, epidemiologists and scientific researchers, we have a more comprehensive and in-depth understanding of the coronavirus and its diffusion. Normalcy is gradually returning to public life alongside the global rise in vaccination rates and more effective implementation of regulations to prevent epidemics. In the fight against COVID-19, various challenges remain to be overcome. Above all, the current procedures for confirming the disease may need patients to visit a clinical facility, and certain methods necessitate the use of specialised clinical equipment. These methods are incapable of reaching an early and automatic diagnosis upon the onset of COVID-19 symptoms, making it impossible to take immediate interventions, such as patients isolation, to limit the extent of the virus' dissemination.

In an effort to combat COVID-19 using deep learning technology [198], we propose deep learning solutions for COVID-19 detection based on several . Our methods for detecting COVID-19 rely on either audio data or heart rate measurements. The possibility of collecting these kinds of data up to 24 hours, for instance via wearable devices such as smartwatches, enables continuous monitoring of an individual's health, and prompts a timely notification of illness detection. To take advantage of this, we construct neural network models that can analyse cough or breath sounds, or combine the two audio kinds, to predict the COVID-19 detection outcome. The other method, which is based on the analysis of heart rate measurements, frames the task as an anomaly detection problem to mitigate the impact of class-imbalance, an issue frequently encountered in the COVID-19 data collection, on training the neural networks.

In the reminder of the chapter, we first provide a summary of the work related to the current COVID-19 detection methods using machine learning techniques. In Section 4.3 and Section 4.5, we detail our presented AI algorithms for COVID-19 detection using cough and breath sounds, and heart rate measurements, respectively. At last, we aim to employ the speech enhancement method introduced in Section 3.4 to improve the performance of a speech based COVID-19 detection model in noisy environments. We detail the test results in Section 4.4, with the hope that it may inspire more research to aid public health and safety in the era of COVID-19.

4.2 Related Work

Current research towards machine learning solutions for COIVD-19 disease screening includes CV [199, 200, 201, 202], CA [203] methods, and biomedical data analysis approaches [204, 205], particularly those derived based on data from wearable devices. By highlighting some of these methods in **??**, **??** and **??**, we aim to provide a concise overview of the present status of this research. In addition, we briefly summarise the work on face mask detection in **??**, focusing on the most popular CV techniques. The speech-based method we propose later can be seen as an alternate solution for completing this task.

CV Solutions to COVID-19 Detection

Based on chest X-ray images or computed tomography (CT) scans, two prominent ideas for using CV approaches to solve COIVD-19 diagnosis [199] are 1. localisation and segmentation of the infected areas [200], and 2. identification of COVID-19 positive cases[201, 202]. For these purposes, numerous classic deep neural network architectures have been investigated [206, 207, 18, 208, 209], including VGG[210, 211, 212], residual network (ResNet) [213, 214], MobileNet [212, 215], Inception nets [214], U-Net[216, 217], visual Transformer (ViT) [218], Capsule Net [219] amongst others. Moreover, transfer learning can be adopted to boost the performance [213, 220, 221, 222]. Within these CNN frameworks, some additional module blocks, such as squeeze-excitation (SE) block [213], have been found effective in improving the classification performance. Similar CV approaches used to ultrasound images have resulted in more cost-effective solutions [223, 224]. More advanced, contrastive learning has already been investigated to detect COVID-19 from CT scans or X-ray images as presented in [225], which exploited a Siamese network with contrastive loss for n-short learning of COVID-19 patients. Similar to this, Chen *et al* [226] suggested momentum contrastive learning for few-shot COVID-19 detection. Hou [227] aims to advance the representation of COVID-19 through a contrastive training training.

A COVID-19 detection system can integrate a hybrid architecture comprising the aforementioned two processes, i.e., a localisation or segmentation module to identify the probable infected region, followed by a COVID-19 classification model [228, 216, 229]. Wang *et al.* [228] introduced a such system to first localise lung anomalies in chest radiographs before classifying them using a pyramid network. Chen *et al.* [216] proposed a similar method for identifying the suspicious COVID-19 pneumonic lesions, in which the segmentation of infection regions is conducted on successive CT scans, and classification is accomplished using a UNet++. The segmentation and classification can also be performed concurrently by using multi-task learning as in [230, 231, 232, 233]. Li *et al.* [234] has confirmed the improvement in generalisability of this method for unseen chest CT and X-ray images.

To access the illness severity, He *et al.* suggested a method analysing CT images using synergistic learning of lung lobe segmentation and hierarchical multi-instance classification approaches [235]. Alongside the development of COVID-19 detection algorithms, the results' interpretability has raised growing interest [222, 236, 237]. Besides, great efforts have been undertaken to differentiate a COVID-19 infection from other kinds of pneumonia with similar symptoms [238, 239].

CA Solutions to COVID-19 Detection

Recent research has investigated alternative data streams that can be easily acquired with a smart device or platform [240], such as audio recordings of coughing and breathing [203] and speech signals [241, 242], that may possibly be used to detect COVID-19. Particularly, Xia *et al.* [243] collected a large-scale crowdsourced audio database titled COVID-19 Sounds encompassing recordings of human speech, cough, breathing. The authors explored methods to identify COVID-19 based on each of these audio types, and exhibited the performance of fusing the audio types to reach a greater detection accuracy. Imran *et al.* [244] built a cough detector and a COVID-19 diagnostic module for an AI-powered smartphone app that identifies COVID-19 from coughing data. The cough detector analyses the Mel-spectrogram of the recorded audio using a CNN, and the diagnostic module integrates three classification models in parallel. The app announces a diagnosis only if all the classification models yield identical findings. To capture the temporal dynamics of cough sounds, it is recommended to utilise an LSTM model supplemented with an attention block [245] or a simple Transformer [246] to improve the processing, which results in improved detection accuracy and reliability. The research of Faezipour and Abuzneid [247] has prompted the analysis of the time and frequency components of breathing sounds for this disease detection. In three distinct works, AUCO ResNet [248], QUCoughScope [249], and [250] have examined the efficacy of using cough and breathing recordings for this detection task.

Transfer learning has also been considered as a potential CA solution for the detection of COVID-19. For example, Laguarta [251] implements a CNN composed of three pre-trained ResNets to identify COVID-19 individuals based on the acoustic biomarker attributes underlying cough sounds. For the same audio type, an ensemble learning approach [252] integrates shallow machining learning, CNN and pre-trained CNNs to analyse six feature representations of a cough sample to test COVID-19 illness. Furthermore, Pal and Sankarasubbu [253] sought to create representative embeddings for interpretable cough symptoms.

COVID-19 Detection using Bio-signals

Wearable technology and smart devices can be utilised to record biomedical signals or health data for COVID-19 detection. [204, 205, 254, 255, 256]. Using data collected from wearable biosensors, Un *et al.* [257] suggested an indicator generated using machine learning technique that measures the general health status of patients with moderate COVID-19. Hirten and colleagues [258] performed an evaluation of heart rate variability (HRV) collected from a wearable device in order to identify and predict COVID-19 and its associated symptoms. In addition to the analysis of resting heart rate, Radin and colleagues [259] considered the duration of sleep for approximately 47 000 individuals to enhance model predictions of influenza rates in five US states. Similarly, Quer *et al.* [260] and Mishra *et al.* [256] have shown the feasibility of utilising heart rate and sleep duration data, as well as activity data which can also be collected from smart wearable devices, to accomplish this detection task. Natarajan and colleagues [261] trained a CNN to predict illness using Fitbit data from 1 181 individuals, and reported an area under the receiver operating characteristics curve (AUC-ROC) of 0.77 ± 0.03. Moreover, Mishra *et al.* showed the possibility to detect COVID-19 prior to the onset of symptoms. The method relies on the continuous acquisition of physiological and activity data to measure vital signs associated with COVID-19 illness [262].

In the ongoing DETECT study[11] [260], researchers are tracking outbreaks of viral infections including COVID-19 based on the resting heart rate collected [263]. In addition to some prior studies [205, 261], similar continuing endeavours include the German project *Corona-Datenspende*[12], which has a cohort of over 500 000 volunteers, and the american TemPredict study [13].

4.3 COVID-19 Detection using Cough and Breath Sounds

Due to the effect of the COVID-19 virus on the respiratory system, coughing and breathing serve as possible diagnostic indicators for the disease. In this section, we first investigate the application of deep learning approaches to this detection problem utilising each of the two audio types individually. To advance the complementarity effect between breathing and coughing signals, we consider several data fusion techniques, such as concatenation, and convolutional fusion, as well as our own deep fusion approach, to merge the information learnt from these two audio types. Unlike the method exploited in section 1 for audio enhancement, we extend deep fusion as a solution for information fusion. In the following, we present the single-type, multi-type fusion and our deep fusion models, with a particular

[11] http://detectstudy.org/ [as of 03 August 2021]

[12] http://corona-datenspende.de/science/en/ [as of 03 August 2021]

[13] http://osher.ucsf.edu/research/current-research-studies/tempredict [as of 03 August 2021]

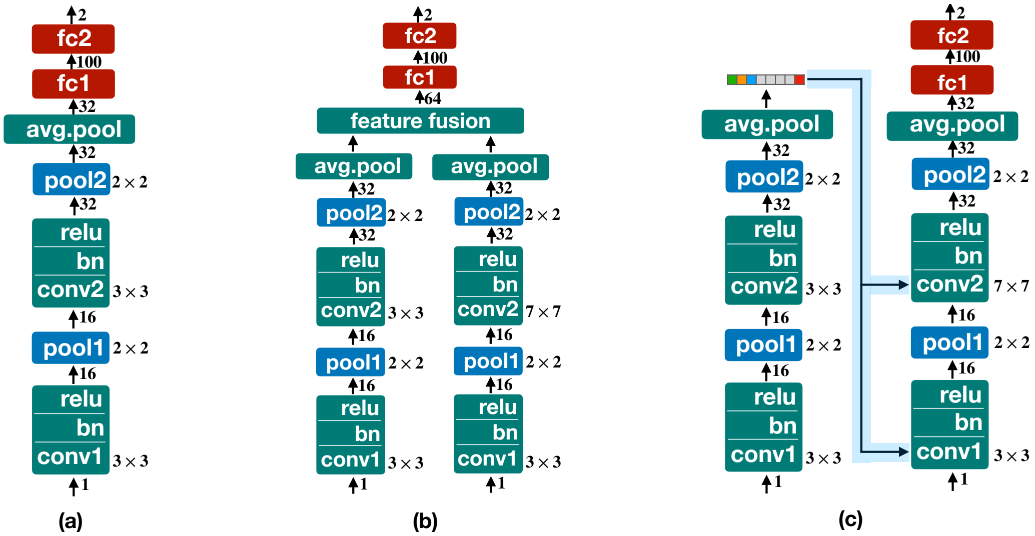


Fig. 4.3.1: Block diagram illustrating **(a)** The single-type model, in which either breathing or coughing segments are used as input. **(b)** The multi-type fusion model implemented, in which both breathing and coughing segments are simultaneously used as input. **(c)** The deep fusion model proposed. The kernel size of each convolutional and max-pooling layer is given next to each block. The channel change is provided next to each transition arrow between adjacent blocks. The feature fusion mechanisms applied to this network are either direct concatenation or 1×1 convolution. It projects the learned embedding features to match the channel-dimension of the convolutional layer, and adds them to the feature map obtained at the output of the convolution.(*Source*: [264])

emphasis on comparing the deep fusion method with the other two conventional data fusion techniques.

4.3.1 Network Architectures

We consistently employ the same CNN backbone to assess different model structures introduced above for the task of COVID-19 detection. The single-type model analyses a single audio type, either a breathing or coughing signal segment. The multi-type model contains two subnetworks, with each subnetwork processing one audio type. The two generated audio representations from the subnetworks are concatenated or processed using an additional convolutional layer to merge the information. Unlike these two late fusion approaches, we suggest the use of the deep fusion approach that allows a more thorough information coupling across all CNN layers involved, hence enhancing the effect of complementarity between the two audio types.

4.3.1.1 Single-type Models

The single-type model (cf. Figure 4.3.1(a)) is constructed as a 2-layers CNN taking the MFCC of an audio segment as input, and the convolutional output is averaged across all locations to squeeze it into an embedding vector. Subsequently, the embedding vector is projected to the predictions, i. e., COVID-19 positive or negative, using two fully-connected layers. Besides, batch normalisation [142] and ReLU activation are used for every convolutional layer in order to obtain a more efficient convergence. Softmax function is used by the output layer to produce the prediction probability of each class. These probability values can thus be seen as the confidence scores used to classify the input audio sample into each possible class.

4.3.1.2 Multi-type Fusion Models

The multi-type model (cf. Figure 4.3.1(b)) should analyse an individual's both breathing and coughing data simultaneously. For achieve this goal, the model is built with two subnetworks sharing the same architecture as the single-type model. One subnetwork processes the breathing segment, and the other processes the coughing segment. Again, average pooling is applied to the outputs of both subnetworks to create the breathing and coughing audio embeddings. To combine the two audio information, we first apply two basic data fusion techniques: concatenation and convolution. The first fusion method concatenates the two embeddings into a single, bigger embedding that preserves the information of both audio types to the greatest extent possible. The alternate way is to stack the embeddings into a two-channel representation, and then compress the channels using a 1×1 convolution. Following data fusion, the resulting embedding is processed via the dense layers for the illness detection.

4.3.1.3 Deep Fusion Model

The proposed deep fusion model (cf. Figure 4.3.1(c)) also contains two subnetworks. Specifically, the audio segment of one audio type is learnt and embedded into a representation vector, and the representation is injected into all the convolutional layers of the other subnetwork. For this, the learnt embedding vector from the first subnetwork is linearly projected to match the channel dimension of each convolutional layer in the second subnetwork. The projected embedding vectors are then added to the convolution activations. In this way, the learning of one audio type takes into consideration the other audio type, producing a deeper information fusion. From a different perspective, the detection is primarily dependent on the learning of one audio type, while an auxiliary network supplies additional audio information by learning from the other audio type.

4.3.2 Experiments & Evaluation

The performance of the three presented CNN models is compared on the basis of two audio levels, namely segment level and sample level. All the presented models are optimised by minimising a cross-entropy loss between the predictions and ground truth using an Adam optimiser with a learning rate of 0.0001. The batch size for training is consistently set at 32.

4.3.2.1 Data Description & Processing

The breathing and coughing audio data used in this experiment were acquired using a web- or Android-based recording platform [265]. For this, each participant was instructed to cough three times and take three to five deep breaths to the app. As the ground truth, the participants were required to report whether or not they tested positive for COVID-19. A portion of the collected data has been made available for research purpose. The original release comprises of 62 COVID-positive patients providing a total of 141 cough and

Table 4.3.1: Summary with the distribution of the data available over the training, validation, and test partitions. In this table, we depict the number of patients populating each partition, the total number of breathing (B) and coughing (C) segments available, and the total number of breathing-coughing segment pairs (B+C) combined. The information from each partition is provided independently for both COVID-19 (Pos) and healthy (Neg) patients. (*Source*: [264])

	Train		**Validation**		**Test**		**Total**	
COVID-19	**Pos**	**Neg**	**Pos**	**Neg**	**Pos**	**Neg**	**Pos**	**Neg**
Patients	22	170	13	13	27	27	62	210
B	464	938	162	52	260	304	886	1294
C	207	410	55	27	109	165	371	602
B+C	1337	2047	370	126	678	695	2385	2868

breathing samples, and 220 non-COVID individuals who submitted 298 audio samples. For the goal of evaluating the models that need both audio types, we excluded the individuals supplying only a single audio type. In addition, we meticulously listened to the audio samples and removed those recordings that are too quiet, with the purpose of improving data quality for training more reliable detection models. This yields a total of 288 audio samples from 210 healthy individuals for our experiments.

These selected audio samples are then partitioned into participant-independent training, validation, and test splits with proportions of 70 %, 10 % and 20 % of all data, while taking into account the balance of COVID positive and negative patients in both the validation and test partitions (cf. Table 4.3.1). The data balance maintained for development and evaluation ensures a fair evaluation of the models under similar conditions. The audio samples are then split into participant-independent training, validation, and test partitions with the proportion 70 %, 10 % and 20 % of all data, considering the balance of the COVID positive and negative patients in both the validation and test partitions (cf. Table 4.3.1). The balance maintained in validation and test guarantee a fair evaluation of the models under similar conditions [14].

Unlike previous work that used the entire audio recordings of variant lengths, we segment each breathing and coughing sample into frames of a certain duration, i. e., 2 seconds for coughing samples and 2.5 seconds for breathing samples, in order to provide the models with a common input format. The duration of breathing segments can guarantee at least one full inspiration or expiration included. Nonetheless, some coughing segments may include a deep inhalation during the preparation phase prior to the actual coughing. The truncation is preformed without using overlap between successive segments, and segments that are insufficiently lengthy are discarded. For examining the effectiveness of fusing coughing and breathing information, a cough and breathing trunk belonging to an individual are combined as a cough-breath pair, resulting in more training and evaluating trunk pairs. The distribution of the cough, breathing samples, and cough-breath pairs in each split is detailed in Table 4.3.1. As input to our models, we take first 40 *Mel-Frequency Cepstral Coefficients* (MFCCs) [266] derived from the audio's short-term power spectrum.

[14] The samples and data partitioning needed to reproduce our experimental findings are publicly available at https://github.com/EIHW/MultiTypeFusionForCOVID19Detection.

Table 4.3.2: Performance comparison [$\mu \pm$ CI in %] of the models trained on the test set, when considering each audio segment as an individual sample. B and C correspond to the single-type models trained using breathing and coughing samples, respectively. B+C corresponds to the multi-type models. The performance of the multi-type models is differentiated in terms of the fusion method they use. B2C/C2B indicate the models that inject breathing representations into the convolutional layers responsible for learning the coughing representations, and vice versa. (*Source: [264]*)

Model	ACC	UAR	UAP	UF1
B	71.1 ± 2.2	70.0 ± 3.5	74.0 ± 2.9	69.2 ± 2.7
C	74.1 ± 2.3	72.7 ± 3.2	73.6 ± 3.3	72.9 ± 2.6
B+C – Concat	74.2 ± 2.1	75.0 ± 3.3	74.7 ± 3.0	74.2 ± 2.5
B+C – Conv	74.4 ± 2.2	75.1 ± 3.2	75.6 ± 3.0	75.1 ± 2.4
B+C – B2C	$\mathbf{83.1 \pm 2.3}$	$\mathbf{83.7 \pm 3.4}$	$\mathbf{83.9 \pm 3.4}$	$\mathbf{83.8 \pm 2.7}$
B+C – C2B	81.4 ± 3.5	81.9 ± 3.3	83.1 ± 2.8	81.9 ± 2.3

4.3.2.2 Evaluation Methods

We report the experimental results separately in terms of the audio segment and full sample levels. The first one considers each audio segment to be an independent sample, whereas the second views each original audio sample as a whole. For sample-level testing, all possible segments extracted from an audio sample are processed separately by the model, and the final prediction for the sample is determined by the majority vote of the individual predictions. As performance measures for assessing our models, we choose *Accuracy* (ACC), *Unweighted Average Recall* (UAR), *Unweighted Average Precision* (UAP), and *Unweighted F1* (UF1) score, allowing for a fair comparison with other similar studies that encounter the glaring class imbalance. Along with the experimental results, we report the 95% *Confidence Interval* (CI) for each metric evaluated by computing 100x bootstraping for testing purpose (random selection with replacement).

It would be unfair to compare the performance of our models to the baseline given in [265] due to the differences in data partitioning and validation method. We divided and fixed the available data into three disjoint participant-independent train, validation and test sets, while the baseline is dependent on a user-based 10-fold-like cross-validation.

4.3.2.3 Performance Comparison — Audio Segments Level

In general, the multi-type models that process both breathing and coughing audio perform better than the single-type models (cf. Table 4.3.2). Based on the performance comparison of the single-type model that analyses either a breathing or coughing segment, we find that cough sounds are more informative for COVID-19 detection, resulting in improved detection accuracy, UAR and UF1. Concerning UAP, the single-type model that processes breathing sounds has better performance and a smaller confidence interval. The two conventional fusion methods investigated in the multi-type models show similar accuracy and UAR outcomes. In terms of UAP and UF1 metrics, however, the convolution fusion model, which applies channel convolution to the embeddings of the two audio types, outperforms the simple concatenation method. Moreover, our suggested deep fusion strategy performs the best and surpasses the performance of other fusion methods studied in terms of all the

Table 4.3.3: Performance comparison [$\mu\pm$CI in %] of the models trained on the test set, when considering each audio sample as a whole. B+C indicate the performance of the multi-type models. The performance of the multi-type models is differentiated in terms of the fusion method they use. B2C/C2B indicate the models that inject the deep representations learned from the breathing audio segment into the convolutional layers responsible for learning the deep representations of the coughing audio segments, and vice versa.

	ACC	**UAR**	**UAP**	**UF1**
B	71.2 ± 2.4	72.1 ± 3.7	73.5 ± 3.6	71.6 ± 3.1
C	73.3 ± 2.3	73.8 ± 3.7	73.1 ± 3.5	72.8 ± 3.0
B+C – Concat	74.6 ± 2.4	73.8 ± 2.9	69.2 ± 3.4	70.2 ± 3.7
B+C – Conv	76.7 ± 2.2	76.9 ± 3.4	71.3 ± 3.3	72.4 ± 2.8
B+C – B2C	$\mathbf{78.4 \pm 2.4}$	$\mathbf{78.3 \pm 3.5}$	77.6 ± 3.6	$\mathbf{78.0 \pm 2.7}$
B+C – C2B	$\mathbf{78.4 \pm 2.0}$	77.6 ± 3.1	$\mathbf{79.7 \pm 2.9}$	$\mathbf{78.0 \pm 2.6}$

evaluation metrics.

With the inclusion of cough information into the learning of an individual's breathing feature, the learnt representation is more discriminative for a better COVID19 detection than without using cough information. Similarly, when learning a cough representation, incorporating breathing information can result in a considerable gain, as shown by comparing it to the feature retrieved from cough sounds alone. Intuitively, deep fusion promotes a more thorough coupling between the two audio types in comparison to the conventional fusion techniques, yielding to an overall improvement in detection performance.

4.3.2.4 Performance Comparison — Audio Samples Level

Regarding audio sample level, we examine further the effectiveness of deep fusion. When an audio sample can be split into multiple segments, our deep fusion model predict COVID-19 for each segment individually, resulting in a sequential COVID-19 predictions. The audio sample corresponds to a COVID-19 patient if the majority of the predictions in this sequence are detected positive. Similar to the findings of audio segment level, the performance of multi-type models outperforms that of single-type models, and our deep fusion model dominates in all the performance measures analysed. The deep fusion mechanism yields superior performance. The accuracy and UF1 scores are comparable whether coughing information is introduced into the learning of breathing representation or vice versa. Incorporating breathing information into the learning of coughing data results in a higher UAR, but the reverse fusion produces a better UAP result.

4.3.3 Section Summary

In this part, we presented a novel CNN-based multi-type feature fusion method. The approach is successfully adopted to the COVID-19 detection by combining breathing and coughing information from the same patient. It surpasses approaches utilising a single audio type, and the newly presented deep fusion yields superior detection results when compared to the two conventional fusion methods examined. Future research should focus on the verifying this methodology with more COVID-19 related datasets encompassing breathing and coughing audio. Considering the impact of information fusion method on the overall model's performance, future study might target at the creation of new information-fusion mechanisms that make greater use of the complementarity between diverse data types.

4.4 COVID-19 Detection using Enhanced Speech

The objective of this section is to integrate the audio enhancement solution presented in Section 3.4 into a speech-based COVID-19 detection system in an attempt to make speech captured in noisy environments of everyday life useful for the disease diagnosis.

4.4.1 Network Architectures

The model for speech enhancement has the same architecture as the U-Net presented in Section 3.4.1.1. The architecture of the COVID-19 detection model is based on a ResNet18 model [267] with the initialisation of using the pre-trained weights. Using a using a following dense layer to shrink the speech information into a more compact representation, the output dimension is reduced to 16. The final classification is accomplished using two fully-connected layers with a dropout rate of 0.3. Following the first layer, the output is activated using a ReLU function, and then fed into the second layer to produce two output neurons that correspond to the probability scores of COVID-postive and -negative predictions.

4.4.2 Experiments & Evaluation

First, we test the robustness of the COVID-19 model against several levels of noise. To do this, we augment the DiCOVA [268] test set with chosen environmental recordings from AudioSet [146]. We then perform speech enhancement using a U-Net previously trained for ASR on the created noisy data, with the expectation that the enhanced speech would have a higher audio quality, hence enhancing the stability of COVID-19 detection from speech. By augmenting the training speech from DiCOVA with environmental noises, we hope to improve the robustness of the COVID-19 model. Finally, we evaluate the performance of our joint optimisation approaches in comparison to these baseline methods.

4.4.2.1 Data Description & Processing

The DiCOVA corpus comprises coughing, breathing and speech recordings collected from individuals with and without COVID-19 infection in several countries using an online application [268]. Only the number counting speech recordings are taken into the our investigation. The corpus has its own data partitioning, with 172 confirmed positive individuals out of 965 in the development set, and 71 positive patients out of 471 in the evaluation set.

To synthesise the noisy samples for training and testing, we mix each speech recording from DiCOVA with an AudioSet sample using an SNR ranging from $0, 5, 10, 15, 20, 25$dB. During training, a random SNR is chosen for synthesising each speech sample in order to maximise the overall generalisation ability of the trained model. At test, the model performance is assessed in terms of all SNRs considered. As input to the COVID-19 model, the logarithmic values of the speech spectrogram are computed.

4.4.2.2 Evaluation Metrics

As suggested by the DiCOVA challenge, we use Area Under the Curve (AUC) as our performance measure. AUC reveals a classifier's ability to differentiate between two classes,

Table 4.4.1: Testing results, AUC [%], using DiCOVA and selected samples from AudioSet corpus. DA stands for the method using only data augmentation. MTL represents the proposed multi-task learning solution.

Methods	Inf	25dB	20dB	15dB	10dB	5dB	0dB	average
original	81.85	74.16	73.48	69.22	65.69	61.85	56.67	66.84
Cold Cascade	-	70.93	70.70	68.01	65.72	64.99	58.08	66.57
Cold Cascade + DA	-	78.42	76.33	73.65	70.02	68.48	66.74	72.27
MTL	-	**81.73**	80.62	**76.98**	**74.59**	74.45	71.15	76.59
iterative optimisation	-	81.35	**81.01**	76.49	74.48	**74.73**	**73.12**	**76.87**

and it summarises the Receiver Operator Characteristic (ROC) curve, which illustrates the probability curve of TPR versus FPR at different threshold values. A higher AUC score indicates that the model is more effective at distinguishing data from two classes.

4.4.2.3 Results Analysis

According to [267], the implemented ResNet-18 can get an AUC of 81.85 % on the clean testing data of DiCOVA (cf. Table 4.4.1). This model is however susceptible to noise disruption, with even a tiny noise (SNR = 25dB) causing an AUC drop of more than 7 %. As the noise rises, the detection performance gradually diminishes until it reaches an AUC of 56.67 % at the SNR of 0dB. Applying an independently trained SE model to the frontend of the COVID-19 model cannot improve the average AUC result. In particular, although the frontend enhancement has some favourable effects in circumstances with low SNRs, such as 0 and 5dB, the audio distortions introduced by the SE system can hinder the COVID-19 diagnosis in the cases with high SNRs. Using the augmented data, i. e., adding noise to speech data of the DiCOVA training set, the noise robustness of the model can be boosted, yielding an average AUC of 72.27 % and improved results across all the SNR conditions. Particularly for the low SNR cases, such as 0dB, the detection performance is improved by more than 10 %.

Our two presented joint optimisation approaches, multi-task learning and iterative optimisation, are able to further enhance the detection, yielding an average AUC of 76.59 % and 76.87 %, respectively. For high SNR cases, such as 20 and 25dB, both approaches can reach a COVID-19 diagnostic success rate comparable to the performance of the original detection model on the clean test set. The iterative optimisation method surpasses the conventional MTL method in conditions with very low SNR like 0dB, demonstrating its advantage in more noisy environments. Overall, the two solutions jointly optimise the models for audio enhancement and COVID-19 detection, resulting in AUC performance gains of over 4 %.

4.4.3 Section Summary

This section explored speech-based COVID-19 detection, with a focus on the model's noise tolerance. We extended the joint optimisation approaches given in Section 3.4 to this endeavour. Experimental findings support that a task-specific speech enhancement system can efficiently recover speech signal from noisy recordings to improve COVID-19 identification performance. Although both solutions have been previously validated for other audio applications, the particular optimisation of the audio enhancement model towards the COVID-19 task substantially boost the detection performance, producing comparable results to the same model processing clean audio.

4.5 COVID-19 Detection using Heart Rate Measurements

Wearable fitness trackers can estimate parameters such as heart rate up to 24 hours per day, enabling for the monitoring of individuals with diverse health states, lifestyles, and demographic variables. The quantity and quality of remotely gathered data has the potential to improve our knowledge of the correlations between a variety of health conditions [269]. Deep learning algorithms, which benefit from large-scale data, can make substantial contributions in this area [270]. Particularly, it has shown progress in the context of infectious diseases, such as COVID-19, allowing the individual screening and population-level surveillance while minimising contact with infected individuals [259, 260, 261, 256].

We aim to apply deep learning techniques to such heart rate data to predict the presence of COVID-19 symptoms. Considering the prevalence of data imbalance, we frame the task as anomaly detection, and explore the use of a convolutional auto-encoder (CAE) with contrastive loss [271, 272]. Specifically, the contrastive loss is used to guide training of the CAE to produce high reconstruction error for positive (symptomatic) input pairs. The method strengthens the model's ability to learn discriminative latent attributes for distinct classes compared to some typical neural network architectures, including simple multi-layer perception (MLP), long short-term memory (LSTM), convolutional neural networks (CNNs), and a standard (CAE) [1] without applying contrastive loss. Our experiments are based on the heart rate measurements collected as part of IMI2 RADAR-CNS programme[15] conducted at multiple clinical sites in several European countries. In addition, we conduct a series of ablation studies to investigate the critical aspects that contribute to the successful adoption of this methodology, especially the setting of the margin value and the necessity for pre-training.

4.5.1 Network Architectures & Training Objectives

As an approach for learning data representations, CAE [273] contains a CNN to encode latent attributes of the input feature map, and a second CNN to reconstruct the original input based on the learnt attributes. As a bottleneck imposed by this model, it is critical to determine the ideal dimension of the latent attributes in order to achieve a reasonable balance between the comprehensiveness and discrimination of the representations. To optimise an auto-encoder network, the reconstruction error between the decoder output and the original input is minimised, by doing this to train the decoder to reproduce the original input from the compressed knowledge representation.

However, this method for optimising a standard CAE via unsupervised learning disregards class differences. To incorporate the class information into training, we exploit a contrastive loss [39] instead of reconstruction error, such as Mean Square Error (MSE), to assist the model in learning sufficiently discriminative latent attributes for different classes.

[15] `https://www.radar-cns.org/`

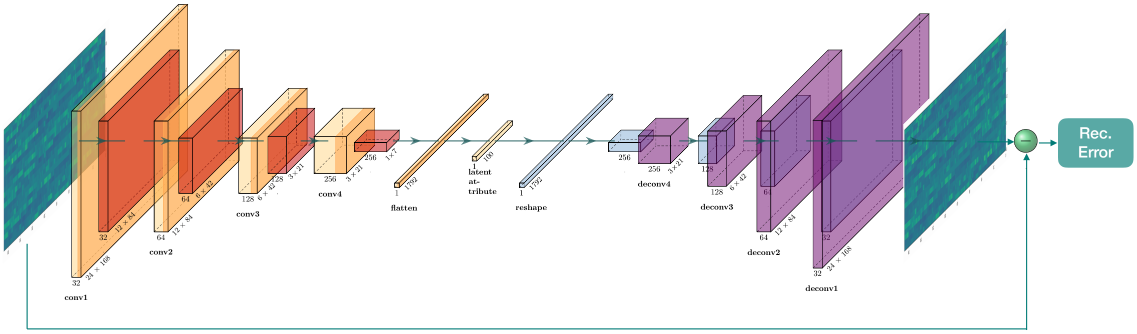


Fig. 4.5.1: The convolutional auto-encoder (CAE) architecture with 4 encoder layers and 4 decoder layers as an example. An encoder layer is a sequence of **convolution** – **batch-normalisation** – **PReLU** – **max-pooling**. A decoder layer is a sequence of **transposed convolution** – **batch-normalisation** – **PReLU** – **transposed max-pooling**. The distance between the original and reconstructed image represents the reconstruction error. (*Source*:[274])

4.5.1.1 Convolutional Auto-encoder

The architecture of our CAE is depicted in Figure 4.5.1, and the encoder and decoder specifications are given in Table 4.5.1. Each encoder layer is made up of a convolutional layer, batch normalisation, PReLU and max pooling. The kernel size and stride determine the receptive field, i. e., the perceptual scope on the original input, which is indicative of the convolutional layer' capability to represent data. Additionally, the representation diversity correlates to the number of kernel filters. By preforming batch normalisation, we can limit the impact of internal covariate shift (ICS), which is caused by distinct training data batches with slightly different distributions [22]. The use of parametric rectified linear unit (PReLU) [275] activation function keeps the fast model convergence of ReLU [276], while using a learnable slope parameter to prevent the frequently observed issue of dead neurons in DNN training. Max-pooling serves to compress the activations into more compact representations with reduced feature size.

We denote the encoding process by $f^{\mathbf{enc}}(\cdot)$ for given features $[x_1, x_2, ..., x_i, ..., x_N]$ extracted from N heart rate segments. Its flattened output is projected to latent attributes using a fully-connected layer:

$$h = f^{\mathbf{enc}}(x_i). \tag{1}$$

The decoder targets at recreating the original input from the latent attributes h; hence, it presents a symmetric structure from the encoder, which can be seen as an inverse of the encoding. To do this, the feature map of each decoding layer is subjected to transposed convolution and transposed max-pooling in order to recover the feature maps to the same size as its corresponding encoding layer. The decoding process is represented as

$$\hat{x}_i = f^{\mathbf{dec}}(h). \tag{2}$$

Note that we describe our CAE structure using a 4-layer example, since it provides the best experimental detection performance. The results of different numbers of convolutional layers are further compared in experiments. The last layer of the encoder controls the size of the flattened encoder output. We add a subsequent dense layer to alter its dimension for

Table 4.5.1: Specifications of our CAE models. Each convolution and pooling layer, as well as de-convolution and de-pooling layer contains its own kernel size, stride, padding size, and number of channels. *=dimensionality depends on the total number of layers, **= dimensionality of latent attributes. fc abbreviates fully-connected layer. (*Source*: [274])

	Blocks	Kernel	Stride	Padding	# Channels
Encoder	**conv1**	$(5,5)$	$(1,1)$	$(2,2)$	32
	pool1	$(2,2)$	$(2,2)$	–	32
	conv2	$(5,5)$	$(1,1)$	$(2,2)$	64
	pool2	$(2,2)$	$(2,2)$	–	64
	conv3	$(5,5)$	$(1,1)$	$(2,2)$	128
	pool3	$(2,2)$	$(2,2)$	–	128
	conv4	$(5,5)$	$(1,1)$	$(2,2)$	256
	pool4	$(3,3)$	$(3,3)$	–	256
	conv5	$(3,3)$	$(1,1)$	$(1,1)$	512
	conv6	$(3,3)$	$(1,1)$	$(1,1)$	1024
	flatten			$*$	
	fc			$**$	
Decoder	**fc**			$**$	
	deconv6	$(3,3)$	$(1,1)$	$(1,1)$	512
	deconv5	$(3,3)$	$(1,1)$	$(1,1)$	256
	deconv4	$(3,3)$	$(1,1)$	$(1,1)$	128
	depool3	$(3,3)$	$(3,3)$	–	128
	deconv4	$(5,5)$	$(1,1)$	$(2,2)$	64
	depool4	$(2,2)$	$(2,2)$	–	64
	deconv5	$(5,5)$	$(1,1)$	$(2,2)$	32
	depool5	$(2,2)$	$(2,2)$	–	32
	deconv6	$(5,5)$	$(1,1)$	$(2,2)$	1
	depool6	$(2,2)$	$(2,2)$	–	1

more flexibility in optimising the CAE model.

Typically, an auto-encoder is optimised by minimising its reconstruction error, for example, root mean squared error (RMSE):

$$\mathbf{RMSE} = \sqrt{\frac{1}{N}\sum_{i}^{N}|x_i - \hat{x}_i|^2}. \tag{3}$$

However, this kind of training losses makes it challenging to find a suitable latent attributes, i. e., to adjust a proper dimension. Due to the absence of class information during the auto-encoder optimisation, too short latent attributes may have inadequate representation capability, while setting it too long may embed the information with a great deal of redundancy that, though helpful for input reconstruction, falls short of concentrating the learning of the saliently features to differentiate between classes. In particular, for our COVID-19 detection task based on heart rate measurements, the auto-encoder may have a tendency to learn the latent attributes that better reconstruct the original pattern, while ignoring the salient attributes that indicate the distinctions between symptomatic and asymptomatic segments.

The deficiency of class information in auto-encoder optimisation has been discussed in previous work [190]. To alleviate this problem, it is recommended that class information be added to the latent attribute layer to make a supervised auto-encoder. As is the case with most supervised learning frameworks, cross-entropy loss can be used to optimise the predictions, and is seen as the regularisation term added to the reconstruction error of the auto-encoder. Nonetheless, this joint optimisation necessitates a proper combination factor capable of balancing the convergence of the two losses involved, which stem from different stages of the auto-encoder model.

4.5.1.2 Contrastive Loss

To enable the CAE to learn latent attributes that are more discriminative between distinct classes, we integrate class information – symptomatic and asymptomatic – into its optimisation by fitting the reconstruction error of the two classes into a contrastive loss [39]. Similar to anomaly detection, it is expected that the CAE would generate a margin difference between the reconstruction errors of the symptomatic and asymptomatic segments, namely a low reconstruction error for asymptomatic segments, and a large reconstruction error for symptomatic segments. To achieve this, the loss function is expressed as

$$\mathbf{Loss} = \sqrt{\frac{1}{N}\sum_{i}^{N}|x_i^n - \hat{x}_i^n|^2} + (\mathbf{m} - \sqrt{\frac{1}{N}\sum_{i}^{N}|x_i^p - \hat{x}_i^p|^2}), \tag{4}$$

where p and n are used to differentiate positive (symptomatic) from negative (asymptomatic) samples.

A typical anomaly detection task involves training an auto-encoder with negative samples only, with the expectation that it would generate a low reconstruction error when processing a negative sample during testing. Since the model has not been exposed to the pattern of a positive sample during training, a large reconstruction error can be expected when it encounters a positive sample. Unlike this, we include both positive and negative samples for model training, and build a contrastive loss function to imitate the effect of anomaly detection (cf. Eq. 4). The reconstruction error for a negative pair, i. e., an original feature map and its reconstructed image for an asymptomatic segment, is suppressed to 0, indicating a successful feature reconstruction. In contrast, the reconstruction error for a positive input pair should converge to a margin value of **m**. The selection of this margin value is crucial to the success of a contrastive CAE, we thus will discuss the effect of different **m** on the model convergence in Section 4.5.2.8. Within this training method, both positive and negative data contribute to the CAE optimisation, and the final model is capable of producing effective anomaly detection. Consequently, we can directly perform classification based on the reconstruction errors using classic machine learning techniques, such as logistic regression.

4.5.1.3 Comparison Methods

The performance of the presented model, contrastive CAE, is compared to that of several classic neural network prototypes, including an MLP, LSTM models operating on 1D and 2D feature types, a CNN with the same architecture as the CAE encoder, and a CAE optimised using RMSE. Additionally, by applying MLP classifiers to the learnt latent attributes, we can compare the quality of the latent attributes generated using our approach to that from a conventional CAE. For this, we explore dimensions of 50, 100, 300, 500, and 1 000 for the latent attributes in our experiments. A 2-layers MLP is used to project the latent attributes of different lengths to classes – symptomatic and asymptomatic.

4.5.2 Experiments & Evaluation

We conduct experiments to access the proposed method for identifying COVID-19 using heart rate data acquired from individuals wearing Fitbit smartwatches. In addition to comparing our model with other typical neural network models, we analyse the contribution of each model component to the successful adoption.

4.5.2.1 Data Description & Processing

The heart rate data for this study was constantly recorded 24 hours a day, 7 days a week using a Fitbit Charge 2 or Charge 3 device connected with participants' own Android smartphones when available, or a Motorola G5, G6, or G7 given, to disseminate questionnaires [277]. To assess the impact of COVID-19, a specific active questionnaire was given to all active RADAR participants on March 25, 2020 and April 8, 2020, separately. We base our study on the Fitbit heart rate measurements of 87 volunteers over the course of ninety days, from 21 February to 20 May 2020. The participants from Denmark, Italy or Spain ranged in age of 23 to 73 (mean = 46.5 ± 10.5 standard deviation).

According to [277], two criteria are utilised to determine the presence of COVID-19 among participants . The first case definition (CD1) states that the participants experienced fever or anosmia/ageusia in addition to any other COVID-19 symptoms, such as respiratory symptoms, fatigue, and gastrointestinal symptoms, or respiratory symptoms plus two other COVID-19 symptoms. The second case definition (CD2) applies to patients who had fever and any additional COVID-19 symptoms, or respiratory symptoms in combination with anosmia/ageusia. Among the 87 individuals who contributed data to this study, 30 female and 38 male participants reported COVID-19 symptoms, including 49 patients who did not fulfil CD1 or CD2 criteria. In addition, each symptomatic patient is paired with a symptom-free control participant matched for site, gender and a similar age. In Table 4.5.2, we provide a summary of the participants counts for each data partition, together with gender, age, and location information.

In Figure 4.5.2, we demonstrate the approach for segmenting and preprocessing the heart rate data of a participant who reported COVID-19 symptoms. The heart rate measurement is divided into temporal segments, each of which spans 7 days prior and after the symptom onset. We aim to identify changes in heart rate associated to COVID-19 infection that are indicative of the illness. According to the research published in [278, 279, 258], the choice of this interval meant to cover a COVID-19 incubation period. It may help to compress the anomalous effects of daily variations in participants' activity, such as those often seen between weekdays and weekends [280]. Specifically, a *symptomatic segment* refers to the heart rate segment spanning 14 days centred at the beginning (0:00) of the day of

Table 4.5.2: Gender-, age-, and site-related distribution of participants per data subset. (*Source*: [274])

		Pre-training	Positive participants for testing	Health control for testing
Genders	**Female**	14	5	5
	Male	35	14	14
Locations	**Italy**	18	7	7
	Spain	19	6	6
	Denmark	12	6	6
Ages	**≤ 30**	1	2	2
	30 - 39	10	3	4
	40 - 49	12	6	5
	50 - 59	19	6	6
	60 - 69	6	1	1
	≥ 70	1	–	–

the reported symptom onset (red box on top of Figure 4.5.2), whereas an *asymptomatic segment* stands for any consecutive 14-days heart rate data (starting from 0 o'clock of a possible day) that is at least 7 days apart from a symptomatic segment (green box in top of Figure 4.5.2).

To make full use of a heart rate measurement, asymptomatic segments are generated by incrementally sliding a segmentation window across data sections that are at least 7 days away from the symptomatic segment. The 7-day separation between asymptomatic segment and the symptomatic segments account for two reasons: first, a participant might not have been infected 14 or more days previous to the beginning of symptoms; and second, the individuals may have fully recovered 14 days after the onset of symptoms. Besides, a *control segment* is truncated from the segment of the same time from a control subject. Consequently, we can achieve 49 symptomatic segments and 1 470 asymptomatic segments from the 49 participants in the pre-training set. Taking into account the disparity between the number of available symptomatic and asymptomatic segments, we up-sample the symptomatic segments to match that size of asymptomatic segments, in this way to bias the model detection towards the minority class. For testing the model, each of the 19 patients who reported symptoms contributes a symptomatic segment. Finally, 1710 asymptomatic segments are extracted, including 570 segments from these 19 patients, and 1 140 from the controls. In Table 4.5.3, we detail the available symptomatic and asymptomatic segments , as well as the data completeness for each partition.

The heart rate estimations should ideally be sent to the Radar sever every five seconds (blue curve in the middle of Figure 4.5.2). To get a smoother signal pattern, the heart rate measurement is averaged every 5 minutes. The average result should still be able to track slow short-term changes in the heart rate. Additionally, this smoothing step helps alleviate two concerns associated with Fitbit data collection: (1) discrepancies in sampling rates of heart rate estimates, and (2) missing values observed in real-life conditions. Both of these issues make it impossible to compare the features in comparison to the mean heart rate in 5-minutes intervals. To mitigate the impact of missing data for a whole 5-

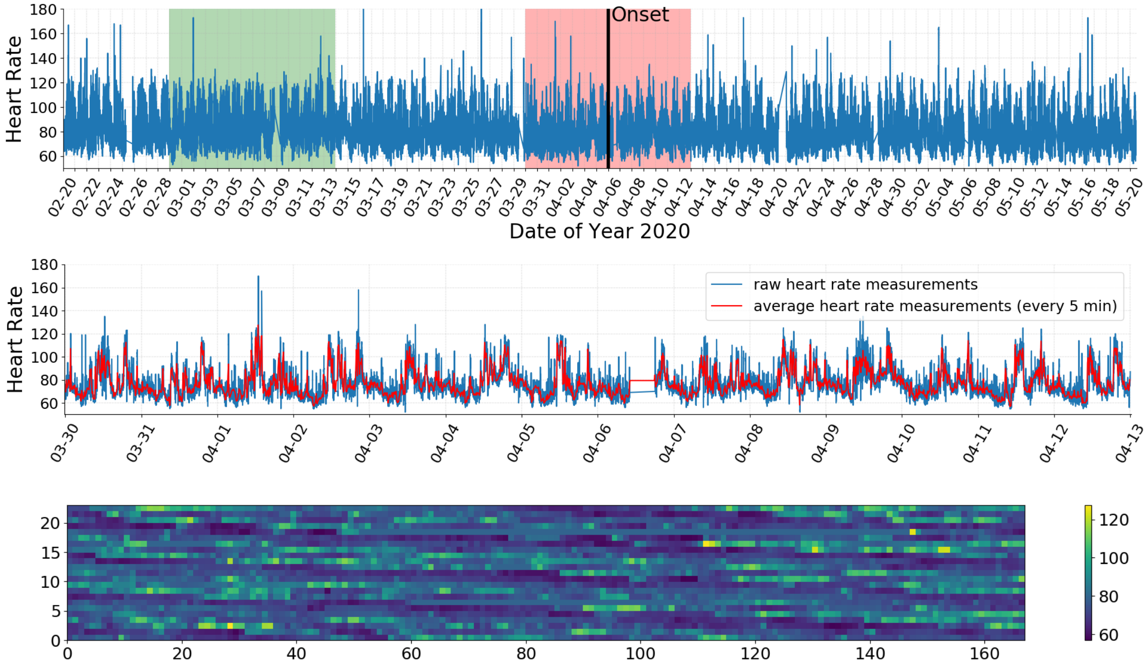


Fig. 4.5.2: Segmentation and pre-processing of heart rate data of a participant with reported COVID-19-like symptoms. **Top**: Heart rate data recorded 24-hours-a-day/7-days-a-week from 21 February to 20 May 2020 (total 90 days). *Onset* (black vertical bar) indicates 0 o'clock at the reported symptom onset date. Red rectangle – 7 days heart rate data before and after symptom onset representing a symptomatic segment; green rectangle – asymptomatic segment. **Middle**: Symptomatic segment. Blue curve – unprocessed heart rate trajectory of the red rectangle above; red curve – heart rate trajectory averaged over 5-minutes intervals. **Bottom**: Representation of the symptomatic segment as 24×168 sized image of 5-minutes heart rate data related pixels. Each column represents an interval of 2 hours, the 168 columns sum up to 14 days. (*Source*: [274])

minutes period, the median estimate of the 14-day segment is substituted, giving robustness against outliers when comparing to the mean value. Despite the completeness of heart rate segments (cf. Table 4.5.3), missing data may occur across an entire heart rate segment. In this situation, a smoothing interval that is too short may result in more empty mean values, whereas a too long interval can cause loss of information on the variations within the heart rate segments. The resultant smoothed heart rate trajectory, consisting of a single heart rate value every 5 minutes (red curve in the middle of Figure 4.5.2), is suitable for modelling the global heart rate patterns associated with COVID-19 symptoms. Subsequently, we convert the averaged 14-day heart rate segment into a 24×168 image feature (bottom of Figure 4.5.2). Each column encodes a heart rate trajectory of two hours (24×5 minutes), and each pixel of the image represents a heart rate mean value of five minutes. We experimentally confirmed the use of this feature size can yield promising detection results.

4.5.2.2 Training Settings

The data from the 49 patients who reported their COVID-19 symptoms but did not satisfy CD1 or CD2 criteria are used to pre-train our model throughout the experiments. Leave

Table 4.5.3: Available symptomatic and asymptomatic segments per data subset. Data completeness [%] of respective heart rate segments is given in parentheses (mean + std). (*Source*: [274])

# (%)	Pre-training	Positive participants for testing	Health control for testing
Symptomatic	49 (98.7 ± 0.3)	19 (97.6 ± 0.2)	–
Asymptomatic	1470 (98.1 ± 0.4)	570 (97.4 ± 0.2)	1140 (99.2 ± 0.5)

one subject out (LOSO) cross-validation (CV) is then applied to evaluate the models using the data from 19 patients whose symptoms satisfy CD1 or CD2 criteria, and the matching symptom-free control group. In particular, for each round of the 19-fold LOSO CV, the pre-trained models were fine-tuned using the data of 18 patients with COVID-19 symptoms and their 18 control participants without symptoms. Then, the model is tested on the remaining pair of symptomatic and asymptomatic participants.

The models presented are optimised using an Adam optimiser with a learning rate fading from 0.03 to about 0.0001 per 50 epochs, corresponding to a decay factor of 0.33. During training, we use a constant batch size of 32. These hyper-parameters are meticulously chosen to ensure the model convergence.

4.5.2.3 Evaluation Metrics

To make fair comparisons, we use mean unweighted average recall (UAR), sensitivity, and specificity, the area under receiver operating characteristic curve (AUC-ROC), and Matthews correlation coefficient (MCC) to assess all the models implemented. The Matthews correlation coefficient serves as a measure of the quality of binary classifications that accounts for true and false positives and negatives. Hence, it is typically considered as a balanced metric that can be used even when the class sizes are vastly different. A MCC value of $+1$ indicates a prefect prediction, 0 reveals an average random prediction, and -1 implies an inverse prediction.

4.5.2.4 Comparison to Classic Neural Networks

As comparison methods, we first choose neural networks that can function on one-dimensional signals (noted as “1D” in Table 4.5.4), such as the MLP and LSTM models [1]. These models impose no extra need to transform the smoothed signal into two-dimensional image (noted as “2D” in Table 4.5.4). The MLP classifier performs optimally when four layers are used, with the number of hidden units in each layer decreasing along the model depth. More effective is the LSTM model with 64 hidden units in the recurrent cell. Another LSTM model that processes the two-dimensional feature map created as described in Section 4.5.2.1 can attain its best performance when its recurrent cell has 128 hidden units. The CNN model, whose architecture is identical to the encoder of our CAE, outperforms the LSTM model in terms of all the evaluation metrics considered, and also exhibits significant improvements over the two models operating on 1D heart rate data in paired t-tests with a significance level α equals to 0.05.

Our contrastive CAE with up to six layers in its encoder and decoder is evaluated. To do this, logistic regression is used to classify the reconstruction errors of the test data. Using a model with two encoder and two decoder layers, the contrastive CAE is able to provide

Table 4.5.4: Evaluation results for the binary COVID-19 yes/no (based on the symptom CD1/CD2 definitions above) classification [%] of the baseline methods and contrastive CAE models with a different number of (#) layers. For the contrastive CAE, classification is performed based on reconstruction error using logistic regression. (*Source*: [274])

	#Layers	UAR	Sensitivity	Specificity	AUC-ROC	MCC
MLP (1D)		61.0	63.2	58.8	0.542	0.046
LSTM (1D)		67.3	73.7	61.0	0.577	0.074
LSTM (2D)		72.8	73.7	71.9	0.685	0.105
CNN (2D)		**76.0**	**78.9**	**73.1**	**0.705**	**0.122**
Contrastive CAE	1	58.8	70.2	47.4	0.508	0.044
	2	83.0	84.2	81.9	0.769	0.176
	3	90.6	**100.0**	81.3	0.878	0.213
	4	**95.3**	**100.0**	**90.6**	**0.944**	**0.310**
	5	93.9	**100.0**	87.7	0.931	0.270
	6	90.9	**100.0**	81.9	0.883	0.217

considerable detection improvements compared to baseline methods. With four layers of encoder and decoder, the CAE reaches its greatest detection results. Consequently, the final UAR of 95.3%, sensitivity of 100.0%, specificity of 90.6%, AUC-ROC of 0.944, and MCC of 0.310 are obtained, demonstrating significant improvements over the CNN approaches in paired t-tests ($p < 0.05$).

4.5.2.5 Comparison to Conventional CAE

Next, we analyse the efficacy of optimising the CAE with a contrastive loss (cf. (Equation (4))) as opposed to RMSE loss (cf. Equation (3)) w. r. t. the quality of the latent attributes. In order to evaluate the quality of the latent attributes, a two-layer MLP classifier is trained to project them onto classes, i. e., symptomatic and asymptomatic. Additional comparison is accomplished by performing classification directly on the reconstruction errors of contrastive CAE.

The conventional CAE model obtains its optimum UAR, specificity, and MCC when the size of the latent attributes is set to 50, and its optimum sensitivity and AUC-ROC when the size is 500. The classification results indicate, however, that this model is incapable of acquiring discriminative latent attributes for distinct classes. Due to the absence of class information during the extraction of these latent attributes, the classification performance is reliant on the MLP classifiers. Its performance is even lower than that of the CNN model (cf. Table 4.5.4), confirming the necessity to incorporate the class information into the CAE training. The binary classes can be integrated in the contrastive loss as given in Equation (4). It guides the CAE training to create a margin between the positive and negative reconstruction errors. To achieve this, the contrastive CAE must learn latent attributes that carry salient information to differentiate between symptomatic and asymptomatic segments. In our experiments, the contrastive CAE with an attribute size of 100 yields the

Table 4.5.5: Comparison of results [%] between convolutional auto-encoders (CAEs) with 4 encoder and 4 decoder layers trained with RMSE loss vs contrastive loss. Classification is performed based on the latent attributes. # Attr: dimensionality of latent attributes. (*Source*: [274])

	# Attr	UAR	Sensitivity	Specificity	AUC-ROC	MCC
CAE	50	**66.6**	57.9	**75.4**	0.545	**0.080**
	100	58.5	47.4	69.5	0.465	0.038
	300	63.4	63.2	63.7	0.527	0.058
	500	65.8	**68.4**	63.2	**0.591**	0.068
	1000	55.3	47.4	63.2	0.448	0.023
Contrastive CAE	50	92.0	**100.0**	83.9	0.904	0.233
	100	**92.2**	**100.0**	84.3	**0.907**	0.236
	300	90.9	**100.0**	81.9	0.890	0.217
	500	90.9	94.7	**87.1**	0.881	**0.247**
	1000	71.9	68.4	75.4	0.597	0.105

Table 4.5.6: Classification results [%] of the contrastive CAE with 4 encoder and 4 decoder layers based on the reconstruction error (rec. error) using logistic regression. # Attr: dimensionality of latent attributes. The last row indicates removing the latent attributes layer. (*Source*: [274])

	# Attr.	UAR	Sensitivity	Specificity	AUC-ROC	MCC
Contrastive CAE (rec. error)	50	93.9	100.0	87.7	0.927	0.270
	100	**95.3**	**100.0**	**90.6**	**0.944**	**0.310**
	300	91.5	100.0	83.0	0.890	0.226
	500	92.4	100.0	84.8	0.895	0.240
	1000	94.4	100.0	88.9	0.936	0.284
	–	93.3	100.0	86.6	0.923	0.258

best UAR, sensitivity and AUC-ROC. As the size climbs to 500, the proposed approach outperforms the conventional CAE in terms of specificity and MCC by a substantial margin.

Performing classification on the reconstruction errors, rather than the latent attributes, reveals a more straightforward yet effective solution to the task at hand (cf. Table 4.5.6). To determine a decision threshold between positive and negative reconstruction errors, logistic regression is applied to the training set of each LOSO CV round. A positive COVID-19 case (CD1/CD2 criteria) is identified if the reconstruction error of a heart rate segment exceeds the decision boundary. The optimal performance is reached when the attribute size equals 100, with a UAR of 95.3%, sensitivity of 100.0%, specificity of 90.6%, AUC-ROC of 0.944, and MCC of 0.310. Our approach preserves the performance consistency across different attribute sizes, hence reducing the difficulty of finding the most appropriate dimension. In

Table 4.5.7: Test results [%] for shifting the sliding window by days. (*Source*: [274])

	#Days	UAR	Sensitivity	Specificity	AUC-ROC	MCC
Contrastive CAE	−3	57.4	52.6	62.2	0.420	0.032
	−2	64.7	68.4	61.0	0.558	0.063
	−1	95.6	100.0	91.2	0.946	0.320
	0	**95.3**	**100.0**	**90.6**	**0.944**	**0.310**
	1	95.4	100.0	90.8	0.945	0.313
	2	96.1	100.0	92.1	0.957	0.337
	3	94.9	100.0	89.9	0.949	0.298
	4	87.4	94.7	80.2	0.823	0.193
	5	61.5	68.4	54.6	0.517	0.048

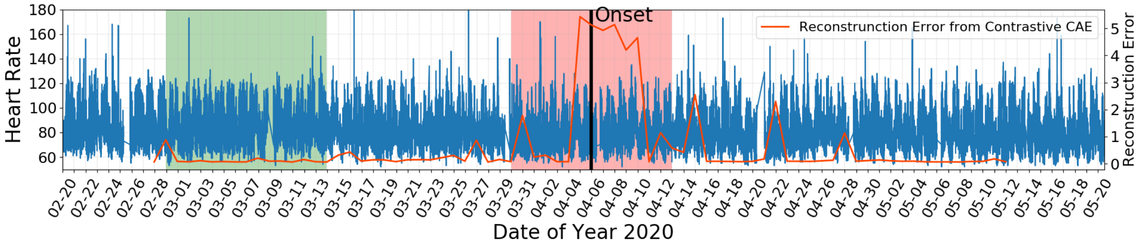


Fig. 4.5.3: Reconstruction errors for continuous binary COVID-19 yes/no classification on 14-days heart rate windows of an exemplary individual (the same as in Figure 4.5.2, top). (*Source*: [274])

fact, excluding the layer of latent attributes from the contrastive CAE (given in the last row of Table 4.5.6) does not diminish the model's detection ability.

4.5.2.6 Decentralised Symptomatic Segments

The preceding experimental findings are contingent on the assumption that the patients reported a positive COVID-19 status at the actual onset of symptoms. As a consequence, our contrastive CAE is optimised for the symptomatic segments centred on the symptom onset. In this part, we explore the possibility of detecting COVID-19 using heart rate segments with a decentralised onset of symptoms. To do this, the 14-day timeframe for slicing the symptomatic segments is shifted to earlier or later days that still encompass the symptom onset. As in prior experiments, the asymptomatic segments remain unchanged.

Our method remains valid when the symptomatic segments are shifted by one day ahead and three days backward (cf. Table 4.5.7). Shifting the sliding window to stray more from the original symptomatic segments, such as by two preceding days or four days later, results in a decrease in classification performance. There are two possible explanations for this observation. On the one hand, perhaps some individuals reported inaccurate onset dates for their symptoms. For example, a patient may have recognised the start of symptoms but

Table 4.5.8: Classification results [%] of the contrastive CAE with 4 encoder and 4 decoder layers based on the reconstruction error (rec. error) using different numbers of (#) participants for pre-training. (*Source*: [274])

	#Participants	UAR	Sensitivity	Specificity	AUC-ROC	MCC
Contrastive CAE (rec. error)	49	95.3	100.0	90.6	0.944	0.310
	40	**95.9**	**100.0**	**91.7**	**0.950**	**0.329**
	30	95.2	100.0	90.3	0.940	0.305
	20	82.3	84.2	80.3	0.823	0.167
	10	79.8	84.2	75.4	0.737	0.143
	0	76.4	78.9	73.8	0.696	0.124

waited days to confirm the infection before reporting the sickness. On the other hand, the symptomatic segments can be shifted up to a few days later, a maximum of three days in our experiments, to increase the assurance that the symptoms are included. Nevertheless, in case the symptoms began earlier but rapidly dissipated, moving the symptomatic segments to many days later may, to some degree, exclude the period containing symptom. Consequently, the final classification results can be impacted (cf. Table 4.5.7). A curve depicting the estimated reconstruction errors is shown in Figure 4.5.3, which identifies the COVID-19 illness using our contrastive CAE to continually on the same data given in the top of Figure 4.5.2.

4.5.2.7 Necessity of Pre-training

Pre-training a neural network can improve the model's generalisation effect, which has been investigated in a variety of machine learning frameworks, including unsupervised learning [281], transfer learning [281] and self-supervised learning [46, 48]. We find that pre-training is crucial for our presented contrastive CAE in order to achieve the model's representation capability. It is worthwhile to evaluate the efficacy of this pre-training with different numbers of participants.

For each number of participants considered for pre-training, a random selection is conducted five times. Using the chosen individuals, we pre-train a contrastive CAE while the LOSO CV assessment is maintained unaltered. The average test results are shown in Table 4.5.8. Using data from a minimum of 30 individuals for pre-training, the model shows promising detection outcomes. As the number of participants for pre-training falls below 20, the classification accuracy decreases considerably, emphasising the need to supply sufficient pre-training data to achieve the optimal performance.

4.5.2.8 The Effect of Margin Size

When utilising the contrastive loss (cf. Equation (4)) to optimise a CAE, the reconstruction errors of a positive and a negative input pair should ideally converge to 0 and $\mathbf{m}$, respectively. The margin $\mathbf{m}$ stands for an expected distance between the two reconstruction errors, which is thus another essential component affecting the classification performance.

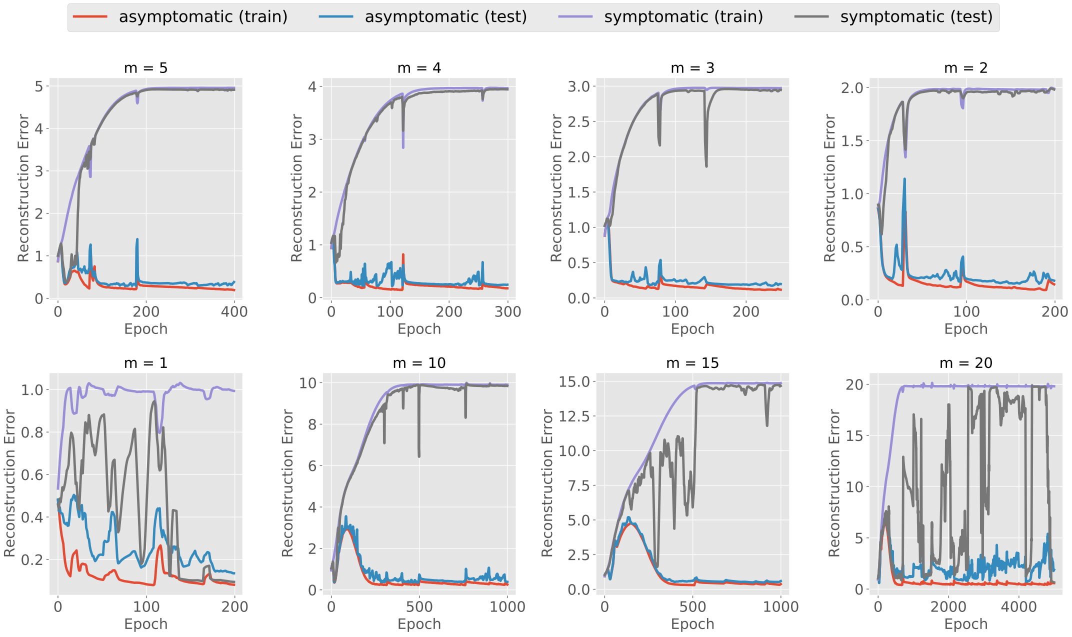


Fig. 4.5.4: Training and testing curves illustrated by the reconstruction errors when using different margin sizes. (*Source*: [274])

To retain model generalisability, its practical optimisation must leave some space around both ideal reconstruction errors, though a smaller fluctuation range usually suggests a more effective convergence. Hence, a too tiny margin, such as $\mathbf{m} = 1$, might be excessively restrictive and limit the allowable fluctuation region during training, preventing the model convergence as seen in Figure 4.5.4. As the margin reaches 2, the model manages to converge after a few training iterations, with a successful creation of the expected margin between the reconstruction errors of the two classes. However, an improperly wide margin, such as $\mathbf{m} = 15$, can cause serious oscillations as the positive reconstruction error approaches its intended margin value. A larger margin can impede the model convergence. A suitable margin size must also account for enough space to determine a decision threshold between the reconstruction errors of the two classes. The classification results for test the influence of various margin sizes are based on our best-performing model (cf. Table 4.5.9).

An intriguing behaviour manifests with the successful training of a contrastive CAE, such as when adjusting the $\mathbf{m}$ to 10 or 15. During the early period of training, the reconstruction errors of both positive and negative samples exhibit similar growing patterns. To a certain point, the two reconstruction errors start diverging before eventually approaching to their respective expected output. This phenomenon can be explained according to Eq. 4. This behaviour is explicable by analysing Equation (4). The optimisation begins with enabling the model to synchronise the encoder input and decoder output, followed by a compromise to create the margin between the positive and negative reconstruction errors. Correspondingly, the two reconstruction errors exhibit parallel growth for the first few epochs, followed by a trade off between the two objectives, i. e., feature reconstruction and margin generation. As a result, the two reconstruction error curves are compelled to diverge.

Table 4.5.9: Classification results [%] of the contrastive CAE with 4 encoder and 4 decoder layers based on the reconstruction error (rec. error) using logistic regression, for using different margin sizes. (*Source*: [274])

	(m)argin	UAR	Sensitivity	Specificity	AUC-ROC	MCC
	2	78.9	84.2	73.6	0.753	0.136
Contrastive	3	91.4	100.0	82.8	0.905	0.224
CAE	4	94.1	100.0	88.2	0.920	0.275
(rec. error)	5	**95.3**	**100.0**	**90.6**	**0.944**	**0.310**
	10	90.5	94.7	86.2	0.861	0.238
	15	90.9	94.7	87.0	0.861	0.247

4.5.3 Section Summary

We presented a contrastive CAE to frame the task of COVID-19 detection given 14-day heart rate measurements into an anomaly detection problem The presence of symptoms is specified by CD1/CD2 criteria. Our approach optimises the CAE using a contrastive loss which integrates class information, It outperformed conventional CNN, CAE and other typical deep learning models for our task. The model was examined using different numbers of layers, and various dimensions of latent attributes. Particular attention was paid to the exploration of the optimal data amount for pre-training and the adjustment of the margin size, both of which were found to be critical for achieving steady convergence and classification performance. In addition to COVID-19 identification, the binary classification method should be generalisable to the prediction of other diseases.

4.6 Summary

In this chapter, we described our deep learning approaches for detecting COVID-19 using cough and breathing sounds, speech data and heart rate measurements, respectively. To boost the noise robustness of the speech-based detection model, we additionally examined the use of audio enhancement to improve speech quality, resulting in improved detection performance in noisy conditions. Due to the availability of continuous data gathering from everyday smart devices such as smartphones and consumer wearables such as smart watches, we anticipate widespread use of our presented methodologies in real-world conditions. Using this technology, it is feasible to continually conduct preliminary illness detection, allowing for timely alerts to be delivered to patients so that they can take appropriate measures, such as managing social isolation or seeking immediate medical treatments.

CHAPTER 5

Face Mask Detection from Speech

5.1 Introduction

Wearing masks in public areas is widely approved as an effective measure to reduce the spread of the COVID-19 virus. Despite the fact that many public COVID-19 epidemic prevention policies are not enforced as strictly as they formerly were, wearing a mask in public is still strong encouraged, even mandatory in closed and crowded venues to ensure public safety. However, its compliance is contingent upon people's commitments. Still, the strategies for continuous mask-wearing monitoring merit additional study. The solutions to automatic detection of whether a person wears a face mask can assist governments worldwide in better monitoring the public respects with the obligatoriness. The study of the detection task using machine learning or deep learning approaches has vastly increased since the pandemic outbreak. A straightforward idea is to take the task as an object detection problem and solve it using computer vision methods [282, 283, 284, 285, 286]. From the perspective of audio, face masks can alter speech properties in frequency since the mask materials absorb speech components differently in different frequency bands. On the other hand, wearing a mask can introduce temporal changes in speech, due to its interference to respiratory process [287, 288, 289], altering the natural tempo, rhythm, and pronunciation speed. Conditioned on these alterations, ML models can be applied to distinguish the speech recorded from a speaker wearing or not wearing a face mask. Previous works for this detection task from speech are based exclusively on the use of CNNs, and have shown promising outcomes. However, CNNs have very limited ability to remember time-dependencies between audio frames, which is very crucial when performing audio signal processing.

To make full use of the temporal dynamics in speech signals, this section presents two effective neural network models to detect surgical masks from speech. Both models are built based on Convolutional Neural Networks (CNNs) to extract the spatial representations of the audio signals. On top of the CNNs, one architecture captures the time-dependencies using a Long Short-Term Memory (LSTM) network, strengthened with an additional attention mechanism, while the other architecture applies a transformer block containing a positional encoder to mark the relative position of a sequence. Furthermore, to assess the complementary effect using both modules, i. e., LSTM and Transformers, to model temporal dynamics, we explore three hybrid models combining them in either a cascade or a parallel structure. Besides, data augmentation techniques, particularly strengthening the transitions between audio frames can advance the performance of our proposed architectures.

In short, this chapter explores potential improvements to mask-wearing detection based on speech with regard to the following aspects:

- Capturing the temporal dependence of speech signals to advance the current state-of-the-art in mask-wearing detection.

- Enriching the temporal dynamics of speech representations by extracting the transient information between audio frames.
- The influence of audio data augmentation techniques, such as SpecAugment, on the detection performance.
- The performance of gender-specific model for this task, and the effectiveness of approaching the challenge in a multi-task framework.

5.2 Related Work

CV Solutions to Face Mask Detection
To identify whether a person is wearing a face mask or not, deep learning techniques that analyse a face image begin processing with detecting or cropping the face region and then sending the cropped outcome to a classifier [290, 291, 284, 285]. Using this strategy, several common CNN architectures, including AlexNet, VGG16/19, ResNet, MobileNet and a customised CNN model were assessed for the task [292]. In [293] and [294], facial representations were learnt from angle-corrected face images so the methods can circumvent the possibility of errors due to face rotations during photography. Goyal *et al.* [295] deployed a CNN-based model to expand this application to multiple-person scenarios. Besides, transfer learning [283, 286] and model assembly [282] that combines machine learning and deep learning methods, have also been investigated for this detection task.

CA Solutions to Face Mask Detection
The research of mask-wearing detection from speech has been carried out from the viewpoints of acoustic features, data augmentation and model architectures [296], primarily based on MASC database collected for the Mask Sub-Challenge (MSC) of the INTERSPEECH 2020 COMputational PARalinguistics challengE (COMPARE). Researchers have made particular efforts to search for feature representations that can differentiate masked speech from regular speech [297, 298, 299]. In [298], Das and Li studied three distinct acoustic features designed to discriminate between speech covered by masks and unmasked speech, resulting in a 1.7 % improvement in classification UAR compared to the baseline method given by the ComParE challenge [300]. Szep and Hariri [297] suggested to enrich the representations by fusing several kinds of acoustic features and assembling three CNN models, VGGNet, ResNet, and DenseNet, in order to improve the detection performance over the approach using a single model. Xu *et al.* [299] aggregated low-level descriptors within the framework of deep neural network, however the process did not lead to further gains in classification performance. Moreover, data augmentation (DA) has been considered in several studies to further enhance the representations [301, 302, 303]. In [301] and [302], a variety of DA techniques, such as SpecAugment or Mixup, that can successfully expand the training data size are compared. Moreover, [302] adapted PANS, a large-scale pretrained audio neural network, for this task, and successfully made further improvements. Using GAN as a DA approach, [303] generates more training utterances for more thoroughly training an ensemble model consisting of a series of ResNets coupled via a *Support Vector Machine* (SVM), surpassing the baseline by an UAR of 2.8 %. Overall, despite these encouraging outcomes on automatic face mask detection from speech, the majority of these earlier investigations have relied solely on CNN methods.

5.3 Face Mask Detection from Speech

In this section, we will first describe the neural network architectures we have to improve the capture of temporal dynamics from speech. Their performance are presented and compared in experiments, with ablation study to analyse the essential components leading to their effectiveness of capturing temporal information.

5.3.1 Network Architectures & Training Objectives

We construct our models by stacking a CNN for analysing spatial features and a second network for processing the temporal information of speech input. To do this, the *attentive convLSTM* model makes use of a bidirectional LSTM (BiLSTM) network with a multi-head attention mechanism to capture the audio temporal dynamics. The alternative approach, *conventional transformer*, employs the encoder of a standard transformer [4], which embeds the temporal information, i. e., the relative position of audio frames, using a positional encoder. In addition, we examine three hybrid models that integrate both a LSTM and a transformer on top of a CNN, by cascading them in a sequence or aligning them in parallel. It is expected that these hybrid architectures would incorporate the advantages of both modules, making them more efficient at acquiring temporal information.

Following is a description of the architectures of our proposed attentive convLSTM (cf. Section 5.3.1.1) and convolutional transformer (cf. Section 5.3.1.2). Then, we present the methods for building the hybrid models (cf. Section 5.3.1.3 and Section 5.3.1.4). The combination of the introduced LSTM with attention and transformer networks in two categorises: cascade mode and parallel mode, according to their relative position alignments. The neural network specifications are summarised in Table 5.3.1.

5.3.1.1 Attentive ConvLSTM

The attentive convLSTM, as seen in Figure 5.3.1a, combines a CNN, a BiLSTM network, and a multi-head attention layer. To allow a steady convergence of this deep neural network, we apply skip-connections in the CNN module, similar to ResNet, to learn the spatial features from its input. Given the input $\tilde{\mathbf{M}}$, the process is expressed as

$$\mathbf{M}^{\mathbf{conv}} = f^{\mathbf{conv}}(\tilde{\mathbf{M}}). \tag{1}$$

The output is then fed into a sequential BiLSTM composed of a forward LSTM and a backward LSTM, which produces:

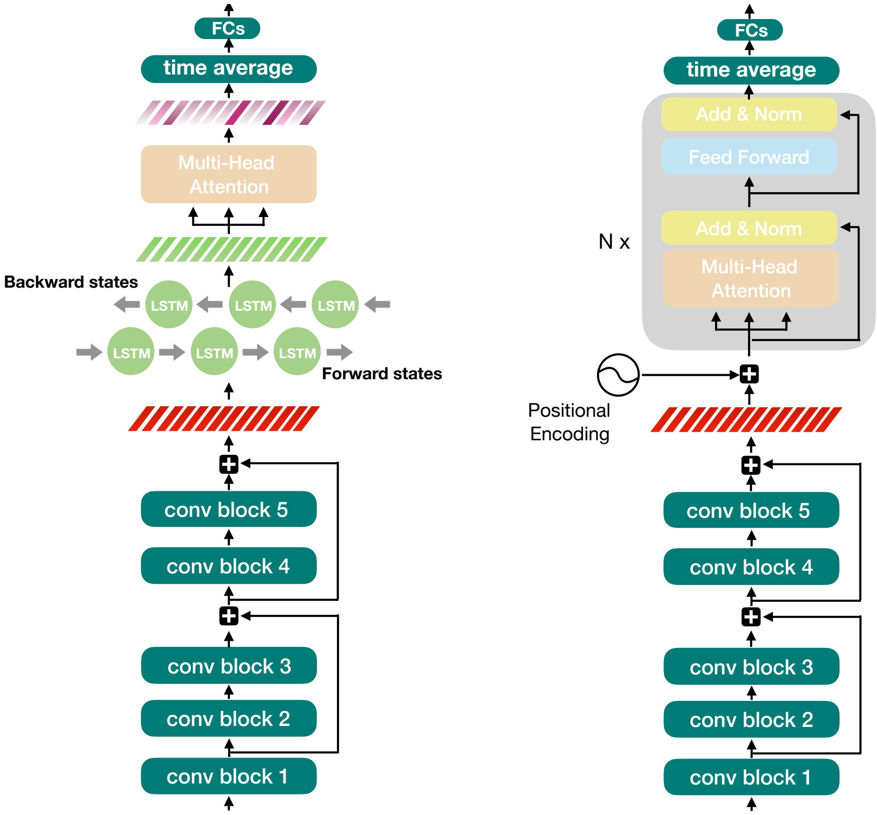
$$\overrightarrow{h_t} = \overrightarrow{\mathbf{LSTM}}(\mathbf{M}_t^{\mathbf{conv}}, \overrightarrow{h_{t-1}}), \tag{2}$$

$$\overleftarrow{h_t} = \overleftarrow{\mathbf{LSTM}}(\mathbf{M}_t^{\mathbf{conv}}, \overleftarrow{h_{t-1}}), \tag{3}$$

where h represents the hidden states of the LSTM cell and t indicates the time-step. The forward and backward outputs are concatenated as

$$h_t = \left[\overrightarrow{h_t}; \overleftarrow{h_t}\right]. \tag{4}$$

To promote the model's capacity to extract temporal information from more salient



(a) Attentive ConvLSTM

(b) Convolutional Transformer

Fig. 5.3.1: The network architectures of mask detection models. (a) Attentive convolutional LSTM, which consists of three cascading modules: a CNN containing 5 convolutional blocks with two skip connections, a bidirectional LSTM, and a multi-head self-attention. (b) Convolutional transformer, which exploits a transformer encoder instead of the BiLSTM and the attention module to capture time-dynamics. Extracted features are averaged along the time axis are then projected onto classes using fully-connected layers. (*Source*: [304])

audio frames, multi-head attention is applied to the LSTM output features

$$H = (h_1, h_2, ...h_T)\,, \tag{5}$$

yielding a more temporally more informative representations

$$\mathbf{MultiHead}(H, H, H). \tag{6}$$

The representations are then averaged across all time-steps and projected onto classes using two fully-connected layers.

5.3.1.2 Convolutional Transformer

A transformer employs an attention mechanism to connect its encoder and decoder, allowing for an efficient global mapping between its sequential input and output. The encoder and decoder are constructed in a similar fashion by stacking a module comprised mostly of a

Table 5.3.1: Specifications of our mask-wearing detection models. (*Source*: [304])

CNN

Block	in_ch	out_ch	kernel	stride	padding
conv1	3	32	$(3,3)$	$(1,1)$	$(1,1)$
conv11	32	32	$(3,3)$	$(1,1)$	$(1,1)$
pool1	32	32	$(2,2)$	$(2,2)$	$(0,0)$
conv2	32	64	$(3,3)$	$(1,1)$	$(1,1)$
conv22	64	64	$(3,3)$	$(1,1)$	$(1,1)$
pool2	64	64	$(1,2)$	$(1,2)$	$(0,0)$
conv3	64	128	$(3,3)$	$(1,1)$	$(1,1)$
conv33	128	128	$(3,3)$	$(1,1)$	$(1,1)$
pool3	128	128	$(1,2)$	$(1,2)$	$(0,0)$
conv4	128	256	$(3,3)$	$(1,1)$	$(1,1)$
conv44	256	256	$(3,3)$	$(1,1)$	$(1,1)$
pool4	256	256	$(1,1)$	$(1,1)$	$(0,0)$
conv5	256	512	$(3,3)$	$(1,1)$	$(1,1)$
conv55	512	512	$(3,3)$	$(1,1)$	$(1,1)$
pool5	512	512	$(1,1)$	$(1,1)$	$(0,0)$
skip13	32	128	$(1,1)$	$(1,1)$	$(0,0)$
pool13	128	128	$(1,4)$	$(1,4)$	$(0,0)$
skip35	128	512	$(1,1)$	$(1,1)$	$(0,0)$
pool35	512	512	$(1,1)$	$(1,1)$	$(0,0)$

LSTM

parameters	values
dim_features	2560
hidden_states	128

Mutli-Head Attention

parameters	values
heads	4

Transformer($\mathbf{N} = 2$)

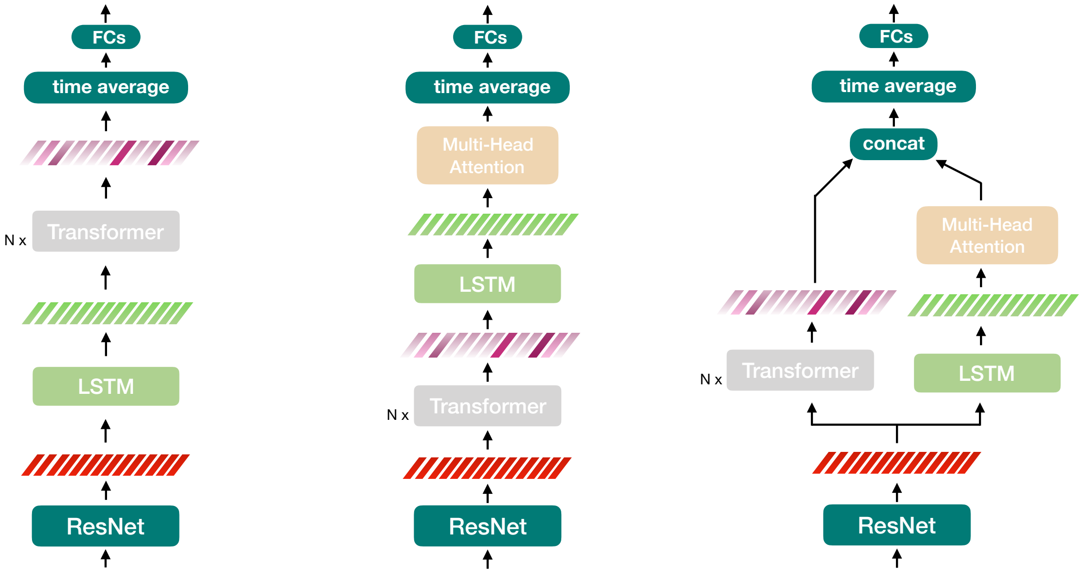
parameters	values
dim_features	2560
heads	4
dim_feed_forward	128
dropout	0.5
dim_fc_in	2560
dim_fc_out	128

FCs

Block	in_features	out_features
fc1	128	64
fc2	64	2

multi-head attention layer and a feed forward component. For a classification problem, we use a transformer encoder only while discarding its decoder for processing the CNN output. Using a positional encoding approach, each time-step of the CNN output is given a unique encoding. The transformer is thus aware of the relative position of each audio frame during processing. A standard positional encoding method is to add

$$p_t^{(i)} = \begin{cases} sin(w_k \cdot t), & \text{if } i = 2k \\ cos(w_k \cdot t), & \text{if } i = 2k+1 \end{cases} \tag{7}$$



(a) ResNet - LSTM - Transformer

(b) ResNet - Transformer - Attentive LSTM

(c) ResNet - Attentive LSTM // Transformer

Fig. 5.3.2: Hybrid model architectures for mask detection: (a) a cascade sequence of **ResNet - LSTM - Transformer**; (b) a cascade sequence of **ResNet - Transformer - Attentive LSTM**; (c) a parallel alignment of **ResNet - Attentive LSTM // Transformer**. (*Source*:[304])

where

$$w_k = \frac{1}{10000^{2k/d}} \tag{8}$$

to the transformer input $\mathbf{M}^{\mathbf{conv}}$. Other more sophisticated or even learnable positional encoder, such as the relative positional encoder presented in previous works [305, 306], have the potential to further progress the convolutional transformer. We adhere to the original positional encoding solution given with the first introduction of transformer [4].

We use a transformer with two encoder layers (**N=2**), and the number of attention heads in each layer is fixed at 4. Besides, dropout of 50 % is performed to improve the model's generalisation ability. The output of the transformer is again temporally averaged into a vector representation, which generates class predictions through two fully-connected layers.

5.3.1.3 Cascade Hybrid Model

The attentive LSTM and transformer modules can be stacked into two sequential structures, i. e., ResNet - LSTM - Transformer (cf. Figure 5.3.2a) and ResNet - Transformer - Attentive LSTM (cf. Figure 5.3.2b). Considering that multi-head attention layers are included within the transformer architecture, we prune the first hybrid model by removing the attention layer after the LSTM, relying on the transformer's proficiency in learning features from more saliently informative audio frames on its own. Additional modifications are made to the extra modules to enable their connection to the original models.

5.3.1.4 Parallel Hybrid Model

An alternate construction is to place the attentive LSTM and the transformer side by side, labelled as ResNet - Attentive LSTM // Transformer (cf. Figure 5.3.2c). Using the CNN output as the input to the attentive LSTM and the transformer, the resulting representations from both modules are concatenated along the frequency-axis. The temporally averaged output is then mapped into classes using fully-connected layers.

5.3.2 Experiments and Evaluation

To test our proposed models, we perform experiments using the MASC database [300]. The evaluation mainly focuses on comparisons to past studies that used CNNs alone to solve the detection problem. In addition, we undertake an ablation analysis to see how much the model can be enhanced by using neural networks that are adept at learning time-dependencies. Additional research is conducted to explore the impact of several audio data augmentation techniques. Finally, we assess the efficacy of gender-specific models for the task at hand.

All presented models are optimised by minimising a cross-entropy loss using an Adam optimiser. The learning rate and batch size are set to 0.0001 and 32, respectively. Using a computation unit of a Geforce RTX 3090, we can perform the model training and inference in a time-efficient manner.

5.3.2.1 Data Description

Mask Augsburg Speech Corpus (MASC) [300] is a dateset made available for the *Mask Sub-Challenge* (MSC) presented at the Computational Paralinguistics Challenge (ComParE) at Interspeech 2020. Using this speech data, several researchers have attempted to establish automated solutions to face mask detection. The dataset contains around 10 hours of speech recordings taken from 16 female and 16 male German native speakers between the ages of 20 and 41 years (mean age 25.6 years, std. dev. 4.5 years). To collect this data, participants were instructed to conduct a number of spoken tasks, including answering questions, reading, or describing pictures, while wearing or not wearing a surgical mask from *Lohmann and Rauscher* (type Sentinex Lite). All the audio files are recorded in the single-channel format at a sampling rate of 16 kHz and then segmented into chunks of one second.

Unlike the approach of data partitioning given in the ComParE challenge, [297] conducted cross-validation on the union of the original training and development data to assure the model generalisability, achieving the current state-of-the-art results on the MASC test set. According to the ComParE guidelines, the fusion of the training and development data set is explicitly permitted and used in baseline generation. To bypass the cross-validation used in [297], we simply include a portion of the development data into the training of the model. Consequently, we train our models using the new expanded training set and observe the training curves on the new reduced development set during training. The test set remains unaltered from the original partitioning. We detail the new data partitions in Table 5.3.2 in terms of class, partition, and gender information.

Table 5.3.2: Data partitioning of the audio samples available in the *Mask Augsburg Speech Corpus* (MASC). The absolute number of instances and the sum ($\sum$) according to each class (No-mask and Mask), partition (Train, Development, Test), and speaker gender (female (f), male (m)) are indicated. Furthermore, the number of speakers for each partition is also given in parentheses. (*Source*: [304])

	Train			**Development**			**Test**			$\sum$		
	f	**m**	**f+m**	**f**	**m**	**f+m**	**f**	**m**	**f+m**	**f**	**m**	**f+m**
	(9)	**(10)**	**(19)**	**(2)**	**(1)**	**(3)**	**(5)**	**(5)**	**(10)**	**(16)**	**(16)**	**(32)**
No-mask	4689	6457	11146	599	274	873	1955	3598	5553	7243	10329	17572
Mask	4266	8221	12487	766	270	1036	1738	3721	5459	6770	12212	18982
$\sum$	8955	14678	23633	1365	544	1909	3693	7319	11012	14013	22541	36554

5.3.2.2 Data Processing

Log Mel spectrogram, denoted as $\mathbf{M}$, is extracted from each one-second audio chunk by mapping its spectrogram onto the Mel-scale frequencies, which is then converted to the logarithmic scale in magnitude in order to reflect the non-linear properties of human hearing with respect to both audio frequency and intensity. For this, we use STFT with a Hamming window of length 25 ms and step size of 3.125 ms, as well as 40 Mel filter-banks for the creation of the Mel spectrogram.

We augment the audio representations with the first- and second-order temporal differences, i. e., *deltas* ($\Delta\mathbf{M}$) & *delta-deltas* ($\Delta^2\mathbf{M}$), between successive audio frames. The transition information can approximate the first and second derivatives of the features to represent the constant flux of speech signals. Subsequently, a Z-score normalisation is performed respectively on the tri-channel feature maps. For a given Log Mel spectrogram $\mathbf{M}$,

$$\overline{\mathbf{M}} = (\mathbf{M} - \text{mean}(\mathbf{M}))/\text{std}(\mathbf{M}), \tag{9}$$

and the final tri-channel feature map as the model input:

$$\tilde{\mathbf{M}} = [\overline{\mathbf{M}}, \overline{\Delta\mathbf{M}}, \overline{\Delta^2\mathbf{M}}]. \tag{10}$$

We also explore SpecAugment [121], an additional data augmentation method commonly used for speech processing. The approach improves the generalisability of a model by training it to resist deformations caused by the partial loss of temporal and frequency information.

5.3.2.3 Evaluation Metrics

To compare with previous works, we choose the assessment criterion, *Unweighted Average Recall* (UAR), the same as in ComParE challenge, for fair comparisons. To conduct the ablation study and make more thorough comparisons among our models presented, we take into account other performance measures including *Unweighted Average Precision* (UAP), *Unweighted Average F1* (UF1), and *Matthews Correlation Coefficients* (MCC) [307].

Table 5.3.3: Results in Unweighted Average Recall (UAR) [%]. **w/ PE** and **wo PE** stand for with and without positional encoding, separately. (*Source*: [304])

Results in the literature	[%]
ComParE baseline[300]	71.8
Four acoustic features[298]	73.5
LLA [299]	69.1
Mask Filer [308]	70.7
DA on Spectrograms[301]	71.5
Cycle-consistent GANs[303]	74.6
PANNs+Mixup[302]	76.2
Image classifiers [297]	80.1
Proposed Models	
CNN	75.3
ConvLSTM	77.4
Attentive ConvLSTM	78.6
Convolutional Transformer (wo PE)	76.6
Convolutional Transformer (w/ PE)	**79.3**
Proposed Hybrid Models	
ResNet - LSTM - Transformer	78.9
ResNet - Transformer - Attentive LSTM	77.7
ResNet - Attentive LSTM // Transformer	79.0

5.3.2.4 Comparisons to Previous Work

The attentive convLSTM and convolutional transformer can obtain UARs of 78.6 % and 79.3 % for mask-wearing detection (cf. Table 5.3.3), indicating substantial gains over the ComParE challenge baseline [300]. Both models outperform the most previously published results with only the exception of [297], which combines the prediction results of 4 distinct models, similar to [300, 298, 299]. These ensemble approaches benefit from the versatility of multiple models and numerous feature sets. Our testing results, on the other hand, are dependent on a single model and typical spectrogram-based features. In fact, in terms of UAR, our two approaches outperform all previous single model approaches in the literature [308, 301, 303, 302]

Table 5.3.4: Testing results of each of our models regarding Unweighted Average Precision (UAP) [%], Unweighted Average F1 measure (UF1) [%], and Matthews Correlation Coefficient (MCC). The number of the model parameters is also provided on the scale of (M)illions. (*Source*: [304])

Models	**UAP** [%]	**UF1** [%]	**MCC**
CNN	75.4	75.3	.507
ConvLSTM	77.4	77.4	.549
Attentive ConvLSTM	78.3	78.3	.565
Convolutional Transformer (wo PE)	76.7	76.5	.532
Convolutional Transformer (w/ PE)	**79.3**	**79.3**	**.586**
Hybrid Models			
ResNet - LSTM - Transformer	78.9	78.9	.578
ResNet - Transformer - Attentive LSTM	77.7	77.7	.554
ResNet - Attentive LSTM // Transformer	79.1	79.0	.581

5.3.2.5 Ablation Study

Ablation tests are conducted to analyse the importance of learning time-dependencies of speech signals for mask-wearing detection. To do this, we compare our models employing the LSTM network or the transformer on top of the CNN with a basic CNN model without modules that capture temporal dynamics. In addition, the impact of positional encoding for the convTx model is also evaluated prior to evaluating the three hybrid models.

Both the presented models exhibit considerable detection improvements over a simple CNN model, which achieves a detection UAR of 75.3 % (cf. Table 5.3.3), according to a one-tailed z-test at significance level $\alpha = 0.05$. By layering an LSTM network on top of the CNN, the detection performance can be improved by 2.1 % in terms of UAR, while the inclusion of a multi-head attention layer yiels an additional 1.2 % performance boost. Other attention mechanisms, such as soft attention [28], local attention [30], and component attention [309, 310] are also investigated; however, they do not produce better detection results than using multi-head attention with 4 heads.

The best performing model, the convolutional transformer, annotates the temporal positions of its input sequence using positional encoding. Without the use of positional encoder in this convolutional transformer, denoted as Convolutional Transformer (wo PE) in Table 5.3.3, results in the loss of the time-dependencies information. Our experimental result for the convolutional transformer without PE falls to 76.6 %, which is inferior to the performance of ConvLSTM that applies CNN and LSTM only. These findings imply that capturing temporal dependencies in speech can promote the identification of whether a speaker is wearing a mask or not.

Regarding the hybrid models, the first hybrid model that stacks a transformer on top of it can slightly improve the detection performance compared to the original attentive convLSTM. The second method employing the attentive LSTM to advance the convTx, however,

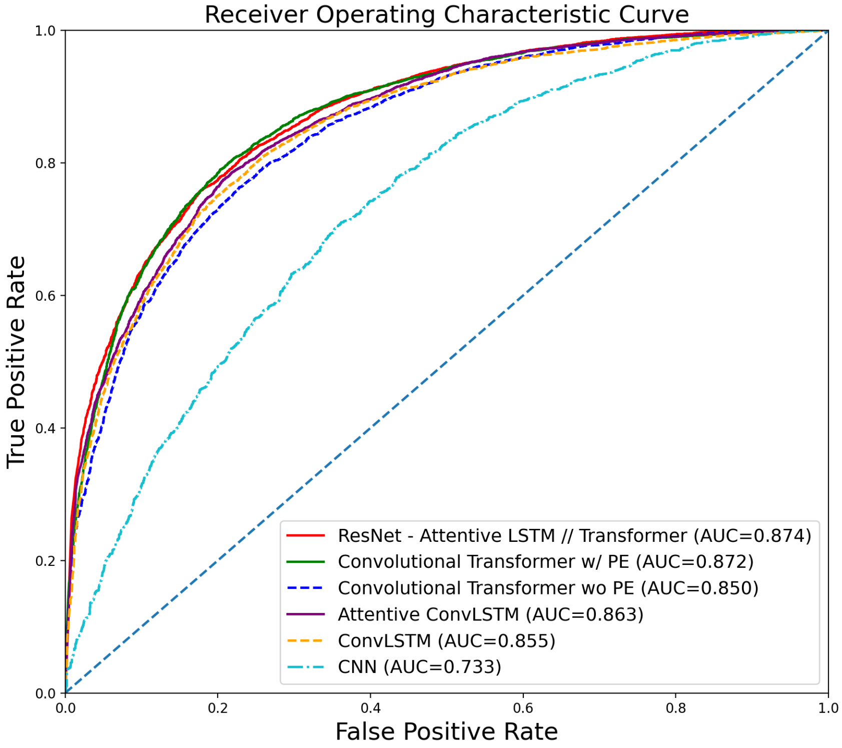


Fig. 5.3.3: Comparison of the ROC curve of our different models, and the corresponding Area Under Curve (AUC).

results in a performance decrease of 1.6 %. The parallel alignment of the attentive LSTM and transformer reaches a UAR of 79.0 %, which is though comparable to the performance of convTx but surpasses the other two hybrid models. The performance of these three hybrid models can be improved by augmenting the input feature using SpecAugment, as detailed in Section 5.3.2.6. As shown in Figure 5.3.3, the *Receiver Operating Characteristic* (ROC) curve [311] are depicted based on the prediction results for all the models presented. The hybrid model, ResNet - attentive LSTM // Transformer, outperforms other techniques with an area under the ROC curve (AUC-ROC) of .874. Overall, a the balance between the ratio of true positives and false positives is reserved for all our proposed models, allowing for accurate mask-wearing detection with a low number of false alarms. This can also be supported by the testing results shown in Table 5.3.4, where the convolutional transformer achieves the highest performance for all the metrics evaluated, obtaining a UAP of 79.3 % and a UF1 of 79.3 %, and a MCC of .586.

5.3.2.6 The Influence of Data Augmentation

The Effect of Deltas & Delta-Deltas

To validate the efficacy of using deltas and delta-deltas of the Mel spectrogram to enrich the model inputs, we separately feed the Mel spectrogram ($\mathbf{M}$), along with the first-order and second-order differences ($\Delta\mathbf{M}$, $\Delta^2\mathbf{M}$) to three tested models, i. e., the CNN, the attentive convLSTM, and the convolutional transformer (cf. Table 5.3.5). Using these transitions as the input to the CNN increases the detection performance by 0.9 % in terms of UAR. The

Table 5.3.5: The testing results in UAR [%], based on models using different set of features. (*Source*: [304])

Models	M	M+ΔM	M+ΔM+Δ^2M
CNN	74.4	75.0	75.3
Attentive ConvLSTM	76.2	78.0	78.6
Convolutional Transformer (w/ PE)	77.5	78.2	**79.3**

Table 5.3.6: Testing results, obtained by applying SpecAugment, for the Attentive ConvLSTM, the Convolutional Transformer, and the three hybrid models. The evaluation measures are Unweighted Average Recall (UAR) [%], Unweighted Average Precision (UAP) [%], Unweighted Average F1 measure (UF1) [%], and Matthews Correlation Coefficient (MCC).SA abbreviates SpecAugment. (*Source*: [304])

Models	SA	UAR [%]	UAP [%]	UF1 [%]	MCC
Attentive ConvLSTM	–	78.6	78.3	78.3	.565
	✔	**79.5**	**79.6**	**79.5**	**.592**
Convolutional Transformer (w/ PE)	–	79.3	79.3	79.3	.586
	✔	**80.8**	**81.0**	**80.8**	**.619**
Hybrid Models					
ResNet-LSTM-Transformer	–	78.9	78.9	78.8	.578
	✔	**82.2**	**82.2**	**82.1**	**.643**
ResNet-Transformer-Attentive LSTM	–	77.7	77.7	77.7	.554
	✔	**79.9**	**79.9**	**79.9**	**.598**
ResNet-Attentive LSTM // Transformer	–	79.0	79.1	79.0	.581
	✔	**81.1**	**81.1**	**81.1**	**.622**

performance of the other two models is improved by 2.4 %, and from 1.8 %, respectively, owing to the features augmented by these transitions. Therefore, in order to attain the optimal detection result, it is prudent to investigate feature augmentation techniques and neural networks that are proficient in collecting time-dependencies across time steps.

The Effect of SpecAugment

We further study the impact of SpecAugment and provide the testing results in Table 5.3.6. Applying SpecAugment to the input Log Mel Spectrogram improves the effectiveness of convTx, obtaining a UAR of 80.8 %, a UAP and UF1 of 81.0 %, and an MCC of 0.619, which surpasses all models presented in the literature (cf. Table 5.3.3). The SpecAugment is particularly efficient in enhancing the three hybrid models. As a result, the first hybrid model, ResNet - LSTM - Transformer, provides a new state-of-the-art result, reaching a UAR and UAP of 82.2 %, a UF1 of 82.1 %, and a MCC of .643.

Table 5.3.7: Results in UAR [%] for the performance of training and testing our best model, ResNet - LSTM - Transformer, on different genders, (f)emale & (m)ale. Overall UAR indicates the combination of the results for both genders. Gender UAR measures the gender classification performance from the multi-task learning. (*Source*: [304])

Single task

Training Set	Test Set	UAR [%]	overall UAR [%]
f + m	**f**	77.6	82.2
f + m	**m**	83.0	
f	**f**	78.6	82.5
m	**m**	84.1	
f	**m**	68.4	–
m	**f**	68.7	

Multi-task

Training Set	Test Set	gender UAR [%]	UAR [%]
f + m	**f + m**	98.6	81.5
f + m	**f**	–	79.0
f + m	**m**	–	82.6

5.3.2.7 Gender-specific Models

Male and female voices may be affected differently by face masks. We therefore further analyse the gender dependence of our best-performing model, i. e., ResNet - LSTM - Transformer, (cf. Table 5.3.7). The detection result obtains a UAR of 77.6% and 83.0 % for the female and male speakers in the MASC test set, respectively. There are fewer female speaker accessible than male speakers in the training set, which contributes in part to the performance discrepancy between the two genders. Hence, designing gender-dependent models might result in greater performance. Individually trained convTx models for the female and male speakers may achieve UARs of 78.6 % and 84.1 %, respectively, yielding an overall UAR of 82.5 %. This suggests that it is advantageous to condition our model on gender information for improved face mask detection. In particular, we can frame our task as a multi-task learning that concurrently recognises the gender of the speaker and determines whether or not the speaker is wearing a mask. To this end, an extra output is introduced to the model architecture given in Figure 5.3.2a, used to predict the speaker gender from the learnt audio representation. The cross-entropy loss for gender prediction is added to the mask detection loss, and the overall loss is reduced to optimise both predictions. As a consequence, a UAR of 98.6 % can be obtained for gender classification. The identification of face masks reaches a UAR of 79.0 % and 82.6 % for the female and male speakers, respectively, minimising gender-related biases between speakers of the two genders.

Regarding the gender of the speakers, we should not infer that the proposed models perform better for male speaker than for female speakers due to the imbalanced data distribution for the two genders in both the training and test sets. The gender inequality of ML systems caused by biases in the data is a well-known issue [312]. Hence, it is highly suggested to collect more data while attempting to balance the data across genders in order to create a model with improved detection performance that is free of gender bias. However, our work reveals that that the use of a gender-dependent model is a plausible way to reduce the gender bias in the data. This method, which further boosts the performance by conditioning the structure on known gender information, is in line with gender de-biasing strategies studied in other disciplines [313]. We hypothesise that a front-end gender classification model may outperform the results obtained by our multi-task approach. This is inspired by the observation that a gender classification model already exists in other speech-related tasks [314], such as emotion recognition [315]; hence, similar techniques should be examined in the context of mask-wearing detection. As the contrast between female or male speech is more distinct than that between a speaker wearing a mask or not, we are optimistic about the prospective outcomes.

5.3.2.8 Additional Efforts

Recent studies have advocated for the exploitation of Self-Supervised Learning (SSL) for audio and speech processing [48]. Considering that our mask-wearing detection task is dependent on speech signal, we attempted to use several SSL models, including Wav2Vec 2.0 [173], HuBERT [174], XLSR [175], and BYOL for audio [316], that have gained great success in speech-related applications. However, the fine-tuning of the big SSL models, such as Wav2Vec 2.0 and HuBERT, is very time-consuming, and shown to be inferior to our proposed solutions. A potential reason is the mismatch between the languages used for training, i. e., Wav2Vec 2.0 is trained using an English corpus, whereas the MASC dataset is in German. XLSR, which is built on Wav2Vec 2.0, learns cross-lingual speech representations, including German. However, we are unable to see gains by fixing the language mismatch problem. Based on these experiments, we believe that a model with great performance for speech recognition may not be as suitable for our purpose, compared to the models that are specifically designed. In addition, the primary objective of ASR is to understand the contextual meaning of the input speech, and hence, an ASR model may suppress the tempo and rhythm changes caused by masks. BYOL-A, unlike the other evaluated SSL models, does not apply contrastive loss during its training and has the potential to learn more complete information from audio signals. Regardless of whether SpecAugment was applied, fine-tuning the BYOL- A model yields experimental results that are marginally inferior to those of our proposed method. As the research on audio SSL is still in its infancy, we are enthusiastic about its efficacy in detecting the use of a mask detection task in future works.

5.4 Summary

In this chapter, we presented several speech-based solutions to the mask-wearing task. The success of these solutions can be attributed to their ability to capture the temporal dynamics of speech signals. In the absence of visual information, these approaches may serve as alternatives to those utilising CV techniques. In spite of the encouraging findings given in this study, several restrictions should be addressed in the future before our models can be applicable to real-life circumstances. First, our methods must be tested for other mask types, such as FFP-2, despite the fact that surgical masks cause relatively less alterations to human voice [289], making the detection task easier. Moreover, cross-lingual tests should be conducted to examine the generalisation effect of the models to other languages, concerning the substantial variation of paralinguistics across different languages [317]. Moreover, the individuals that contributed to the MASC data collection wore their masks correctly. In reality, however, this is not always the case, since people do not always wear a mask properly, such as covering their mouths only [318, 319]. Hence, future models thus incorporate the identification of incorrect mask-wearing. At last, the proposed models may encounter real-world noise; thus, audio enhancement techniques, including those presented in previous chapters of this dissertation, should be applied to improve the robustness of a mask-wearing detection model.

CHAPTER 6

Discussion

This dissertation focus on the use of cutting-edge deep learning techniques to solve pressing concerns in speech and digital health. In the course of our study, we have extended the algorithms to real-world settings in order to satisfy several real-world application criteria, including robustness and generalisabiity. Due to the constraints of problem modelling (we cannot exhaust all possibilities in actual applications), and deep learning's reliance on data characteristics, these approaches still have certain limitations in practical applications.

For audio enhancement, we have shown N-HANS, developed based on our proposed ±Auxiliary Network, can serve several audio enhancement purposes. The approach is adaptable to diverse environments and audio sources, including speakers and backgrounds that were not seen during training. Nonetheless, this toolkit takes into account the variety of speakers and environmental contexts. In practical applications is the distance between speakers and recording equipments during data recording can arise a formidable obstacle, as it not only results in low SNR circumstances but also causes audio reverberation. In order to facilitate the system's adaptability, we compromise its performance in low SNR settings. Therefore, future research should concentrate on boosting speech intelligibility under conditions of very low SNRs to compensate for the distortions that are occasionally present in audio. Due to the limitations in creating our training data, the source separation model is primarily designed to separate overlapping speech from two speakers. Further efforts are needed to extend the system to any number of audio sources, including music source separation.

The joint optimisation methods are intended for adjusting an audio enhancement model for multiple computer audition applications, particularly when these applications confront the extreme low SNR circumstances. Collectively, in five distinct application areas, i. e., ASR, SCR, SER, ASC and COVID-19 detection, our findings demonstrate that the suggested methods outperform comparable baselines. In most cases, it is possible to recover a substantial percentage of the performance loss caused by noise. This highlights the necessity of specialised AE systems that can differentiate between the task-specific relevancy of audio signals and noise sources. However, there are some limitations associated with this study, since the performance of some baseline models is inferior to that of recent state-of-the-art methods when simply trained and tested on clean data. For instance, an ASR model may benefit from self-supervised learning [130], which allows them to scale up the amount of data and gain the associated advantages. Using joint SSL and enhancement pre-training on larger size of data, followed by fine-tuning with our iterative optimisation on the target downstream task is a promising future research direction.

The experiments in Fitbeat study were intended to determine whether COVID-19 symptoms emerged within a period of recorded heart rate data, the models have limitations in a causative setting, i.e., when attempting to anticipate possible symptoms before they occur. To this purpose, future research will seek to address the issue of how many days in advance it is feasible to accurately and reliably forecast the beginning of COVID-19 symptoms.

Our suggested COVID-19 detection algorithm is expected to benefit with a larger training dataset. Additionally, other time frames of data segments should be analysed to further decrease the needs for model input. Overall, we are optimistic that an accurate identification of the presence of COVID-19 can be reached based on the symptoms outlined in this study and machine learning analysis of heart rate measurements. From the perspective of algorithm, the effectiveness of contrastive CAE provides a foundation for future study. As a general binary classification approach, it should be extended for a widespread adoption, particularly for predicting diseases outside COVID-19. In contrast to typical unsupervised learning methods for anomaly detection using auto-encoders, our method frames the task as a supervised learning approach by supplying a training objective analogous to the that of anomaly detection during the model optimisation. The goal-oriented optimisation should not limited to the task of COVID-19 detection alone. Since our proposed method introduces a new parameter indicating margin size, the challenge of transferring our method to other applications may reside in determining the optimal margin size. In addition, the proposed model requires a sufficient quantity of data for pre-training, which hinders its use, for instance to the diagnosis of rare diseases. In the near future, our suggested contrastive CAE will be expanded to multi-class paradigms to accommodate a broader range of applications.

In the research of using deep learning techniques to detect face masks from speech, our experimental findings reveal the importance of considering the time-dependencies in audio. Thus, the neural networks that can account for the positional information facilitate the modelling of the time-dynamics. Our method overcomes the difficulty of seeking for proper features, which is typically a time-consuming procedure that normally needs specialised expertise, and substantial manual effort. In general, some biases can be unintentionally introduced based on the design parameters during model construction. The lack of adequate representative training data may lead to a decrease in overall accuracy or biased results, when a deep learning model reinforces patterns in non-representative data set. Due to the acquisition of data, this issue emerges in our investigation of mask recognition from speech. As a consequence of having more male speakers in training set, testing results for male speakers are superior. Indeed, we also built gender-dependent models in an attempt to narrow the performance gap between both genders, it may be advantageous to incorporate more personalisation factors, such as age, location, etc, to improve the overall recognition performance. A more sophisticated method of including general and intricate speaker information has been proposed in the model design of N-HANS. Intuitively, we anticipate future research on personalised models to increase in order to meet practical needs, particularly those deployed by edge devices.

Besides the encouraging results, it is crucial to emphasise that future research should concentrate on several aspects that were not assessed in our experiments. First, we conducted the experiments only using surgical masks. In comparison with other mask types, such as FFP-2, surgical masks induce relatively less alterations to the human voice [289], resulting in more difficulty in differentiating speakers wearing masks or not. More study should be carried out with other kinds of mask in order to examine the generalisability of the presented approach. Second, all of the findings have been obtained from German-speaking participants. Since para-linguistics may vary substantially across languages [317], and these may be affected differently by the effect of wearing a mask, more cross-lingual studies are necessary to assess the generalisability of the proposed approaches to other languages. Third, participants were instructed to wear their masks appropriately during data collection of MASC (with proper mask size and fit). In daily life, however, some individuals may wear masks improperly, such as by covering their mouth only but not nose [318, 319].

In light of the actual application of automatic wearing-mask detection that can include the identification of irregular mask usages, it is necessary to carry out more research on mask positioning. We believe the detection will be more precise after considering these cases.

The importance of data ethics and AI ethics is concerned throughout the whole dissertation. Except for the Fitbeat project, which uses participant-consented data for research purposes, we rely on publicly available data for the rest of our work. Particular ethical care is given to the medical applications. Our objective for the task of COVID-19 detection should not only be to obtain a high level of detection accuracy, but also to minimise the risk of false alarms and miss detection from the designed models. Reducing the likelihood of a missed detection might lessen the possibility of ignoring COVID-19 positive patients, which could lead to a delay in treatment and isolation; nevertheless, false alarms can cause unwarranted stress and anxiety. When building the COVID-19 models based on heart rate data and speech data, we prioritise the former concern due to the hazards involved involved in the two cases. For the model processing coughing and breathing sounds, a good balance between the two concerns is retained. Besides, our COVID-19 detection solutions should not be seen as a method of clinical diagnosis; rather they are indicated as means of assisting in the monitoring of potential COVID-19 cases. For a more exact model for practical application, personalisation, or case adaptation in general, should be considered as a following effort to better fit the encountering context. However, model personalisation or customisation may be intrusive to user privacy, and a reasonable trade-off between privacy protection and model efficacy should be taken into account for production.

CHAPTER 7

Conclusion and Future Work

From the perspective of applications, this thesis focuses mostly on 1) the employment of AI techniques to overcome certain remaining challenges in the process of audio enhancement, and 2) the potential contribution of deep learning approaches to COVID-19 related problems. Our study focuses on discovering versatile neural network frameworks and generic training paradigms across these two research domains.

To address our first research question (**Q1**), we presented a model assisted with auxiliary networks that yields a solution to integrate multiple audio enhancement functionalities into a single framework. The same structure is also effective for data fusion, as demonstrated by the method for detecting COVID-19 utilising data of two audio types. The joint optimisation strategies, including our suggested iterative training solution, were proposed as a response to second research question (**Q2**), which aimed to train an audio enhancement model reliant on its subsequent audio applications in order to maximise the application performance in noisy environments. Such audio applications include COVID-19 detection based on speech, which confirmed in part the feasibility of exploiting speech enhancement technology to improve the effectiveness of COVID-19 detection per speech in real-world circumstances (**Q3**).

Additionally, we expanded our search for COVID-19 detection methods to heart rate data to investigate a deep learning alternative, thereby addressing the third research question (**Q3**). Concerning the typical issue of class imbalance in the data for this task , we formulate the binary classification problem into an anomaly detection problem, and proposed a contrastive CAE algorithm. These modelling methodologies and deep learning approaches should not be restricted to the tasks presented in this paper, but should also be instructive for other tasks with a similar learning objective. As a response to the fourth research question (**Q4**), we provide several basic and hybrid deep learning models for the efficient recognition of face masks from speech, therefore establishing the foundation for monitoring the public usage of masks. In the section of Discussion, we analysed the merits and drawbacks of our proposed methodologies, as well as their proximity to the actual implementation.

Inspired by the outcomes of the joint optimisation approach, more attention should be give to the study of combining neural network modules. Though each module of the sequential chain has a specific purpose, these modules not only benefit from adapting to one another to retain their functional connection, but a synergistic effect can be achieved. This enables end-to-end learning of multiple neural networks as a substitute for the error-accumulating cold cascade of neural networks. Moreover, unlike the cold-cascade combination, each module of the entire sequential framework contributes more fairly to the overall performance, i. e., although each module has its own primary functionality, it also assists other modules on the chain. Due to the expansion of the model structure and the interactions between these modules, the constraints on training individual module may be relaxed, yet the system as a whole has the opportunity to shoot a higher score. In this regard, further study should first consider these two factors: the suitable connection of these modules in terms of, for instance, data format and transformation, and the alignment of modules that may provide unique interaction effects. Given the overwhelming success of AI technology in the present day, the connection of AI components will play a crucial role in bringing AI solutions to satisfy real-world requirements. This may eventually lead to significant advancements to several systems, including audio systems.

In addition to their great performance in terms of evaluation metrics such as detection or classification accuracy, the models developed in this paper account for a number of practical application requirements, such as storage efficiency and computational complexity, inference speed and reliability, etc. In fact, we place a premium on the robustness of these models when applying them to real-world circumstances. First of all, we exploit large-scale data to assure the generalisability of our enhancement models, so that they can manage varied recording conditions, including difference in speakers and noise sources, recording equipment and so on. Taking into account all these distinctions, the enhancement process can format the audio style for future processing, so lowering the requirements on the following model's robustness against noise disturbance. We assess the efficacy of this technique using several audio applications, including ASR, SER, SCR, ASC and COVID-19 detection, A similar method should also be applicable to other applications and signals, such as the identification of COVID-19 based on heart rate measurements. Due to daily activities, heart rate measurements, particularly those available from wearable devices, may contain perturbations more than just additive noise. Using front-end enhancement to improve the signal quality is a viable solution for this issue. For example, we may use the N-HANS model, which learns from extra samples indicating the desired and unwanted components, to help acquire clean heart rate data. In this kind of framework, we also made attempts to strengthen the connection between the enhancement module and the model for subsequent audio application in order to further increase the system's robustness. Specifically, using joint optimisation allows the training of the two modules to mutually benefit, and as the training anticipates their cascade, it eliminates the difficulty that would be encountered if they are trained separately.

We evaluate speech, breathing and cough sounds, as well as heart rate data in the search for feasible deep learning methods for COVID-19 diagnosis. Currently, the diagnostic accuracy of these solutions falls behind that of other clinical methods, such as Polymerase Chain Reaction (PCR) or rapid tests; however, they rely on more efficient ways to collect data without interfering with the participants' daily activities, and can therefore be used as a supplement to alert patients for earlier diagnosis. We anticipate that, despite the time required for the technology to mature, such research will have a significant commercial impact on the general public. As of the writing date, AI approaches are thriving to this study field. However, the majority of research are dependent on exploiting fundamental machine learning methods, or applying transfer learning. Unlike them, we proposed a more appropriate modelling technique in terms of neural network design and training objective, which were tailored for the task of COVID-19 detection, resulting in improved illness diagnosis over the conventional methods. In order to further the clinical use of these technologies, we advocate for additional study on the investigation of deep learning algorithms.

Nevertheless, there are still some remaining problems that need to be resolved in the future. First of all, although the N-HANS toolkit integrates numerous audio enhancement functionalities via auxiliary networks, the enhancement performance can significantly degrade under certain challenging conditions, such as when the noise or interference is so loud that it overwhelms the audio of interest. Similarly, when the recording device is placed far from the sound source, two issues, i. e., low SNR and the far field effect, are frequently encountered during audio recording, which can impede a number of research. Such kind of disturbance can even obliterate the presence of the audio of interest, rendering it inaudible to human hearing and scarcely possible to extract using machine learning technology. Even worse, distant recordings may be affected by reverberation, a sort of multiplicative noise that prolongs the audio of interest with attenuated sound tails in the same structure. De-reverberation, the countermeasure against reverberation, has been considered as a separate task from noise reduction. Recent research attempts to solve both audio enhancement and de-reverberation using a single model, shedding light on the integration of more audio enhancement functionalities into the tool.

Despite increasing efforts to the development of COVID-19 detection models, most current research only targets at the detection accuracy. This is owing to the strong hopes for finding an early COVID-19 surveillance solution. In spite of this, it has to be tested whether the established methods are reliable and robust when being applied in actual world. Future work should focus more on model interpretability to reveal the role of AI for the task at hand. Concerning the task of mask-wearing detection, in order to better fulfil the requirements of its practical applications, the presented methods should be extended in two ways: their generalisation to other types of face mask, such as FFP-2 and fabric masks, and to more conceivable mask-wearing positions. Additionally, to expand the applicability of AI technology, one should continue research on methodologies that promote both the the front-end processing and its intended application, as opposed to treating them as two independent parts. Its benefits urge us to apply deep learning algorithms to a broader range of practical applications.

Nomenclature

x_i	Feature map of the ith heart rate segment
x_i^p	Feature map of the ith positive (symptomatic) heart rate segment
x_i^n	Feature map of the ith negative (asymptomatic) heart rate segment
$\hat{x}_i$	Reconstructed feature map of the ith heart rate segment
$\hat{x}_i^p$	Reconstructed feature map of the ith positive (symptomatic) heart rate segment
$\hat{x}_i^n$	Reconstructed feature map of the ith negative (asymptomatic) heart rate segment
$f^{\mathbf{enc}}(\cdot)$	Encoder processing
$f^{\mathbf{dec}}(\cdot)$	Decoder processing
m	Margin value
N	Number of samples
conv	Convolutional layer
deconv	Transposed convolutional layer
pool	Max-pooling layer
depool	Transposed max-pooling layer
flatten	Flatten layer
fc	Fully-connected layer
avg. pool	Average pooling operation
PReLU	Parametric rectified linear unit
ReLU	Rectified linear unit
CNN	Convolutional neural network
CAE	Convolutional auto-encoder

MLP	Multi-layer perception
LSTM	Long short-term memory
LOSO	Leave-one-subject-out
CV	Cross validation
UAR	Unweighted average recall
RMSE	Root mean square error
AUC	Area under the (receiver operating characteristics) curve
CD1/2	First/Second case definition
ICS	Internal covariate shift
Attr.	Latent attributes
std	Standard deviation

LIST OF TABLES

To Myself

During the second round of reviewing the manuscript, I began to recall the past four years in Augsburg and extended to the past ten years living and studying in Germany. The growth in these days not only helped me gain a deeper understanding in the professional fields I love, such as artificial intelligence, speech processing, and human health, but also helped me internalise many contradictions, such as misunderstandings and grievances. Such growth has allowed me to downplay the trivialities in my life, allowing me to focus on the realisation of my dreams, goals and values. Needless to say, during my Ph.D. research, I had a difficult time and almost gave up my career. For this, I made more efforts and attempts which brought more setbacks and blows. However, in this case, you can't let your failures define you, you have to let the failure teach you. Your failures are telling you that you need to do differently next time, and sometimes they are hinting you what you should do.

My superstar, Kobe Bryant once said "rest at the end, not in the middle“. I have made my study and work for the past ten years very compact. In January 2020, a morning that I was about to wake up, I was told by my family and friends that Kobe passed away in helicopter crash. I didn't subconsciously cry because physically he was always so far away from me, and he might have gone to play basketball with God this time. Today, I often feel regret and sad that I have never met him in person, since he left the precious treasure of life in my heart, Mamba Mentality. The four years of my Ph.D. have been arduous but super worthwhile, a lot of growth comes along the time. It provides me a good start for considering some questions I am facing in my life: When to keep persistent and when to give up, how to approach success and how to accept failure, how to work alone and when to ask for help.

I often reminisce about two friends from my youth. Whenever I am flustered and hesitant, I will think of one of them. He is very good at calming himself in the face of difficulties and never seems to be flustered. From time to time, he makes jokes of himself in difficult situations. He became an excellent surgeon after graduating from a famous medical school. The other friend works as a news reporter and interpreter, working across several countries. When I was only six or seven years old, for the first time in my life, he made the point that one should do things with attention to the sense of beauty. The pursuit of beauty brings your patience, curiosity and respect to your work. It can make what you are doing more passionate.

I should be more grateful for the tolerance and patience of my family, supervisor, mentors, friends, colleagues, and many familiar and unfamiliar around me. Those tolerances have given me more freedom to be willful, to think, and to adjust the trajectory of my career. Those patience carry their expectations of me, and allows me the time to think and explain. It is very appreciated that the world treats me kindly, and I wish everyone deserves all the kindness in the world.

Don't forget those work habits
Don't forget what has gotten you to where you are.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. CVPR*, (Las Vegas, NV, USA), pp. 770–778, 2016.

[3] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, p. 32 pages, 1997.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, (Long Beach, CA, USA), 2017. 11 pages.

[5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *Proc. ICLR*, 2021.

[6] S. Indolia, A. Goswami, S. Mishra, and P. Asopa, “Conceptual understanding of convolutional neural network – A deep learning approach,” *Procedia Comput. Sci.*, vol. 132, pp. 679–688, 2018.

[7] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in *Proceedings of the European Conference on Computer Vision*, (Zurich, Switzerland), pp. 818–833, 2014.

[8] K. Simonyan, A. Vedaldi, and A. Zisserman, “Deep inside convolutional networks: Visualising image classification models and saliency maps,” in *Proceedings of the International Conference on Learning Representations*, (Banff, Canada), 2014. 8 pages.

[9] F. Zhao, H. Li, and X. Zhang, “A robust text-independent speaker verification method based on speech separation and deep speaker,” in *Proc. ICASSP*, (Brighton, UK), pp. 6101–6105, 2019.

[10] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, “Large-scale video classification with convolutional neural networks,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, (Columbus, OH), pp. 1725–1732, 2014.

[11] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, “Deep learning for time series classification: A review,” *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[12] K. Yasaka, H. Akai, O. Abe, and S. Kiryu, "Deep learning with convolutional neural network for differentiation of liver masses at dynamic contrast-enhanced CT: A preliminary study," *Radiology*, vol. 286, no. 3, pp. 887–896, 2017.

[13] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, R. Kim, R. Raman, P. C. Nelson, J. L. Mega, and D. R. Webster, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *Journal of the American Medical Association*, vol. 316, no. 22, pp. 2402–2410, 2016.

[14] B. Ehteshami Bejnordi, M. Veta, P. Johannes van Diest, B. van Ginneken, N. Karssemeijer, G. Litjens, J. A. W. M. van der Laak, and the CAMELYON16 Consortium, "Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer," *Journal of the American Medical Association*, vol. 318, no. 22, pp. 2199–2210, 2017.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proceedings of the International Conference on Medical Image Computing & Computer Assisted Intervention*, (Munich, Germany), pp. 234–241, 2015.

[16] Z. Hu, J. Tang, Z. Wang, K. Zhang, L. Zhang, and Q. Sun, "Deep learning for image-based cancer detection and diagnosis: A survey," *Pattern Recognit.*, vol. 83, pp. 134 – 149, 2018.

[17] F. Shi, J. Wang, J. Shi, Z. Wu, Q. Wang, Z. Tang, K. He, Y. Shi, and D. Shen, "Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for COVID-19," *IEEE Reviews in Biomedical Engineering*, p. 13 pages, 2020.

[18] A. Oulefki, S. Agaian, T. Trongtirakul, and A. K. Laouar, "Automatic COVID-19 lung infected region segmentation and measurement using CT-scans images," *Pattern Recognit.*, p. 13 pages, 2020.

[19] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NeurIPS*, vol. 25, 2012.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[22] J. Bjorck, C. P. Gomes, and B. Selman, "Understanding batch normalization," in *Proceedings of the Conference on Neural Information Processing Systems*, (Montreal, Canada), 2018. 12 pages.

[23] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets," in *Proc. NeurIPS*, (Montreal, Canada), pp. 6389–6399, 2018.

[24] H. K. Vydana and A. K. Vuppala, "Residual neural networks for speech recognition," in *Proc. EUSIPCO*, (Kos island, Greece), pp. 543–547, 2017.

[25] H. Jung, M.-K. Choi, J. Jung, J.-H. Lee, S. Kwon, and W. Young Jung, "Resnet-based vehicle classification and localization in traffic surveillance systems," in *Proc. CVPR*, (Honolulu, HI, USA), pp. 61–67, 2017.

[26] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions," *Journal of big Data*, vol. 8, no. 1, pp. 1–74, 2021.

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NeurIPS*, (Long Beach, CA), pp. 5998–6008, 2017.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. International Conference on Learning Representations (ICLR)*, (San Diego, CA, USA), 2015. 15 pages.

[29] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," in *Proc. EMNLP*, (Lisbon, Portugal), pp. 379–389, 2015.

[30] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Empirical Methods in Natural Language Processing (EMNLP)*, (Lisbon, Portugal), pp. 1412–1421, 2015.

[31] G. Dong, G. Liao, H. Liu, and G. Kuang, "A review of the autoencoder and its variants: A comparative perspective from target recognition in synthetic-aperture radar images," *IEEE Geoscience and Remote Sensing Magazine*, vol. 6, no. 3, pp. 44–68, 2018.

[32] L. Meng, S. Ding, and Y. Xue, "Research on denoising sparse autoencoder," *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 5, pp. 1719–1729, 2017.

[33] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[34] S. Liu, G. Keren, E. Parada-Cabaleiro, and B. Schuller, "N-HANS: A neural network-based toolkit for in-the-wild audio enhancement," *Multimedia Tools and Applications*, vol. 80, pp. 28365–28389, 2021.

[35] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, (Munich, Germany), pp. 234–241, 2015.

[36] R. Agarwal, A. A. Sekh, K. Agarwal, and D. K. Prasad, "Auxiliary network: Scalable and agile online learning for dynamic system with inconsistently available inputs," 2020.

[37] A. Triantafyllopoulos, S. Liu, and B. W. Schuller, "Deep speaker conditioning for speech emotion recognition," in *Proc. ICME*, pp. 1–6, 2021.

[38] H.-S. Choi, J.-H. Kim, J. Huh, A. Kim, J.-W. Ha, and K. Lee, "Phase-aware speech enhancement with deep complex u-net," in *Proc. ICLR*, (Vancouver, Canada), 2018. 20 pages.

[39] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Proceedings of the Conference on Neural Information Processing Systems*, (Vancouver, Canada), 2020. 13 pages.

[40] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proc. CVPR*, (Boston, MA, USA), pp. 815–823, 2015.

[41] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. CVPR*, vol. 1, (San Diego, CA, USA), pp. 539–546, 2005.

[42] K. Sohn, "Improved deep metric learning with multi-class N-pair loss objective," in *Proc. NeurIPS*, (Barcelona, Spain), 2016. 9 pages.

[43] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proc. AISTATS*, (Sardinia, Italy), pp. 297–304, 2010.

[44] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv:1807.03748*, 2018.

[45] N. Frosst, N. Papernot, and G. Hinton, "Analyzing and improving representations with the soft nearest neighbor loss," in *Proc. ICML*, pp. 2012–2020, 2019.

[46] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, 2021. 20 pages.

[47] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 4037–4058, 2020.

[48] S. Liu, A. Mallol-Ragolta, E. Parada-Cabeleiro, K. Qian, X. Jing, A. Kathan, B. Hu, and B. W. Schuller, "Audio self-supervised learning: A survey," *arXiv*, 2022.

[49] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained models for natural language processing: A survey," *Science China Technological Sciences*, vol. 63, pp. 1–26, 2020.

[50] A. Triantafyllopoulos, G. Keren, J. Wagner, I. Steiner, and B. Schuller, "Towards robust speech emotion recognition using deep residual networks for speech enhancement," in *Proc. INTERSPEECH*, (Graz, Austria), pp. 1691–1695, 2019.

[51] J. Monaghan, T. Goehring, X. Yang, F. Bolner, S. Wang, Guo, M. Wright, and S. Bleeck, "Auditory inspired machine learning techniques can improve speech intelligibility and quality for hearing-impaired listeners," *The Journal of the Acoustical Society of America*, vol. 141, no. 3, pp. 1985–1998, 2017.

[52] G. Keren, J. Han, and B. Schuller, "Scaling speech enhancement in unseen environments with noise embeddings," in *Proc. CHiME*, (Hyderabad, India), pp. 25–29, 2018.

[53] D. Liu, P. Smaragdis, and M. Kim, "Experiments on deep learning for speech denoising," in *Proc. INTERSPEECH*, (Singapore, Singapore), pp. 2685–2689, 2014.

[54] M. Kolbæk, Z. H. Tan, and J. Jensen, "Speech intelligibility potential of general and specialized deep neural network based speech enhancement systems," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 1, pp. 153–167, 2016.

[55] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder.," in *Proc. INTERSPEECH*, (Lyon, France), pp. 436–440, 2013.

[56] S. Pascual, A. Bonafonte, and J. Serrà, "SEGAN: Speech enhancement generative adversarial network," in *Proc. INTERSPEECH*, (Stockholm, Sweden), pp. 3642–3646, 2017.

[57] B. Tolooshams, R. Giri, A. H. Song, U. Isik, and A. Krishnaswamy, "Channel-attention dense u-net for multichannel speech enhancement," in *Proc. ICASSP*, (Barcelona, Spain), pp. 836–840, 2020.

[58] H.-S. Choi, J.-H. Kim, J. Huh, A. Kim, J.-W. Ha, and K. Lee, "Phase-aware speech enhancement with deep complex u-net," in *Proc. ICLR*, (New Orleans, LA, USA), 2019. 20 pages.

[59] N. L. Westhausen and B. T. Meyer, "Dual-signal transformation LSTM network for real-time noise suppression," in *Proc. INTERSPEECH*, (Shanghai, China), pp. 2477–2481, 2020.

[60] J. Ming, R. Srinivasan, and D. Crookes, "A corpus-based approach to speech enhancement from nonstationary noise," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 822–836, 2011.

[61] T. Goehring, F. Bolner, J. J. Monaghan, B. Van Dijk, A. Zarowski, and S. Bleeck, "Speech enhancement based on neural networks improves speech intelligibility in noise for cochlear implant users," *Hearing research*, vol. 344, pp. 183–194, 2017.

[62] L. Girin, S. Gannot, and X. Li, "Audio source separation into the wild," *Computer Vision and Pattern Recognition*, pp. 53–78, 2018.

[63] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.

[64] P. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Deep learning for monaural speech separation," in *Proc. ICASSP*, (Florence, Italy), pp. 1562–1566, 2014.

[65] J. Hershey, Z. Chen, J. Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. ICASSP*, (Shanghai, China), pp. 31–35, 2016.

[66] Y. Luo and N. Mesgarani, "TaSNet: Time-domain audio separation network for real-time, single-channel speech separation," in *Proc. ICASSP*, (Calgary, Canada), pp. 696–700, 2018.

[67] Y. Luo and N. Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech and Language Processing*, vol. 27, no. 8, p. 1256–1266, 2019.

[68] V. Delic, Z. Peric, M. Secujski, N. Jakovljevic, J. Nikolic, D. Miskovic, N. Simic, S. Suzic, and T. Delic, "Speech technology progress based on new machine learning paradigm," *Computational Intelligence and Neuroscience*, vol. 2019, pp. 1–19, 2019.

[69] A. Kumar and D. Florêncio, "Speech enhancement in multiple-noise conditions using deep neural networks," in *Proc. INTERSPEECH*, (San Francisco, CA, USA), pp. 3738–3752, 2016.

[70] D. Rethage, J. Pons, and X. Serra, "A wavenet for speech denoising," in *Proc. ICASSP*, (Calgary, Canada), pp. 5069–5073, 2018.

[71] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, "Acoustic scene classification: Classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 16–34, 2015.

[72] S. Gharib, H. Derrar, D. Niizumi, T. Senttula, J. Tommola, T. Heittola, T. Virtanen, and H. Huttunen, "Acoustic scene classification: A competition review," in *Proc. MLSP*, (Aalborg, Denmark), pp. 1–6, 2018.

[73] T. Virtanen, M. D. Plumbley, and D. Ellis, *Computational Analysis of Sound Scenes and Events.* Springer, 2018.

[74] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Transactions on Multimedia*, vol. 17, no. 10, pp. 1733–1746, 2015.

[75] J. Cheng, R. Liang, Z. Liang, L. Zhao, C. Huang, and B. Schuller, "A deep adaptation network for speech enhancement: Combining a relativistic discriminator with multi-kernel maximum mean discrepancy," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 41–53, 2021.

[76] M. Krawczyk and T. Gerkmann, "STFT phase reconstruction in voiced speech for an improved single-channel speech enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1931–1940, 2014.

[77] G. Wichern and J. Le Roux, "Phase reconstruction with learned time-frequency representations for single-channel speech separation," in *Proc. IWAENC*, pp. 396–400, 2018.

[78] Z.-Q. Wang, K. Tan, and D. Wang, "Deep learning based phase reconstruction for speaker separation: A trigonometric perspective," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 71–75, 2019.

[79] M. Strake, B. Defraene, K. Fluyt, W. Tirry, and T. Fingscheidt, "Fully convolutional recurrent networks for speech enhancement," in *Proc. ICASSP*, pp. 6674–6678, 2020.

[80] A. Défossez, G. Synnaeve, and Y. Adi, "Real time speech enhancement in the waveform domain," in *INTERSPEECH*, 2020.

[81] R. Giri, U. Isik, and A. Krishnaswamy, "Attention Wave-U-Net for speech enhancement," in *Proc. WASPAA*, pp. 249–253, 2019.

[82] H. R. Guimarães, H. Nagano, and D. W. Silva, "Monaural speech enhancement through deep wave-U-net," *Expert Systems with Applications*, vol. 158, p. 113582, 2020.

[83] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Proc. NeurIPS*, vol. 27, 2014.

[84] J. Su, Z. Jin, and A. Finkelstein, "HiFi-GAN: High-fidelity denoising and dereverberation based on speech deep features in adversarial networks," 2020.

[85] S.-W. Fu, C.-F. Liao, Y. Tsao, and S.-D. Lin, "Metricgan: Generative adversarial networks based black-box metric scores optimization for speech enhancement," in *International Conference on Machine Learning*, pp. 2031–2041, 2019.

[86] S. Pascual, J. Serrà, and A. Bonafonte, "Towards generalized speech enhancement with generative adversarial networks," in *Proc. INTERSPEECH*, 2019.

[87] Z. Li, L. Dai, Y. Song, and I. Mcloughlin, "A conditional generative model for speech enhancement," *Circuits, Systems, and Signal Processing*, vol. 37, pp. 5005–5022, 2018.

[88] D. Baby and S. Verhulst, "Sergan: Speech enhancement using relativistic generative adversarial networks with gradient penalty," in *Proc. ICASSP*, pp. 106–110, 2019.

[89] H. Phan, I. V. McLoughlin, L. Pham, O. Y. Chén, P. Koch, M. De Vos, and A. Mertins, "Improving GANs for speech enhancement," *IEEE Signal Processing Letters*, vol. 27, pp. 1700–1704, 2020.

[90] S. Winter, H. Sawada, and S. Makino, "Geometrical interpretation of the PCA subspace approach for overdetermined blind source separation," *EURASIP J. ADV. SIG PR.*, vol. 2006, no. 1, 2006. 11 pages.

[91] N. Mitianoudis and M. E. Davies, "Audio source separation: Solutions and problems," *INT J. ADAPT CONTROL.*, vol. 18, no. 3, pp. 299–314, 2004.

[92] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 18, no. 3, pp. 550–563, 2010.

[93] T. Beierholm, B. D. Pedersen, and O. Winther, "Low complexity bayesian single channel source separation," in *Proc. ICASSP*, (Montreal, Canada), pp. 529–532, 2004.

[94] M. Schmidt and R. Olsson, "Single-channel speech separation using sparse nonnegative matrix factorization," in *Proc. ICSPL*, (Pittsburgh, PA), pp. 2–5, 2006.

[95] F. Weninger and B. W. Schuller, "Optimization and parallelization of monaural source separation algorithms in the openBliSSART toolkit," *J. Signal Process. Syst.*, vol. 69, no. 3, pp. 267–277, 2012.

[96] B. Gao, W. L. Woo, and S. Dlay, “Single-channel source separation using emd-subband variable regularized sparse features,” *IEEE Trans. Acoustics, Speech, Signal Process.*, vol. 19, no. 4, pp. 961–976, 2011.

[97] K. Patki, “Review of single channel source separation techniques,” in *Proc. ISMIR*, (Curitiba, Brazil), pp. 1–5, 2013.

[98] D. Wang and J. Chen, “Supervised speech separation based on deep learning: An overview,” *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[99] X. Zhang and D. Wang, “A deep ensemble learning method for monaural speech separation,” *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 24, no. 5, pp. 967–977, 2016.

[100] S. Xia, H. Li, and X. Zhang, “Using optimal ratio mask as training target for supervised speech separation,” in *Proc. APSIPA*, (Kuala Lumpur, Malaysia), pp. 163–166, 2017.

[101] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S. Chang, and T. Sainath, “Deep learning for audio signal processing,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 206–219, 2019.

[102] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *Proc. MICCAI*, (London, UK), pp. 234–241, 2015.

[103] A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde, “Singing voice separation with deep U-Net convolutional networks,” in *Proc. ISMIR*, (Suzhou, China), pp. 745–751, 2017.

[104] N. Takahashi and Y. Mitsufuji, “Multi-scale multi-band densenets for audio source separation,” in *Proc. WASPAA*, (New Paltz, NY, USA), pp. 21–25, 2017.

[105] M. Ikram and D. Morgan, “Permutation inconsistency in blind speech separation: Investigation and solutions,” *IEEE Trans. Speech and Audio Processing*, vol. 13, pp. 1 – 13, 2005.

[106] Z. Chen, Y. Luo, and N. Mesgarani, “Deep attractor network for single-microphone speaker separation,” in *Proc. ICASSP*, (New Orleans, LA), pp. 246–250, 2017.

[107] Y. Luo and N. Mesgarani, “Tasnet: Time-domain audio separation network for real-time, single-channel speech separation,” in *Proc. ICASSP*, pp. 696–700, 2018.

[108] K. Žmolíková, M. Delcroix, K. Kinoshita, T. Ochiai, T. Nakatani, L. Burget, and J. Černocký, “Speakerbeam: Speaker aware neural network for target speaker extraction in speech mixtures,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 4, pp. 800–814, 2019.

[109] Q. Wang, H. Muckenhirn, K. W. Wilson, P. Sridhar, Z. Wu, J. R. Hershey, R. A. Saurous, R. J. Weiss, Y. Jia, and I. L. Moreno, “Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking,” in *Proc. Interspeech*, (Graz, Austria), 2019.

[110] J. Wang, J. J. Chen, D. Su, L. Chen, M. Yu, Y. Qian, and D. Yu, "Deep extractor network for target speaker recovery from single channel speech mixtures," in *Proc. INTERSPEECH*, 2018.

[111] M. Delcroix, T. Ochiai, K. Zmolikova, K. Kinoshita, N. Tawara, T. Nakatani, and S. Araki, "Improving speaker discrimination of target speech extraction with time-domain speakerbeam," in *Proc. ICASSP*, pp. 691–695, 2020.

[112] R. Xu, R. Wu, Y. Ishiwaka, C. Vondrick, and C. Zheng, "Listening to sounds of silence for speech denoising," in *Proc. NeurIPS*, (Vancouver, Canada), 2020. 6 pages.

[113] Z. Zhang, B. He, and Z. Zhang, "X-tasnet: Robust and accurate time-domain speaker extraction network," in *Proc. INTERSPEECH*, 2020.

[114] M. Ge, C. Xu, L. Wang, E. S. Chng, J. Dang, and H. Li, "L-spex: Localized target speaker extraction," *arXiv preprint arXiv:2202.09995*, 2022.

[115] J. R. Zapata and E. Gomez, "Using voice suppression algorithms to improve beat tracking in the presence of highly predominant vocals," in *Proc. ICASSP*, (Vancouver, Canada), pp. 51–55, 2013.

[116] Y. A. Chung, W. N. Hsu, H. Tang, and J. Glass, "An Unsupervised Autoregressive Model for Speech Representation Learning," in *Proc. INTERSPEECH*, (Graz, Austria), pp. 146–150, 2019.

[117] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. Schuller, "Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR," in *Proc. LVA/ICA*, (Liberec, Czech Republic), pp. 91–99, 2015.

[118] K. Kinoshita, T. Ochiai, M. Delcroix, and T. Nakatani, "Improving noise robust automatic speech recognition with single-channel time-domain enhancement network," in *Proc. ICASSP*, pp. 7009–7013, 2020.

[119] S. Sivasankaran, A. A. Nugraha, E. Vincent, J. A. Morales-Cordovilla, S. Dalmia, I. Illina, and A. Liutkus, "Robust asr using neural network based speech enhancement and feature simulation," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 482–489, IEEE, 2015.

[120] C. Zorilă, C. Boeddeker, R. Doddipatla, and R. Haeb-Umbach, "An investigation into the effectiveness of enhancement in asr training and test for chime-5 dinner party transcription," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 47–53, IEEE, 2019.

[121] D. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. Cubuk, and Q. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Proc. INTERSPEECH*, (Graz, Austria), pp. 2613–2617, 2019.

[122] K. Iwamoto, T. Ochiai, M. Delcroix, R. Ikeshita, H. Sato, S. Araki, and S. Katagiri, "How bad are artifacts?: Analyzing the impact of speech enhancement errors on asr," *arXiv preprint arXiv:2201.06685*, 2022.

[123] Z.-Q. Wang and D. Wang, "A joint training framework for robust automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 796–806, 2016.

[124] A. Narayanan, A. Misra, and K. K. Chin, "Large-scale, sequence-discriminative, joint adaptive training for masking-based robust asr,"

[125] D. Ma, N. Hou, H. Xu, E. S. Chng, *et al.*, "Multitask-based joint learning approach to robust asr for radio communication speech," in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 497–502, IEEE, 2021.

[126] Z. Chen, S. Watanabe, H. Erdogan, and J. R. Hershey, "Speech enhancement and recognition using multi-task learning of long short-term memory recurrent neural networks," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[127] C. Kim, A. Garg, D. Gowda, S. Mun, and C. Han, "Streaming end-to-end speech recognition with jointly trained neural feature enhancement," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6773–6777, IEEE, 2021.

[128] B. Liu, S. Nie, S. Liang, W. Liu, M. Yu, L. Chen, S. Peng, and C. Li, "Jointly Adversarial Enhancement Training for Robust End-to-End Speech Recognition," in *Proc. Interspeech 2019*, pp. 491–495, 2019.

[129] L. Li, Y. Kang, Y. Shi, L. Kürzinger, T. Watzel, and G. Rigoll, "Adversarial joint training with self-attention mechanism for robust end-to-end speech recognition," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2021, no. 1, pp. 1–16, 2021.

[130] Q.-S. Zhu, J. Zhang, Z.-Q. Zhang, and L.-R. Dai, "Joint training of speech enhancement and self-supervised model for noise-robust asr," *arXiv preprint arXiv:2205.13293*, 2022.

[131] G. Cámbara, F. López, D. Bonet, P. Gómez, C. Segura, M. Farrús, and J. Luque, "TASE: Task-aware speech enhancement for wake-up word detection in voice assistants," *Applied Sciences*, vol. 12, no. 4, p. 1974, 2022.

[132] Y. Gu, Z. Du, H. Zhang, and X. Zhang, "A monaural speech enhancement method for robust small-footprint keyword spotting," *arXiv preprint arXiv:1906.08415*, 2019.

[133] H. Zhou, J. Du, Y.-H. Tu, and C.-H. Lee, "Using speech enhancement preprocessing for speech emotion recognition in realistic noisy conditions," in *Proc. INTERSPEECH*, p. E1, 2020.

[134] S. Liu, A. Triantafyllopoulos, Z. Ren, and B. W. Schuller, "Towards speech robustness for acoustic scene classification," 2020.

[135] J. Kim and M. Hahn, "Speech enhancement using a two-stage network for an efficient boosting strategy," *IEEE Signal Processing Letters*, vol. 26, no. 5, pp. 770–774, 2019.

[136] Y. Xu, J. Du, L. Dai, and C. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 7–19, 2015.

[137] G. Roma, E. Grais, A. Simpson, I. Sobieraj, and M. Plumbley, "Untwist: A new toolbox for audio source separation," in *Proc. ISMIR*, (New York City, NY, USA), 2016. 4 pages.

[138] M. Pariente, S. Cornell, J. Cosentino, S. Sivasankaran, E. Tzinis, J. Heitkaemper, M. Olvera, F.-R. Stöter, M. Hu, J. M. Martín-Doñas, D. Ditter, A. Frank, A. Deleforge, and E. Vincent, "Asteroid: the PyTorch-based audio source separation toolkit for researchers," *arXiv preprint arXiv:2005.04132*, 2020.

[139] Z. Wang, J. L. Roux, and J. R. Hershey, "Alternative objective functions for deep clustering," in *Proc. ICASSP*, (Calgary, Canada), pp. 686–690, 2018.

[140] F. Weninger, A. Lehmann, and B. Schuller, "OpenBliSSART: Design and evaluation of a research toolkit for blind source separation in audio recognition tasks," in *Proc. ICASSP)*, (Wuhan, China), pp. 1625–1628, 2011.

[141] Y. Salaün, E. Vincent, N. Bertin, N. Souviraà-Labastie, X. Jaureguiberry, D. Tran, and F. Bimbot, "The flexible audio source separation toolbox version 2.0," in *Proc. ICASSP*, (Florence, Italy), 2014. 3 pages.

[142] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, (Lille, France), pp. 448–456, 2015.

[143] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?," in *Proc. NeurIPS*, (Montreal, Canada), pp. 2483–2493, 2018.

[144] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. ICML*, (Haifa, Israel), pp. 807–814, 2010.

[145] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *Proc. ICASSP*, (Brisbane, Australia), pp. 5206–5210, 2015.

[146] J. Gemmeke, D. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. Moore, M. Plakal, and M. Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *Proc. ICASSP*, (New Orleans, LA, USA), pp. 776–780, 2017.

[147] A. Nagrani, J. Chung, and A. Zisserman, "VoxCeleb: A large-scale speaker identification dataset," in *Proc. INTERSPEECH*, (Stockholm, Sweden), pp. 2616–2620, 2017.

[148] J. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep speaker recognition," in *Proc. INTERSPEECH*, (Hyderabad, India), pp. 1086–1090, 2018.

[149] J. S. Garofolo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) Other," in *Philadelphia: Linguistic Data Consortium*, 1993.

[150] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," *NASA STI/Recon Technical Report*, vol. 93, p. 27403, 1993.

[151] Y. Liu and D. Wang, "Divide and conquer: A deep casa approach to talker-independent monaural speaker separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 2092–2102, 2019.

[152] C. Valentini-Botinhao, "Noisy speech database for training speech enhancement algorithms and TTS models," (University of Edinburgh. School of Informatics. Centre for SpeechTechnology Research (CSTR)), 2017.

[153] Y. Xu, J. Du, L. R. Dai, and C. H. Lee, "Dynamic noise aware training for speech enhancement based on deep neural networks," in *Proc. INTERSPEECH*, (Singapore, Singapore), pp. 2670–2674, 2014.

[154] K. Jeon and H. Kim, "Audio enhancement using local SNR-based sparse binary mask estimation and spectral imputation," *Digital Signal Processing*, vol. 68, pp. 138–151, 2017.

[155] L. Sari and M. Hasegawa-Johnson, "Speaker adaptation with an auxiliary network," in *Proc. MLSLP*, (Hyderabad, India), 2018. 3 pages.

[156] J. Zhang, G. Tian, Y. Mu, and W. Fan, "Supervised deep learning with auxiliary networks," in *Proc. KDD*, (New York, NY, USA), pp. 353–361, 2014.

[157] M. H. Soni, N. Shah, and H. A. Patil, "Time-frequency masking-based speech enhancement using generative adversarial network," in *Proc. ICASSP*, (Calgary, Canada), pp. 5039–5043, 2018.

[158] J. Thiemann, N. Ito, and E. Vincent, "The diverse environments multi-channel acoustic noise database (DEMAND): A database of multichannel environmental noise recordings," *The Journal of the Acoustical Society of America*, vol. 133, no. 5, 2013. 6 pages.

[159] C. Veaux, J. Yamagishi, and S. King, "The voice bank corpus: Design, collection and data analysis of a large regional accent speech database," *Proc. O-COCOSDA/CASLRE*, pp. 1–4, 2013.

[160] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, 2008.

[161] Y. Xu, J. Du, L. Dai, and C. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, 2014.

[162] F.-R. Stöter, A. Liutkus, and N. Ito, "The 2018 signal separation evaluation campaign," in *Proc. LVA/ICA*, (Guildford, UK), pp. 293–305, 2018.

[163] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[164] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks,"

IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 10, pp. 1901–1913, 2017.

[165] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *Proc. ICASSP*, pp. 241–245, 2017.

[166] Z. Ren, Q. Kong, J. Han, M. D. Plumbley, and B. W. Schuller, "Attention-based atrous convolutional neural networks: Visualisation and understanding perspectives of acoustic scenes," in *Proc. ICASSP*, (Brighton, UK), pp. 56–60, 2019.

[167] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. ICLR*, (San Juan, PR), pp. 1–13, 2016.

[168] Z. Zhang, J. Geiger, J. Pohjalainen, A. E.-D. Mousa, W. Jin, and B. Schuller, "Deep learning for environmentally robust speech recognition: An overview of recent developments," *ACM Trans. Intell Syst. Technol.*, vol. 9, no. 5, 2018. 14 pages.

[169] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, (Munich, Germany), pp. 234–241, 2015.

[170] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.

[171] I. Aslan, F. Xu, H. Uszkoreit, A. Krüger, and J. Steffen, "Compass2008: Multimodal, multilingual and crosslingual interaction for mobile tourist guide applications," in *International Conference on Intelligent Technologies for Interactive Entertainment*, pp. 3–12, 2005.

[172] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L.-L. Lim, B. Roomi, and P. Hall, "English conversational telephone speech recognition by humans and machines," in *Proc. INTERSPEECH*, (Stockholm, Sweden), pp. 132–136, 2017.

[173] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," *Proc. NeurIPS*, 2020. 12 pages.

[174] W. N. Hsu, B. Bolte, Y. H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," in *arXiv:2106.07447*, 2021.

[175] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. Pino, A. Baevski, A. Conneau, and M. Auli, "XLS-R: Self-supervised cross-lingual speech representation learning at scale," in *arXiv:2111.09296*, 2021.

[176] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, 2021. 20 pages.

[177] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, *et al.*, "Deep Speech 2 : End-to-end speech recognition in english and mandarin," in *Proc. ICML*, (New York, New York, USA), pp. 173–182, 2016.

[178] N. Zheng, Y. Shi, W. Rong, and Y. Kang, "Effects of skip connections in CNN-based architectures for speech enhancement," *Journal of Signal Processing Systems*, vol. 92, p. 875–884, 2020.

[179] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. ICML*, (Pittsburgh, Pennsylvania), pp. 369–376, 2006.

[180] C. Spille, B. Kollmeier, and B. T. Meyer, "Comparing human and automatic speech recognition in simple and complex acoustic scenes," *Computer Speech & Language*, vol. 52, pp. 123–140, 2018.

[181] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[182] S. Yin, C. Liu, Z. Zhang, Y. Lin, D. Wang, J. Tejedor, F. Zheng, and Y. Li, "Noisy training for deep neural networks in speech recognition," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2015, no. 1, 2015. 21 pages.

[183] J. Kim, M. El-Khamy, and J. Lee, "Bridgenets: Student-teacher transfer learning based on recursive neural networks and its application to distant speech recognition," in *Proc. ICASSP*, (Alberta, Canada), pp. 5719–5723, 04 2018.

[184] Z. Meng, J. Li, Y. Gaur, and Y. Gong, "Domain adaptation via teacher-student learning for end-to-end speech recognition," in *Proc. ASRU*, (Sentosa, Singapore), pp. 268–275, 2019.

[185] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," 2018.

[186] W. Dai, C. Dai, S. Qu, J. Li, and S. Das, "Very deep convolutional neural networks for raw waveforms," in *Proc. ICASSP*, pp. 421–425, 2017.

[187] B. W. Schuller, "Speech emotion recognition: Two decades in a nutshell, benchmarks, and ongoing trends," *Communications of the ACM*, vol. 61, no. 5, pp. 90–99, 2018.

[188] M. D. McDonnell and W. Gao, "Acoustic scene classification using deep residual networks with late fusion of separated high and low frequency paths," in *Proc. ICASSP*, pp. 141–145, 2020.

[189] T. Heittola, A. Mesaros, and T. Virtanen, "Acoustic scene classification in dcase 2020 challenge: generalization across devices and low complexity solutions," in *Proc. DCASE2020*, pp. 56–60, 2020.

[190] L. Le, A. Patterson, and M. White, "Supervised autoencoders: Improving generalization performance with unsupervised regularizers," in *Proceedings of the Conference on Neural Information Processing Systems*, pp. 107–117, 2018.

[191] Z. Ren, A. Baird, J. Han, Z. Zhang, and B. Schuller, "Generating and protecting against adversarial attacks for deep speech-based emotion recognition models," in *Proc. ICASSP*, pp. 7184–7188, 2020.

[192] S. Wang, A. Mesaros, T. Heittola, and T. Virtanen, "A curated dataset of urban scenes for audio-visual scene analysis," in *Proc. ICASSP*, 2021.

[193] "Coronavirus disease 2019 in elderly patients: Characteristics and prognostic factors based on 4-week follow-up," *Journal of Infection*, vol. 80, no. 6, pp. 639–645, 2020.

[194] The Lancet Infectious Diseases, "Explaining the unexplained hepatitis in children," *The Lancet Infectious Diseases*, vol. 22, no. 6, 2022.

[195] B. Singh, B. Datta, A. Ashish, and G. Dutta, "A comprehensive review on current COVID-19 detection methods: From lab care to point of care diagnosis," *Sensors International*, vol. 2, p. 100119, 2021.

[196] C. Zheng, W. Shao, X. Chen, B. Zhang, G. Wang, and W. Zhang, "Real-world effectiveness of COVID-19 vaccines: a literature review and meta-analysis," *International Journal of Infectious Diseases*, vol. 114, pp. 252–260, 2022.

[197] M. V. Fasano, M. Padula, M. A. Azrak, A. J. Avico, M. Sala, and M. F. Andreoli, "Consequences of lockdown during COVID-19 pandemic in lifestyle and emotional state of children in argentina," *Frontiers in Pediatrics*, vol. 9, 2021.

[198] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Deep learning applications for COVID-19," *Journal of Big Data*, vol. 8, no. 1, pp. 1–54, 2021.

[199] F. Zhang, "Application of machine learning in CT images and X-rays of COVID-19 pneumonia," *Medicine*, vol. 100, no. 36, 2021.

[200] "contour-enhanced attention cnn for ct-based covid-19 segmentation,"

[201] A. I. Khan, J. L. Shah, and M. M. Bhat, "CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images," *Computer Methods and Programs in Biomedicine*, vol. 196, p. 105581, 2020.

[202] S. A. Harmon, T. H. Sanford, S. Xu, and et al., "Artificial intelligence for the detection of COVID-19 pneumonia on chest CT using multinational datasets," *Nature Communications*, vol. 11, 2020. 7 pages.

[203] C. Brown, J. Chauhan, A. Grammenos, J. Han, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, and C. Mascolo, "Exploring automatic diagnosis of COVID-19 from crowdsourced respiratory sound data," in *Proceedings of the SIGKDD International Conference on Knowledge Discovery & Data Mining*, (New York, NY, USA), p. 3474–3484, 2020.

[204] B. L. Smarr, K. Aschbacher, S. M. Fisher, A. Chowdhary, S. Dilchert, K. Puldon, A. Rao, F. M. Hecht, and A. E. Mason, "Feasibility of continuous fever monitoring using wearable devices," *Sci. Rep.*, vol. 10, no. 1, pp. 1–11, 2020.

[205] A. Shapiro, N. Marinsek, I. Clay, B. Bradshaw, E. Ramirez, J. Min, A. Trister, Y. Wang, T. Althoff, and L. Foschini, "Characterizing COVID-19 and Influenza illnesses in the real world via person-generated health data," *Patterns*, vol. 2, no. 1, p. 100188, 2021.

[206] A. Bhargava and A. Bansal, "Novel coronavirus (covid-19) diagnosis using computer vision and artificial intelligence techniques: a review," *Multimedia tools and applications*, vol. 80, no. 13, pp. 19931–19946, 2021.

[207] F. M. Shah, S. K. S. Joy, F. Ahmed, T. Hossain, M. Humaira, A. S. Ami, S. Paul, M. A. R. K. Jim, and S. Ahmed, "A comprehensive survey of COVID-19 detection using medical images," *SN Computer Science*, vol. 2, no. 6, pp. 1–22, 2021.

[208] C. Zhao, Y. Xu, Z. He, J. Tang, Y. Zhang, J. Han, Y. Shi, and W. Zhou, "Lung segmentation and automatic detection of COVID-19 using radiomic features from chest CT images," *Pattern Recognition*, vol. 119, p. 108071, 2021.

[209] V. de Carvalho Brito, P. R. S. dos Santos, N. R. de Sales Carvalho, and A. O. de Carvalho Filho, "COVID-index: A texture-based approach to classifying lung lesions based on CT images," *Pattern Recognition*, vol. 119, p. 108083, 2021.

[210] S. Vaid, R. Kalantar, and M. Bhandari, "Deep learning COVID-19 detection bias: accuracy through artificial intelligence," *International Orthopaedics*, vol. 44, no. 8, pp. 1539–1542, 2020.

[211] I. D. Apostolopoulos, S. I. Aznaouridis, and M. A. Tzani, "Extracting possibly representative COVID-19 biomarkers from x-ray images with deep learning approach and image data related to pulmonary diseases," *Journal of Medical and Biological Engineering*, vol. 40, no. 3, pp. 462–469, 2020.

[212] I. D. Apostolopoulos and T. A. Mpesiana, "COVID-19: Automatic detection from x-ray images utilizing transfer learning with convolutional neural networks," *Physical and engineering sciences in medicine*, vol. 43, no. 2, pp. 635–640, 2020.

[213] S. Ahuja, B. K. Panigrahi, N. Dey, V. Rajinikanth, and T. K. Gandhi, "Deep transfer learning-based automated detection of COVID-19 from lung CT scan slices," *Applied Intelligence*, vol. 51, no. 1, pp. 571–585, 2021.

[214] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (COVID-19) using x-ray images and deep convolutional neural networks," *Pattern Analysis and Applications*, vol. 24, no. 3, pp. 1207–1220, 2021.

[215] N. M. Elshennawy and D. M. Ibrahim, "Deep-pneumonia framework using deep learning models based on chest x-ray images," *Diagnostics*, vol. 10, no. 9, p. 649, 2020.

[216] J. Chen, L. Wu, J. Zhang, L. Zhang, D. Gong, Y. Zhao, Q. Chen, S. Huang, M. Yang, X. Yang, *et al.*, "Deep learning-based model for detecting 2019 novel coronavirus pneumonia on high-resolution computed tomography," *Sci. Rep.*, vol. 10, no. 1, pp. 1–11, 2020.

[217] C. Zheng, X. Deng, Q. Fu, Q. Zhou, J. Feng, H. Ma, W. Liu, and X. Wang, "Deep learning-based detection for COVID-19 from chest CT using weak label," *MedRxiv*, 2020.

[218] D. Shome, T. Kar, S. N. Mohanty, P. Tiwari, K. Muhammad, A. AlTameem, Y. Zhang, and A. K. J. Saudagar, "Covid-transformer: Interpretable COVID-19 detection using vision transformer for healthcare," *International Journal of Environmental Research and Public Health*, vol. 18, no. 21, p. 11086, 2021.

[219] P. Afshar, S. Heidarian, F. Naderkhani, A. Oikonomou, K. N. Plataniotis, and A. Mohammadi, "Covid-caps: A capsule network-based framework for identification of COVID-19 cases from X-ray images," *Pattern Recognition Letters*, vol. 138, pp. 638–643, 2020.

[220] R. Punia, L. Kumar, M. Mujahid, and R. Rohilla, "Computer vision and radiology for covid-19 detection," in *Proc. INCET*, pp. 1–5, 2020.

[221] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao, J. Guo, M. Cai, J. Yang, Y. Li, X. Meng, *et al.*, "A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19)," *Eur. Radiol.*, pp. 1–9, 2021.

[222] M. F. Aslan, M. F. Unlersen, K. Sabanci, and A. Durdu, "CNN-based transfer learning–BiLSTM network: A novel approach for COVID-19 infection detection," *Appl. Soft Comput.*, vol. 98, p. 106912, 2021.

[223] J. Diaz-Escobar, N. E. Ordóñez-Guillén, S. Villarreal-Reyes, A. Galaviz-Mosqueda, V. Kober, R. Rivera-Rodriguez, and J. E. Lozano Rizk, "Deep-learning based detection of COVID-19 using lung ultrasound imagery," *Plos one*, vol. 16, no. 8, p. e0255886, 2021.

[224] J. Born, N. Wiedemann, M. Cossio, C. Buhre, G. Brändle, K. Leidermann, J. Goulet, A. Aujayeb, M. Moor, B. Rieck, *et al.*, "Accelerating detection of lung pathologies with explainable ultrasound image analysis," *Applied Sciences*, vol. 11, no. 2, p. 672, 2021.

[225] M. Shorfuzzaman and M. S. Hossain, "MetaCOVID: A Siamese neural network framework with contrastive loss for n-shot diagnosis of COVID-19 patients," *Pattern Recognit.*, vol. 113, p. 107700, 2021. this issue.

[226] X. Chen, L. Yao, T. Zhou, J. Dong, and Y. Zhang, "Momentum contrastive learning for few-shot COVID-19 diagnosis from chest CT images," *Pattern Recognit.*, vol. 113, p. 107826, 2021. this issue.

[227] J. Hou, J. Xu, L. Jiang, S. Du, R. Feng, Y. Zhang, F. Shan, and X. Xue, "Periphery-aware COVID-19 diagnosis with contrastive representation enhancement," *Pattern Recognition*, vol. 118, p. 108005, 2021.

[228] Z. Wang, Y. Xiao, Y. Li, J. Zhang, F. Lu, M. Hou, and X. Liu, "Automatically discriminating and localizing COVID-19 from community-acquired pneumonia on chest X-rays," *Pattern Recognit.*, vol. 110, p. 107613, 2021.

[229] K. Zhang, X. Liu, J. Shen, Z. Li, Y. Sang, X. Wu, Y. Zha, W. Liang, C. Wang, K. Wang, *et al.*, "Clinically applicable AI system for accurate diagnosis, quantitative measurements, and prognosis of COVID-19 pneumonia using computed tomography," *Cell*, vol. 181, no. 6, pp. 1423–1433, 2020.

[230] A. Malhotra, S. Mittal, P. Majumdar, S. Chhabra, K. Thakral, M. Vatsa, R. Singh, S. Chaudhury, A. Pudrod, and A. Agrawal, "Multi-task driven explainable diagnosis of COVID-19 using chest X-ray images," *Pattern Recognition*, vol. 122, p. 108243, 2022.

[231] J. Li, G. Zhao, Y. Tao, P. Zhai, H. Chen, H. He, and T. Cai, "Multi-task contrastive learning for automatic CT and X-ray diagnosis of COVID-19," *Pattern Recognit.*, vol. 114, p. 107848, 2021. this issue.

[232] G. Bao, H. Chen, T. Liu, G. Gong, Y. Yin, L. Wang, and X. Wang, "COVID-MTL: Multitask learning with Shift3D and random-weighted loss for COVID-19 diagnosis and severity assessment," *Pattern Recognition*, vol. 124, p. 108499, 2022.

[233] J. Wu, H. Xu, S. Zhang, X. Li, J. Chen, J. Zheng, Y. Gao, Y. Tian, Y. Liang, and R. Ji, "Joint segmentation and detection of COVID-19 via a sequential region generation network," *Pattern Recognition*, vol. 118, p. 108006, 2021.

[234] J. Li, G. Zhao, Y. Tao, P. Zhai, H. Chen, H. He, and T. Cai, "Multi-task contrastive learning for automatic CT and X-ray diagnosis of COVID-19," *Pattern Recognition*, vol. 114, p. 107848, 2021.

[235] K. He, W. Zhao, X. Xie, W. Ji, M. Liu, Z. Tang, Y. Shi, F. Shi, Y. Gao, J. Liu, J. Zhang, and D. Shen, "Synergistic learning of lung lobe segmentation and hierarchical multi-instance classification for automated severity assessment of COVID-19 in ct images," *Pattern Recognition*, vol. 113, p. 107828, 2021.

[236] L. Brunese, F. Mercaldo, A. Reginelli, and A. Santone, "Explainable deep learning for pulmonary disease and coronavirus COVID-19 detection from X-rays," *Comput. Methods Programs Biomed.*, vol. 196, p. 105608, 2020.

[237] A. Kumar, A. R. Tripathi, S. C. Satapathy, and Y.-D. Zhang, "SARS-Net: COVID-19 detection from chest x-rays by combining graph convolutional network and convolutional neural network," *Pattern Recognition*, vol. 122, p. 108255, 2022.

[238] L. Li, L. Qin, Z. Xu, Y. Yin, X. Wang, B. Kong, J. Bai, Y. Lu, Z. Fang, Q. Song, *et al.*, "Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT," *Radiology*, p. 16 pages, 2020.

[239] J. Wang, Y. Bao, Y. Wen, H. Lu, H. Luo, Y. Xiang, X. Li, C. Liu, and D. Qian, "Prior-attention residual learning for more discriminative COVID-19 screening in CT images," *IEEE Trans. Med. Imaging*, vol. 39, no. 8, pp. 2572–2583, 2020.

[240] G. Deshpande, A. Batliner, and B. W. Schuller, "AI-based human audio processing for COVID-19: A comprehensive overview," *Pattern Recognition*, vol. 122, p. 108289, 2022.

[241] K. Qian, M. Schmitt, H. Zheng, T. Koike, J. Han, J. Liu, W. Ji, J. Duan, M. Song, Z. Yang, Z. Ren, S. Liu, Z. Zhang, Y. Yamamoto, and B. W. Schuller, "Computer audition for fighting the SARS-CoV-2 Corona crisis – Introducing the multi-task speech corpus for COVID-19," *IEEE Internet Things J.*, p. 12 pages, 2021.

[242] T. K. Dash, S. Mishra, G. Panda, and S. C. Satapathy, "Detection of COVID-19 from speech signal using bio-inspired based cepstral features," *Pattern Recognit.*, vol. 117, p. 107999, 2021.

[243] T. Xia, D. Spathis, J. Ch, A. Grammenos, J. Han, A. Hasthanasombat, E. Bondareva, T. Dang, A. Floto, P. Cicuta, *et al.*, "COVID-19 sounds: A large-scale audio dataset for digital respiratory screening," in *Proc. NeurIPS*, 2021.

[244] A. Imran, I. Posokhova, H. N. Qureshi, U. Masood, M. S. Riaz, K. Ali, C. N. John, M. I. Hussain, and M. Nabeel, "AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app," *Informatics in Medicine Unlocked*, vol. 20, p. 100378, 2020.

[245] T. Yan, H. Meng, E. Parada-Cabaleiro, S. Liu, M. Song, and B. W. Schuller, "Coughing-based recognition of covid-19 with spatial attentive convlstm recurrent neural networks," in *Proc. INTERSPEECH*, (Brno, Czechia), 2021.

[246] T. Yan, H. Meng, S. Liu, E. Parada-Cabaleiro, Z. Ren, and B. W. Schuller, "Convolutional transformer with adaptive position embedding for COVID-19 detection from cough sounds," in *Proc. ICASSP*, (Singapore, Singapore), pp. 9092–9096, 2022.

[247] M. Faezipour and A. Abuzneid, "Smartphone-Based Self-Testing of COVID-19 Using Breathing Sounds," *Telemedicine and e-Health*, vol. 26, no. 10, pp. 1202–1205, 2020.

[248] V. Dentamaro, P. Giglio, D. Impedovo, L. Moretti, and G. Pirlo, "Auco resnet: an end-to-end network for covid-19 pre-screening from cough and breath," *Pattern Recognition*, vol. 127, p. 108656, 2022.

[249] T. Rahman, N. Ibtehaz, A. Khandakar, M. S. A. Hossain, Y. M. S. Mekki, M. Ezeddin, E. H. Bhuiyan, M. A. Ayari, A. Tahir, Y. Qiblawey, *et al.*, "Qucoughscope: An intelligent application to detect covid-19 patients using cough and breath sounds," *Diagnostics*, vol. 12, no. 4, p. 920, 2022.

[250] H. Coppock, A. Gaskell, P. Tzirakis, A. Baird, L. Jones, and B. Schuller, "End-to-end convolutional neural network enables COVID-19 detection from breath and cough audio: a pilot study," *BMJ innovations*, vol. 7, no. 2, 2021.

[251] J. Laguarta, F. Hueto, and B. Subirana, "COVID-19 artificial intelligence diagnosis using only cough recordings," *IEEE Open Journal of Engineering in Medicine and Biology*, vol. 1, pp. 275–281, 2020.

[252] E. A. Mohammed, M. Keyhani, A. Sanati-Nezhad, S. H. Hejazi, and B. H. Far, "An ensemble learning approach to digital corona virus preliminary screening from cough sounds," *Scientific Reports*, vol. 11, no. 1, pp. 1–11, 2021.

[253] A. Pal and M. Sankarasubbu, "Pay attention to the cough: Early diagnosis of COVID-19 using interpretable symptoms embeddings with cough sound signal processing," in *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, (Korea), pp. 620–628, 2021.

[254] S. Anand, V. Sharma, R. Pourush, and S. Jaiswal, "A comprehensive survey on the biomedical signal processing methods for the detection of COVID-19," *Annals of Medicine and Surgery*, 2022. 9 pages.

[255] M. Mitratza, B. M. Goodale, A. Shagadatova, V. Kovacevic, J. van de Wijgert, T. B. Brakenhoff, R. Dobson, B. Franks, D. Veen, A. A. Folarin, *et al.*, "The performance of wearable sensors in the detection of SARS-CoV-2 infection: a systematic review," *The Lancet Digital Health*, vol. 4, no. 5, pp. e370–e383, 2022.

[256] T. Mishra, M. Wang, A. A. Metwally, G. Bogu, A. W. Brooks, A. Bahmani, A. Alavi, A. Celli, E. Higgs, O. Dagan-Rosenfeld, B. Fay, S. Kirkpatrick, R. Kellogg, M. Gibson, T. Wang, E. Hunting, P. Mamic, A. Gany, B. Rolnik, A. B. Ganz, X. Li, and M. P. Snyder, "Pre-symptomatic detection of COVID-19 from smartwatch data," *Nat. Biomed. Eng.*, vol. 4, no. 12, pp. 1208–1220, 2020.

[257] K.-C. Un, C.-K. Wong, Y.-M. Lau, J. C.-Y. Lee, F. C.-C. Tam, W.-H. Lai, Y.-M. Lau, H. Chen, S. Wibowo, X. Zhang, *et al.*, "Observational study on wearable biosensors and machine learning-based remote monitoring of COVID-19 patients," *Sci. Rep.*, vol. 11, no. 1, pp. 1–9, 2021.

[258] R. P. Hirten, M. Danieletto, L. Tomalin, K. H. Choi, M. Zweig, E. Golden, S. Kaur, D. Helmus, A. Biello, R. Pyzik, A. Charney, R. Miotto, B. S. Glicksberg, M. Levin, I. Nabeel, J. Aberg, D. Reich, D. Charney, E. P. Bottinger, L. Keefer, M. Suarez-Farinas, G. N. Nadkarni, and Z. A. Fayad, "Use of physiological data from a wearable device to identify SARS-CoV-2 infection and symptoms and predict COVID-19 diagnosis: Observational study," *Journal of Medical Internet Research*, vol. 23, no. 2, p. e26107, 2021.

[259] J. Radin, N. Wineinger, E. Topol, and S. Steinhubl, "Harnessing wearable device data to improve state-level real-time surveillance of influenza-like illness in the USA: A population-based study," *Lancet Digital Health*, vol. 2, no. 2, pp. 85–93, 2020.

[260] G. Quer, J. Radin, M. Gadaleta, K. Baca-Motes, L. Ariniello, E. Ramos, V. Kheterpal, E. Topol, and S. Steinhubl, "Wearable sensor data and self-reported symptoms for COVID-19 detection," *Nat. Med.*, vol. 2, pp. 1–5, 2020.

[261] A. Natarajan, H.-W. Su, and C. Heneghan, "Assessment of physiological signs associated with COVID-19 measured using wearable devices," *Digital Med.*, vol. 3, no. 156, pp. 1–8, 2020.

[262] A. Natarajan, H.-W. Su, and C. Heneghan, "Assessment of physiological signs associated with covid-19 measured using wearable devices," *NPJ digital medicine*, vol. 3, no. 1, pp. 1–8, 2020.

[263] J. M. Radin, G. Quer, E. Ramos, K. Baca-Motes, M. Gadaleta, E. J. Topol, and S. R. Steinhubl, "Assessment of prolonged physiological and behavioral changes associated with COVID-19 infection," *JAMA Network*, vol. 4, no. 7, p. 4 pages, 2021.

[264] S. Liu, A. Mallol-Ragolta, and B. W. Schuller, "COVID-19 detection with a novel multi-type deep fusion method using breathing and coughing information," in *Proc. EMBC*, pp. 1840–1843, 2021.

[265] C. Brown, J. Chauhan, A. Grammenos, J. Han, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, and C. Mascolo, "Exploring Automatic Diagnosis of COVID-19 from Crowdsourced Respiratory Sound Data," in *Proceedings of the* 26^{th} *International Conference on Knowledge Discovery & Data Mining*, (Virtual Conference), pp. 3474–3484, ACM, 2020.

[266] S. Davis and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," *Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[267] A. Mallol-Ragolta, H. Cuesta, E. Gomez, and B. Schuller, "Multi-Type Outer Product-Based Fusion of Respiratory Sounds for Detecting COVID-19," in *Proc. INTERSPEECH*, (Incheon, Korea), pp. 2163–2167, 2022.

[268] N. K. Sharma, S. R. Chetupalli, D. Bhattacharya, D. Dutta, P. Mote, and S. Ganapathy, "The second dicova challenge: Dataset and performance analysis for diagnosis of COVID-19 using acoustics," in *Proc. ICASSP 2022*, (Singapore, Singapore), pp. 556–560, 2022.

[269] L. Piwek, D. A. Ellis, S. Andrews, and A. Joinson, "The rise of consumer health wearables: Promises and barriers," *PLOS Med.*, vol. 13, no. 2, p. 9 pages, 2018.

[270] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, "Deep Learning for IoT big data and streaming analytics: A survey," *IEEE Commun. Surv. Tutorials*, vol. 20, no. 4, pp. 2923–2960, 2018.

[271] C. Aytekin, X. Ni, F. Cricri, and E. Aksu, "Clustering and unsupervised anomaly detection with l2 normalized deep auto-encoder representations," in *Proceedings of the International Joint Conference on Neural Networks*, (Rio, Brazil), pp. 1–6, 2018.

[272] Z. Chen, C. K. Yeo, B. S. Lee, and C. T. Lau, "Autoencoder-based network anomaly detection," in *Proceedings of the Wireless Telecommunications Symposium*, (Phoenix, AZ), pp. 1–5, 2018.

[273] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, p. 3371–3408, 2010.

[274] S. Liu, J. Han, E. L. Puyal, S. Kontaxis, S. Sun, P. Locatelli, J. Dineley, F. B. Pokorny, G. D. Costa, L. Leocani, A. I. Guerrero, C. Nos, A. Zabalza, P. S. Sørensen, M. Buron, M. Magyari, Y. Ranjan, Z. Rashid, P. Conde, C. Stewart, A. A. Folarin, R. J. Dobson, R. Bailón, S. Vairavan, N. Cummins, V. A. Narayan, M. Hotopf, G. Comi, B. Schuller, and R.-C. Consortium, "Fitbeat: COVID-19 estimation based on wristband heart rate using a contrastive convolutional auto-encoder," *Pattern Recognition*, vol. 123, p. 108403, 2022.

[275] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-Level performance on ImageNet classification," *Proceedings of the International Conference on Computer Vision*, pp. 1026–1034, 2015.

[276] W. Shang, K. Sohn, D. Almeida, and H. Lee, "Understanding and Improving Convolutional Neural Networks via Concatenated Rectified Linear Units," in *Proceedings of the* 33rd *International Conference on Machine Learning*, (New York City, NY, USA), pp. 2217–2225, PMLR, 2016.

[277] G. Dalla Costa, L. Leocani, X. Montalban, A. I. Guerrero, P. S. Søorensen, M. Magyari, R. Dobson, N. Cummins, V. Narayan, M. Hotopf, G. Comi, and The RADAR-CNS consortium, "Real-time assessment of COVID-19 prevalence among multiple sclerosis patients: A multicenter European study," *Neurol. Sci.*, vol. 41, no. 7, pp. 1647–1650, 2020.

[278] S. Lauer, K. Grantz, Q. Bi, F. Jones, Q. Zheng, H. Meredith, A. Azman, N. Reich, and J. Lessler, "The incubation period of coronavirus disease 2019 (COVID-19) from publicly reported confirmed cases: Estimation and application," *Ann. Intern. Med.*, vol. 172, no. 9, pp. 577–582, 2020.

[279] J. Backer, D. Klinkenberg, and J. Wallinga, "Incubation period of 2019 novel coronavirus (2019-nCoV) infections among travellers from Wuhan, China, 20–28 January 2020," *Eurosurveillance*, vol. 25, no. 5, p. 6 pages, 2020.

[280] S. Sun, A. A. Folarin, Y. Ranjan, Z. Rashid, P. Conde, C. Stewart, N. Cummins, F. Matcham, G. Dalla Costa, S. Simblett, *et al.*, "Using smartphones and wearable devices to monitor behavioral changes during COVID-19," *Journal of medical Internet research*, vol. 22, no. 9, p. e19992, 2020.

[281] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, *et al.*, "Greedy layer-wise training of deep networks," *Proceedings of Advances in Neural Information Processing Systems*, vol. 19, pp. 153–160, 2006.

[282] M. Loey, G. Manogaran, M. H. N. Taha, and N. E. M. Khalifa, "A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic," *Measurement*, vol. 167, p. 108288, 2021.

[283] M. Loey, G. Manogaran, M. Taha, and N. E. Khalifa, "Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection," *Sustainable Cities and Society*, vol. 65, p. 102600, 2020.

[284] M. Kumar, K. Saluja, M. Sachdeva, S. Singh, and U. Ahuja, "Face mask detection using YOLO v3 and faster R-CNN models: COVID-19 environment," *Multimedia Tools and Applications*, 2021. 16 pages.

[285] J. Yu and W. Zhang, "Face mask wearing detection algorithm based on improved YOLO-v4," *Sensors*, vol. 21, no. 9, 2021. 21 pages.

[286] G. J. Chowdary, N. S. Punn, S. K. Sonbhadra, and S. Agarwal, "Face mask detection using transfer learning of inception v3," in *International Conference on Big Data Analytics*, (Sonipat, India), pp. 81–90, 2020.

[287] M. Magee, C. Lewis, G. Noffs, H. Reece, J. Chan, C. Zaga, C. Paynter, O. Birchall, S. Rojas Azocar, A. Ediriweera, M. Caverle, B. Schultz, and A. Vogel, "Effects of face masks on acoustic analysis and speech perception: Implications for peri-pandemic protocols," *The Journal of the Acoustical Society of America*, vol. 148, no. 6, pp. 3562–3568, 2020.

[288] D. D. Nguyen, P. McCabe, D. Thomas, A. Purcell, M. Doble, D. Novakovic, A. Chacon, and C. Madill, "Acoustic voice characteristics with and without wearing a facemask," *Scientific Reports*, vol. 11, 2021. 11 pages.

[289] R. M. Corey, U. Jones, and A. C. Singer, "Acoustic effects of medical, cloth, and transparent face masks on speech signals," *The Journal of the Acoustical Society of America*, vol. 148, no. 4, pp. 2371–2375, 2020.

[290] E. Mbunge, S. Simelane, S. G. Fashoto, B. Akinnuwesi, and A. S. Metfula, "Application of deep learning and machine learning models to detect COVID-19 face masks - A review," *Sustainable Operations and Computers*, vol. 2, pp. 235–245, 2021.

[291] B. Wang, J. Zheng, and C. L. P. Chen, "A survey on masked facial detection methods and datasets for fighting against COVID-19," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 3, pp. 323–343, 2022.

[292] H. Farman, T. Khan, Z. Khan, S. Habib, M. Islam, and A. Ammar, "Real-time face mask detection to ensure COVID-19 precautionary measures in the developing countries," *Applied Sciences*, vol. 12, no. 8, p. 3879, 2022.

[293] W. Hariri, "Efficient masked face recognition method during the COVID-19 pandemic," *Signal, image and video processing*, pp. 1–8, 2021.

[294] W. Boulila, A. Alzahem, A. Almoudi, M. Afifi, I. Alturki, and M. Driss, "A deep learning-based approach for real-time facemask detection," in *Proc. ICMLA*, pp. 1478–1481, 2021.

[295] H. Goyal, K. Sidana, C. Singh, A. Jain, and S. Jindal, "A real time face mask detection system using convolutional neural network," *Multimedia Tools and Applications*, pp. 1–17, 2022.

[296] M. M. Mohamed, M. A. Nessiem, A. Batliner, C. Bergler, S. Hantke, M. Schmitt, A. Baird, A. Mallol-Ragolta, V. Karas, S. Amiriparian, *et al.*, "Face mask recognition from audio: The MASC database and an overview on the mask challenge," *Pattern Recognition*, vol. 122, p. 108361, 2022.

[297] J. Szep and S. Hariri, "Paralinguistic classification of mask wearing by image classifiers and fusion," in *Proc. INTERSPEECH*, (Shanghai,China), pp. 2087–2091, 2020.

[298] R. K. Das and H. Li, "Classification of speech with and without face mask using acoustic features," in *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, (Virtual Conference), pp. 747–752, 2020.

[299] X. Xu, J. Deng, Z. Zhang, C. Wu, and B. Schuller, "Identifying surgical-mask speech using deep neural networks on low-level aggregation," in *Proc. ACM Symposium on Applied Computing*, (New York, NY, USA), pp. 580–585, 2021.

[300] B. Schuller, A. Batliner, C. Bergler, E.-M. Messner, A. Hamilton, S. Amiriparian, A. Baird, G. Rizos, M. Schmitt, L. Stappen, H. Baumeister, A. D. MacIntyre, and S. Hantke, "The INTERSPEECH 2020 computational paralinguistics challenge: Elderly emotion, breathing & masks," in *Proc. INTERSPEECH*, (Shanghai, China), pp. 2042–2046, 2020.

[301] A. S. Steffen Illium, Robert Muller and C. Linnhoff-Popien, "Surgical mask detection with convolutional neural networks and data augmentations on spectrograms," in *Proc. INTERSPEECH*, (Shanghai, China), pp. 2052–2056, 2020.

[302] T. Koike, K. Qian, B. W. Schuller, and Y. Yamamoto, "Learning higher representations from pre-trained deep models with data augmentation for the ComParE 2020

challenge mask task," in *Proc. INTERSPEECH*, (Shanghai, China), pp. 2047–2051, 2020.

[303] N.-C. Ristea and R. T. Ionescu, "Are you wearing a mask? Improving mask detection from speech using augmentation by cycle-consistent GANs," in *Proc. INTERSPEECH*, (Shanghai, China), 2020. 5 pages.

[304] S. Liu, A. Mallol-Ragolta, T. Yan, K. Qian, E. Parada-Cabaleiro, B. Hu, and B. W. Schuller, "Capturing time dynamics from speech using neural networks for surgical mask detection," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 8, pp. 4291–4302, 2022.

[305] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," in *Proc. North American Chapter of the Association for Computational Linguistics (NAACL)*, (New Orleans, Louisiana), pp. 464–468, 2018.

[306] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-XL: Attentive language models beyond a fixed-length context," in *Proc. Association for Computational Linguistics (ACL)*, (Florence, Italy), pp. 2978–2988, 2019.

[307] D. Chicco and G. Jurman, "The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC genomics*, vol. 21, no. 1, pp. 1–13, 2020.

[308] A. Mallol-Ragolta1, S. Liu, and B. W. Schuller, "The filtering effect of face masks in their detection from speech," in *Proc. Engineering in Medicine and Biology Society (EMBC)*, (Guadalajara, Mexico), 2021. 4 pages.

[309] A. Das, J. Li, R. Zhao, and Y. Gong, "Advancing connectionist temporal classification with attention modeling," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (Alberta, Canada), pp. 4769–4773, 2018.

[310] S. Liu, J. Jiao, Z. Zhao, J. Dineley, N. Cummins, and B. Schuller, "Hierarchical component-attention based speaker turn embedding for emotion recognition," in *Proc. International Joint Conference on Neural Networks (IJCNN)*, (Glasgow, UK), pp. 1–7, 2020.

[311] J. N. Mandrekar, "Receiver operating characteristic curve in diagnostic test assessment," *Journal of Thoracic Oncology*, vol. 5, no. 9, pp. 1315–1316, 2010.

[312] S. Leavy, G. Meaney, K. Wade, and D. Greene, "Mitigating gender bias in machine learning data sets," in *Proc. International Workshop on Algorithmic Bias in Search and Recommendation*, pp. 12–26, 2020.

[313] T. Wang, J. Zhao, M. Yatskar, K.-W. Chang, and V. Ordonez, "Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations," in *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5310–5319, 2019.

[314] F. Ertam, "An effective gender recognition approach using voice data via deeper LSTM networks," *Applied Acoustics*, vol. 156, pp. 351–358, 2019.

[315] K. P. Rao, M. C. S. Rao, and N. H. Chowdary, "An integrated approach to emotion recognition and gender classification," *Journal of Visual Communication and Image Representation*, vol. 60, pp. 339–345, 2019.

[316] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, "BYOL for audio: Self-supervised learning for general-purpose audio representation," in *Proc. IJCNN*, pp. 1–8, 2021.

[317] M. Karpiński, "The boundaries of language: Dealing with paralinguistic features," *Lingua Posnaniensis*, vol. 54, no. 2, pp. 37–54, 2013.

[318] A. Cabani, K. Hammoudi, H. Benhabiles, and M. Melkemi, "MaskedFace-Net: A dataset of correctly/incorrectly masked face images in the context of COVID-19," *Smart Health*, vol. 19, p. 100144, 2021.

[319] N. Fasfous, M. R. Vemparala, A. Frickenstein, L. Frickenstein, and W. Stechele, "BinaryCoP: Binary neural network-based COVID-19 face-mask wear and positioning predictor on edge devices," (Portland, OR, USA), pp. 108–115, 2021.

[320] L. Fritschi, A. Brown, R. Kim, D. Schwela, and S. Kephalopoulos, *Burden of disease from environmental noise: Quantification of healthy life years lost in Europe*. 2011.

[321] M. D. Seidman and R. T. Standring, "Noise and quality of life," *International Journal of Environmental Research and Public Health*, vol. 7, pp. 3730 – 3738, 2010.

[322] L. Goines and L. Hagler, "Noise pollution: A modem plague," *Southern Medical Journal*, vol. 100, no. 3, pp. 287–94, 2007.

[323] A. Varga and H. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Communication*, vol. 12, no. 3, pp. 247–251, 1993.

[324] R. Shenoy, P. P. Patwardhan, and G. G. Putraya, "Spatial audio enhancement apparatus," 2017. United States Patent 9769588.

[325] M. Kim and P. Smaragdis, "Collaborative audio enhancement using probabilistic latent component sharing," in *Proc. ICASSP*, (Vancouver, Canada), pp. 896–900, 2013.

[326] A. I. Klayman, "Audio enhancement system for use in a surround sound environment," 1999. United States Patent 5970152.

[327] S. Gustafsson, P. Jax, and P. Vary, "A novel psychoacoustically motivated audio enhancement algorithm preserving background noise characteristics," in *Proc. ICASSP*, (Seattle, WA, USA), pp. 397–400, 1998.

[328] E. Vincent, T. Virtanen, and S. Gannot, *Audio source separation and speech enhancement*. 2018.

[329] G. Yu, S. Mallat, and E. Bacry, "Audio denoising by time-frequency block thresholding," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1830–1839, 2008.

[330] B. Wright, E. Peters, U. Ettinger, E. Kuipers, and V. Kumari, “Understanding noise stress-induced cognitive impairment in healthy adults and its implications for schizophrenia,” *Noise and Health*, vol. 16, no. 70, pp. 166–176, 2014.

[331] L. Tzivian, M. Dlugaj, A. Winkler, G. Weinmayr, F. Hennig, K. B. Fuks, M. Vossoughi, T. Schikowski, C. Weimar, R. Erbel, *et al.*, “Long-term air pollution and traffic noise exposures and mild cognitive impairment in older adults: A cross-sectional analysis of the Heinz Nixdorf recall study,” *Environmental Health Perspectives*, vol. 124, no. 9, pp. 1361–1368, 2016.

[332] K. M. Prashanth and V. Sridhar, “The relationship between noise frequency components and physical, physiological and psychological effects of industrial workers,” *Noise and Health*, vol. 10, pp. 90–98, 2008.

[333] E. Parada-Cabaleiro, A. Batliner, A. Baird, and B. W. Schuller, “The perception of emotional cues by children in artificial background noise,” *International Journal of Speech Technology*, vol. 23, pp. 169–182, 2020.

[334] E. Parada-Cabaleiro, A. Baird, A. Batliner, N. Cummins, S. Hantke, and B. Schuller, “The perception of emotions in noisified nonsense speech,” in *Proc. INTERSPEECH*, (Stockholm, Sweden), pp. 3246–3250, 2017.

[335] P. H. Zannin, A. Calixto, F. B. Diniz, and J. A. Ferreira, “A survey of urban noise annoyance in a large Brazilian city: The importance of a subjective analysis in conjunction with an objective analysis,” *Environmental Impact Assessment Review*, vol. 23, no. 2, pp. 245–255, 2003.

[336] E. Kanjo, “Noisespy: A real-time mobile phone platform for urban noise monitoring and mapping,” *Mobile Networks and Applications*, vol. 15, no. 4, pp. 562–574, 2010.

[337] E. Atmaca, I. Peker, and A. Altin, “Industrial noise and its effects on humans.,” *Polish Journal of Environmental Studies*, vol. 14, no. 6, pp. 721–726, 2005.

[338] H. Miedema and C. Oudshoorn, “Annoyance from transportation noise: Relationships with exposure metrics DNL and DENL and their confidence intervals.,” *Environmental Health Perspectives*, vol. 109, no. 4, pp. 409–416, 2001.

[339] E. Healy, S. Yoho, Y. Wang, and D. Wang, “An algorithm to improve speech recognition in noise for hearing-impaired listeners,” *The Journal of the Acoustical Society of America*, vol. 134, no. 4, pp. 3029–3038, 2013.

[340] M. You, C. Chen, J. Bu, J. Liu, and J. Tao, “Emotion recognition from noisy speech,” in *Proc. ICME*, (Toronto, Canada), pp. 1653–1656, 2006.

[341] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, “Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs,” in *Proc. ICASSP*, (Salt Lake City, UT), pp. 749–752, 2001.

[342] J. Hansen and B. Pellom, “An effective quality evaluation protocol for speech enhancement algorithms,” in *Proc. ICSLP*, (Sydney, Australia), pp. 2819–2822, 1998.

[343] C. Taal, R. Hendriks, R. Heusdens, and J. Jensen, “A short-time objective intelligibility measure for time-frequency weighted noisy speech,” in *Proc. ICASSP*, (Dallas, TX), pp. 4214–4217, 2010.

[344] A. Gray and J. Markel, “Distance measures for speech processing,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 5, pp. 380–391, 1976.

[345] M. Freitag, S. Amiriparian, S. Pugachevskiy, N. Cummins, and B. Schuller, “auDeep: Unsupervised learning of representations from audio with deep recurrent neural networks,” *Journal of Machine Learning Research*, vol. 18, no. 173, pp. 1–5, 2018.

[346] M. Schmitt and B. Schuller, “openXBOW – introducing the Passau open-source crossmodal Bag-of-Words toolkit,” *Journal of Machine Learning Research*, vol. 18, no. 96, pp. 1–5, 2017.

[347] R. Kubichek, “Mel-cepstral distance measure for objective speech quality assessment,” in *Proc. PacRim*, (Victoria, Canada), pp. 125–128, 1993.

[348] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An ASR corpus based on public domain audio books,” in *Proc. ICASSP*, (Brisbane, Australia), pp. 5206–5210, 2015.

[349] G. Hu and D. Wang, “A tandem algorithm for pitch estimation and voiced speech segregation,” *IEEE/ACM Trans. Audio, Speech, Language Process*, vol. 18, no. 8, pp. 2067–2079, 2010.

[350] P. Tzirakis, G. Trigeorgis, M. A. Nicolaou, B. W. Schuller, and S. Zafeiriou, “End-to-end multimodal emotion recognition using deep neural networks,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1301–1309, 2017.

[351] S. B. Davis and P. Mermelstein, “Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 28, no. 4, pp. 357–366, 1980.

[352] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition,” *Proceedings of IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[353] Y. Wang and D. Wang, “Towards scaling up classification-based speech separation,” *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 21, no. 7, pp. 1381–1390, 2013.

[354] M. Seltzer, D. Yu, and Y. Wang, “An investigation of deep neural networks for noise robust speech recognition,” in *Proc. ICASSP*, (Vancouver, Canada), pp. 7398–7402, 2013.

[355] F. Weninger, J. R. Hershey, J. Le Roux, and B. Schuller, “Discriminatively trained recurrent neural networks for single-channel speech separation,” in *Proc. GlobalSIP*, pp. 577–581, 2014.

[356] Z. Zhang, F. Weninger, M. Wöllmer, J. Han, and B. Schuller, “Towards intoxicated speech recognition,” in *Proc. IJCNN*, (Anchorage, Alaska), pp. 1555–1559, 2017.

[357] J. Han, Z. Zhang, F. Ringeval, and B. Schuller, "Reconstruction-error-based learning for continuous emotion recognition in speech," in *Proc. ICASSP*, (New Orleans, LA, USA), pp. 2367–2371, 2017.

[358] Y.-S. Lee, C.-Y. Wang, S.-F. Wang, J.-C. Wang, and C.-H. Wu, "Fully complex deep neural network for phase-incorporating monaural source separation," in *Proc. ICASSP*, (New Orleans, LA, USA), pp. 281–285, 2017.

[359] S. Araki, T. Hayashi, M. Delcroix, M. Fujimoto, K. Takeda, and T. Nakatani, "Exploring multi-channel features for denoising-autoencoder-based speech enhancement," in *Proc. ICASSP*, (Brisbane, Australia), pp. 116–120, 2015.

[360] S. Samui, I. Chakrabarti, and S. K. Ghosh, "Deep recurrent neural network based monaural speech separation using recurrent temporal restricted boltzmann machines.," in *Proc. INTERSPEECH*, (Stockholm, Sweden), pp. 3622–3626, 2017.

[361] D. Wang and G. J. Brown, *Computational auditory scene analysis: Principles, algorithms, and applications.* Wiley-IEEE press, 2006.

[362] K. Zmolikova, M. Delcroix, K. Kinoshita, T. Higuchi, T. Nakatani, and J. Černockỳ, "Optimization of speaker-aware multichannel speech extraction with asr criterion," in *Proc. ICASSP*, (Alberta, Canada), pp. 6702–6706, 2018.

[363] B. King, I. F. Chen, Y. Vaizman, Y. Liu, R. Maas, H. K. Parthasarathi, and B. Hoffmeister, "Robust speech recognition via anchor word representations.," in *Proc. INTERSPEECH*, (Stockholm, Sweden), pp. 2471–2475, 2017.

[364] K. Zmolikova, M. Delcroix, K. Kinoshita, T. Higuchi, A. Ogawa, and T. Nakatani, "Speaker-Aware neural network based beamformer for speaker extraction in speech mixtures.," in *Interspeech*, (Stockholm, Sweden), pp. 2655–2659, 2017.

[365] M. Delcroix, K. Zmolikova, K. Kinoshita, A. Ogawa, and T. Nakatani, "Single channel target speaker extraction and recognition with speaker beam," in *Proc . ICASSP*, (Calgary, Alberta, Canada), pp. 5554–5558, 2018.

[366] K. Veselỳ, S. Watanabe, K. Žmolíková, M. Karafiát, L. Burget, and J. H. Černockỳ, "Sequence summarizing neural network for speaker adaptation," in *Proc. ICASSP*, (Shanghai, China), pp. 5315–5319, 2016.

[367] S. Shon, H. Tang, and J. R. Glass, "VoiceID loss: Speech enhancement for speaker verification," in *Proc. INTERSPEECH*, (Graz, Austria), pp. 2888–2892, 2019.

[368] A. H. Moore, P. P. Parada, and P. A. Naylor, "Speech enhancement for robust automatic speech recognition: Evaluation using a baseline system and instrumental measures," *Computer Speech & Language*, vol. 46, pp. 574–584, 2017.

[369] A. R. Avila, M. J. Alam, D. D. O'Shaughnessy, and T. H. Falk, "Investigating speech enhancement and perceptual quality for speech emotion recognition," in *Proc. INTERSPEECH*, (Hyderabad, India), pp. 3663–3667, 2018.

[370] D. Michelsanti and Z.-H. Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," in *Proc. INTERSPEECH*, (Stockholm, Sweden), pp. 2008–2012, 2017.

[371] S. Shon, H. Tang, and J. Glass, "VoiceID loss: Speech enhancement for speaker verification," in *Proc. INTERSPEECH*, (Graz, Austria), pp. 2888–2892, 2019.

[372] M. Kolbœk, Z. Tan, and J. Jensen, "Speech enhancement using long short-term memory based recurrent neural networks for noise robust speaker verification," in *Proc. SLT*, (San Diego, CA, USA), pp. 305–311, 2016.

[373] E. J. H. R. M. Bittner and J. P. Bello, "pysox: Leveraging the audio signal processing power of sox in python," in *Proc. ISMIR*, (New York City, NY, USA), 2018. 3 pages.

[374] K. Sekiguchi, A. Nugraha, Y. Bando, and K. Yoshii, "Fast multichannel source separation based on jointly diagonalizable spatial covariance matrices," in *Proc. EUSIPCO*, (Coruña, Spain), pp. 1–5, 2019.

[375] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. CVPR*, (Boston, MA, USA), pp. 1–9, 2015.

[376] S. Pascual, J. Serrà, and A. Bonafonte, "Towards generalized speech enhancement with generative adversarial networks," in *Proc. INTERSPEECH*, (Graz, Austria), pp. 1791–1795, 2019.

[377] J. Valin, "A hybrid DSP/deep learning approach to real-time full-band speech enhancement," in *Proc. MMSP*, (Vancouver, Canada), pp. 1–5, 2018.

[378] S. Bharitkar and C. Kyriakakis, "Selective signal cancellation for multiple-listener audio applications using eigenfilters," *IEEE Transactions on Multimedia*, vol. 5, no. 3, pp. 329–338, 2003.

[379] T. Wittkop and V. Hohmann, "Strategy-selective noise reduction for binaural digital hearing aids," *Speech Communication*, vol. 39, pp. 111–138, 2003.

[380] M. Büchler, S. Allegro, S. Launer, and N. Dillier, "Sound classification in hearing aids inspired by auditory scene analysis," *EURASIP Journal on Advances in Signal Processing*, vol. 2005, no. 18, pp. 2991–3002, 2005.

[381] B. Kuehn, A. Belkin, A. Swerdlow, T. Machmer, J. Beyerer, J. Beyerer, and K. Kroschel, "Knowledge-driven opto-acoustic scene analysis based on an object-oriented world modeling approach for humanoid robots," in *Proc. ISR/ROBOTIK*, (Munich, Germany), pp. 1–8, 2010.

[382] S. Aziz, M. Awais, T. Akram, U. Khan, M. Alhussein, and K. Aurangzeb, "Automatic scene recognition through acoustic classification for behavioral robotics," *Electronics*, vol. 8, no. 5, pp. 483–500, 2019.

[383] S. Chu, S. Narayanan, and C. J. Kuo, "Content analysis for acoustic environment classification in mobile robots," in *Proc. AAAI Fall Symp: Aurally Informed Performance*, (Arlington, VA), pp. 16–21, 2006.

[384] D. Fabry and J. Tchorz, "Results from a new hearing aid using acoustic scene analysis," *The Hearing Journal*, vol. 4, no. 58, pp. 30–36, 2005.

[385] N. D. Lane, P. Georgiev, and L. Qendro, "DeepEar: robust smartphone audio sensing in unconstrained acoustic environments using deep learning," in *Proc. UbiComp*, (Osaka, Japan), pp. 283–294, 2015.

[386] M. Won, H. Alsaadan, and Y. Eun, "Adaptive audio classification for smartphone in noisy car environment," in *Proc. ACM Multimedia*, (Mountain View, CA), pp. 1672–1679, 2017.

[387] M. Green and D. Murphy, "Environmental sound monitoring using machine learning on mobile devices," *Applied Acoustics*, vol. 159, 2019. 8 pages.

[388] A. Mesaros, T. Heittola, and T. Virtanen, "Acoustic scene classification in DCASE 2019 challenge: closed and open set classification and data mismatch setups," in *Proc. DCASE2019*, (New York, NY), pp. 164–168, 2019.

[389] A. Mesaros, A. Diment, B. Elizalde, T. Heittola, E. Vincent, B. Raj, and T. Virtanen, "Sound event detection in the DCASE 2017 Challenge," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 6, pp. 992–1006, 2019.

[390] A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. D. Plumbley, "Detection and classification of acoustic scenes and events: Outcome of the DCASE 2016 Challenge," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 2, pp. 379–393, 2018.

[391] H. Chen, Z. Liu, Z. Liu, P. Zhang, and Y. Yan, "Integrating the data augmentation scheme with various classifiers for acoustic scene modeling," in *Proc. DCASE2019*, (New York, NY), 2019. 5 pages.

[392] Y. Sakashita and M. Aono, "Acoustic scene classification by ensemble of spectrograms based on adaptive temporal divisions," in *Proc. DCASE2018*, (Woking, Surrey, UK), 2018. 5 pages.

[393] B. Gulmezoglu, A. Zankl, C. Tol, S. Islam, T. Eisenbarth, and B. Sunar, "Undermining user privacy on mobile devices using AI," in *Proc. Asia CCS*, (Auckland, New Zealand), pp. 214–227, 2019.

[394] I. H. Hann, K. L. Hui, S. Y. T. Lee, and I. P. Png, "Overcoming online information privacy concerns: An information-processing theory approach," *Journal of Management Information Systems*, vol. 24, no. 2, pp. 13–42, 2007.

[395] J. Benesty, S. Makino, and J. Chen, *Speech enhancement.* Springer Science & Business Media, 2005.

[396] N. Saleem and M. Khattak, "A review of supervised learning algorithms for single channel speech enhancement," *International Journal of Speech Technology*, vol. 22, no. 2019, pp. 1051–1075, 2019.

[397] A. Vafeiadis, D. Kalatzis, K. Votis, D. Giakoumis, D. Tzovaras, L. Chen, and R. Hamzaoui, "Acoustic scene classification: from a hybrid classifier to deep learning," in *Proc. DCASE2017*, (Munich, Germany), 2017. 5 pages.

[398] M. Valenti, S. Squartini, A. Diment, G. Parascandolo, and T. Virtanen, "A convolutional neural network approach for acoustic scene classification," in *Proc. IJCNN*, (Anchorage, Alaska), pp. 1547–1554, 2017.

[399] A. Moore, P. Peso Parada, and P. Naylor, "Speech enhancement for robust automatic speech recognition: Evaluation using a baseline system and instrumental measures," *Computer Speech & Language*, vol. 46, pp. 574–584, 2016.

[400] T. Zhang, J. Liang, and B. Ding, "Acoustic scene classification using deep cnn with fine-resolution feature," *Expert Systems with Applications*, vol. 143, no. 2020, 2020.

[401] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 23, no. 12, p. 2136–2147, 2015.

[402] G.-P. Yang, C.-I. Tuan, H. yi Lee, and L.-S. Lee, "Improved speech separation with time-and-frequency cross-domain joint embedding and clustering," in *Proc. INTERSPEECH*, 2019.

[403] N. Sharma, P. Krishnan, R. Kumar, S. Ramoji, S. R. Chetupalli, N. R., P. K. Ghosh, and S. Ganapathy, "Coswara – A Database of Breathing, Cough, and Voice Sounds for COVID-19 Diagnosis," in *Proceedings of Interspeech*, (Shanghai, China), pp. 4811–4815, ISCA, 2020.

[404] A. Hassan, I. Shahin, and M. B. Alsabek, "COVID-19 Detection System Using Recurrent Neural Networks," in *Proceedings of the International Conference on Communications, Computing, Cybersecurity, and Informatics*, (Sharjah, United Arab Emirates), IEEE, 2020. 5 pages.

[405] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems*, (Lake Tahoe, NV, USA), pp. 1097–1105, Curran Associates Inc., 2012.

[406] J. Han, K. Qian, M. Song, Z. Yang, Z. Ren, S. Liu, J. Liu, H. Zheng, W. Ji, T. Koike, X. Li, Z. Zhang, Y. Yamamoto, and B. W. Schuller, "An Early Study on Intelligent Analysis of Speech Under COVID-19: Severity, Sleep Quality, Fatigue, and Anxiety," in *Proceedings of Interspeech*, (Shanghai, China), pp. 4946–4950, ISCA, 2020.

[407] B. W. Schuller, A. Batliner, C. Bergler, C. Mascolo, J. Han, I. Lefter, H. Kaya, S. Amiriparian, A. Baird, L. Stappen, S. Ottl, M. Gerczuk, P. Tzirakis, C. Brown, J. Chauhan, A. Grammenos, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, L. J. M. Rothkrantz, J. Zwerts, J. Treep, and C. Kaandorp, "The INTERSPEECH 2021 Computational Paralinguistics Challenge: COVID-19 Cough, COVID-19 Speech, Escalation & Primates," in *Proceedings of Interspeech*, (Brno, Czech Republic), ISCA, 2021. To appear.

[408] M. Cohen-McFarlane, R. Goubran, and F. Knoefel, "Novel Coronavirus Cough Database: NoCoCoDa," *IEEE Access*, vol. 8, pp. 154087–154094, 2020.

[409] S. Ahmad, A. Tejuja, K. D. Newman, R. Zarychanski, and A. J. Seely, "Clinical review: A review and analysis of heart rate variability and the diagnosis and prognosis of infection," *Crit. Care*, vol. 13, no. 232, pp. 1–7, 2009.

[410] J. K. Triedman, R. J. Cohen, and J. P. Saul, "Mild hypovolemic stress alters autonomic modulation of heart rate," *Hypertension*, vol. 21, no. 2, pp. 236–247, 1993.

[411] D. Bonaduce, M. Petretta, F. Marciano, M. L. Vicario, C. Apicella, M. A. Rao, E. Nicolai, and M. Volpe, "Independent and incremental prognostic value of heart rate variability in patients with chronic heart failure," *American Heart Journal*, vol. 138, no. 2, pp. 273–284, 1999.

[412] J. Huang, S. M. Sopher, E. Leatham, S. Redwood, A. J. Camm, and J. C. Kaski, "Heart rate variability depression in patients with unstable angina," *American Heart Journal*, vol. 130, no. 4, pp. 772–779, 1995.

[413] Y. Ranjan, Z. Rashid, C. Stewart, P. Conde, M. Begale, D. Verbeeck, S. Boettcher, T. Hyve, R. Dobson, A. Folarin, and T. R.-C. Consortium, "RADAR-base: Open source mobile health platform for collecting, monitoring, and analyzing data using sensors, wearables, and mobile devices," *JMIR Mhealth and Uhealth*, vol. 7, no. 8, p. 13 pages, 2019.

[414] G. Zhu, J. Li, Z. Meng, Y. Yu, Y. Li, X. Tang, Y. Dong, G. Sun, R. Zhou, H. Wang, K. Wang, and W. Huang, "Learning from large-scale wearable device data for predicting epidemics trend of COVID-19," *Discrete Dyn. Nat. Soc.*, vol. 2020, 2020. 8 pages.

[415] R. Kamaleswaran, O. Sadan, P. Kandiah, Q. Li, J. M. Blum, C. M. Coopersmith, and T. G. Buchman, "Changes in non-linear and time-domain heart rate variability indices between critically ill COVID-19 and all-cause sepsis patients – a retrospective study." medRxiv, 2020.

[416] C. Menni, A. Valdes, M. Freidin, C. Sudre, L. Nguyen, D. Drew, S. Ganesh, T. Varsavsky, M. Cardoso, J. El-Sayed Moustafa, A. Visconti, P. Hysi, R. Bowyer, M. Mangino, M. Falchi, J. Wolf, S. Ourselin, A. Chan, C. Steves, and T. Spector, "Real-time tracking of self-reported symptoms to predict potential COVID-19," *Nat. Med.*, vol. 26, no. 7, pp. 1037–1040, 2020.

[417] D. J. Miller, J. V. Capodilupo, M. Lastella, C. Sargent, G. D. Roach, V. H. Lee, and E. R. Capodilupo, "Analyzing changes in respiratory rate to predict the risk of COVID-19 infection," *PLOS ONE*, vol. 15, no. 12, pp. 1–10, 2020.

[418] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[419] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[420] M. W. Gardner and S. Dorling, "Artificial neural networks (the multilayer perceptron) – A review of applications in the atmospheric sciences," *Atmospheric Environment*, vol. 32, no. 14-15, pp. 2627–2636, 1998.

[421] A. Khamparia, D. Gupta, N. G. Nguyen, A. Khanna, B. Pandey, and P. Tiwari, "Sound classification using convolutional neural network and tensor deep stacking network," *IEEE Access*, vol. 7, no. 99, pp. 7717–7727, 2019.

[422] C. Huang, Y. Wang, X. Li, L. Ren, J. Zhao, Y. Hu, L. Zhang, G. Fan, J. Xu, X. Gu, Z. Cheng, T. Yu, J. Xia, Y. Wei, W. Wu, X. Xie, W. Yin, H. Li, M. Liu, Y. Xiao, H. Gao, L. Guo, J. Xie, G. Wang, R. Jiang, Z. Gao, Q. Jin, J. Wang, and B. Cao, "Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China," *Lancet*, vol. 395, no. 10223, pp. 497–506, 2020.

[423] M. Chen, X. Shi, Y. Zhang, D. Wu, and M. Guizani, "Deep features learning for medical image analysis with convolutional autoencoder neural network," *IEEE Transactions on Big Data*, p. 10 pages, 2017.

[424] S. Chen, H. Liu, X. Zeng, S. Qian, J. Yu, and W. Guo, "Image classification based on convolutional denoising sparse autoencoder," *Mathematical Problems in Engineering*, vol. 2017, pp. 1–16, 2017.

[425] B. Du, W. Xiong, J. Wu, L. Zhang, L. Zhang, and D. Tao, "Stacked convolutional denoising auto-encoders for feature representation," *IEEE Transactions on Cybernetics*, vol. 47, no. 4, pp. 1017–1027, 2017.

[426] A. Ruiz-Garcia, M. Elshaw, A. Altahhan, and V. Palade, "Stacked deep convolutional auto-encoders for emotion recognition from facial expressions," in *Proceedings of the International Joint Conference on Neural Networks*, (Anchorage, AK), pp. 1586–1593, 2017.

[427] Q. Hu, M. Feng, L. Lai, and J. Pei, "Prediction of drug-likeness using deep autoencoder neural networks," *Frontiers in Genetics*, vol. 9, no. 585, p. 8 pages, 2018.

[428] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the International Conference on Machine Learning*, (Lille, France), pp. 448–456, 2015.

[429] J. F. Kenney, *Mathematics of Statistics*. D. Van Nostrand, 1939.

[430] S. Bektas and Y. Sisman, "The comparison of L1 and L2-norm minimization methods," *International Journal of the Physical Sciences*, vol. 5, no. 11, pp. 1721–1727, 2010.

[431] R. A. Horn and C. R. Johnson, "Norms for vectors and matrices," *Matrix analysis*, pp. 313–386, 1990.

[432] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products*. Academic press, 2014.

[433] H. Park, "An introduction to logistic regression: From basic concepts to interpretation with particular attention to nursing domain," *Journal of Korean Academy of Nursing*, vol. 43, no. 2, pp. 154–64, 2013.

[434] T.-T. Wong, "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation," *Pattern Recognit.*, vol. 48, no. 9, pp. 2839–2846, 2015.

[435] G. Costa, L. Leocani, X. Montalban, A. Guerrero, P. Soerensen, M. Magyari, R. Dobson, N. Cummins, V. Narayan, M. Hotopf, and G. Comi, "Real-time assessment of COVID-19 prevalence among multiple sclerosis patients: A multicenter European study," *Neurol. Sci.*, vol. 41, no. 7, pp. 1647–1650, 2020.

[436] M. Makkie, H. Huang, Y. Zhao, A. V. Vasilakos, and T. Liu, "Fast and scalable distributed deep convolutional autoencoder for fMRI big data analytics," *Neurocomputing*, vol. 325, pp. 20–30, 2019.

[437] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Deep convolutional autoencoder-based lossy image compression," in *Proceedings of the Picture Coding Symposium*, (San Francisco, CA), pp. 253–257, 2018.

[438] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proceedings of the Special Interest Group on Knowledge Discovery and Data Mining*, (Nova Scotia, Canada), pp. 665–674, 2017.

[439] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho, and H. Chen, "Deep autoencoding Gaussian mixture model for unsupervised anomaly detection," in *Proceedings of the International Conference on Learning Representations*, (Vancouver, Canada), 2018. 19 pages.

[440] Y. Zhao, B. Deng, C. Shen, Y. Liu, H. Lu, and X.-S. Hua, "Spatio-temporal autoencoder for video anomaly detection," in *Proceedings of the ACM International Conference on Multimedia*, (New York, NY), pp. 1933–1941, 2017.

[441] C. Aytekin, X. Ni, F. Cricri, and E. Aksu, "Clustering and unsupervised anomaly detection with l2 normalized deep auto-encoder representations," in *Proceedings of the International Joint Conference on Neural Networks*, (Rio, Brazil), pp. 1–6, 2018.

[442] J. Han, K. Qian, M. Song, Z. Yang, Z. Ren, S. Liu, J. Liu, H. Zheng, W. Ji, T. Koike, X. Li, Z. Zhang, Y. Yamamoto, and B. Schuller, "An early study on intelligent analysis of speech under COVID-19: Severity, sleep quality, fatigue, and anxiety," in *Proceedings of INTERSPEECH*, (Shanghai, China), pp. 4946–4950, 2020.

[443] B. Schuller, D. Schuller, K. Qian, J. Liu, H. Zheng, and X. Li, "COVID-19 and computer audition: An overview on what speech & sound analysis could contribute in the SARS-CoV-2 Corona crisis," in *arxiv*, 2020. 6 pages.

[444] J. Backer, D. Klinkenberg, and J. Wallinga, "Incubation period of 2019 novel coronavirus (2019-nCoV) infections among travellers from Wuhan, China, 20–28 January 2020," *Eurosurveillance*, vol. 25, no. 5, 2020.

[445] Y. Ding, X. Zhang, and J. Tang, "A noisy sparse convolution neural network based on stacked auto-encoders," in *Proceedings of the International Conference on Systems, Man and Cybernetics*, (Banff, Canada), pp. 3457–3461, 2017.

[446] X. Bai, C. Fang, Y. Zhou, S. Bai, Z. Liu, L. Xia, Q. Chen, Y. Xu, T. Xia, S. Gong, X. Xudong, D. Song, R. Du, C. Zhou, C. Chen, D. Nie, L. Qin, and W. Chen,

"Predicting COVID-19 Malignant Progression with AI Techniques," *Medrxiv*, p. 29 pages, 2020.

[447] S. N. Karmali, A. Sciusco, S. M. May, and G. L. Ackland, "Heart rate variability in critical care medicine: A systematic review," *Intensive Care Medicine Experimental*, vol. 5, no. 1, pp. 1–15, 2017.

[448] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: A review," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1–38, 2021.

[449] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, (Fort Lauderdale, FL, USA), pp. 315–323, 2011.

[450] H. C. Ates, A. K. Yetisen, F. Güder, and C. Dincer, "Wearable devices for the detection of COVID-19," *Nature Electronics*, vol. 4, no. 1, pp. 13–14, 2021.

[451] S. Leavy, "Gender bias in artificial intelligence: The need for diversity and gender theory in machine learning," in *Proc. of International Workshop on Gender Equality in Software Engineering*, pp. 14–16, 2018.

[452] A. B. Melchiorre, N. Rekabsaz, E. Parada-Cabaleiro, S. Brandl, O. Lesota, and M. Schedl, "Investigating gender fairness of recommendation algorithms in the music domain," *Information Processing & Management*, vol. 58, no. 5, 2021. 17 pages.

[453] G. Costantini, E. Parada-Cabaleiro, D. Casali, and V. Cesarini, "The emotion probe: On the universality of cross-linguistic and cross-gender speech emotion recognition via machine learning," *Sensors*, vol. 22, no. 7, 2022.

[454] S. R. Mallinas, J. K. Maner, and E. A. Plant, "What factors underlie attitudes regarding protective mask use during the COVID-19 pandemic?," *Personality and Individual Differences*, p. 111038, 2021.

[455] J. Lang, W. W. Erickson, and Z. Jing-Schmidt, "# maskon!# maskoff! digital polarization of mask-wearing in the united states during COVID-19," *PloS ONE*, vol. 16, no. 4, p. e0250817, 2021.

[456] E. Parada-Cabaleiro, G. Costantini, A. Batliner, M. Schmitt, and B. W. Schuller, "DEMoS: An Italian emotional speech corpus: Elicitation methods, machine learning, and perception," *Language, Resources, and Evaluation*, vol. 54, pp. 341–383, 2020.

[457] S. Taylor and G. J. Asmundson, "Negative attitudes about facemasks during the COVID-19 pandemic: The dual importance of perceived ineffectiveness and psychological reactance," *PLoS ONE*, vol. 16, no. 2, p. e0246317, 2021.

[458] T. Bhasin, C. Butcher, E. Gordon, M. Hallward, and R. LeFebvre, "Does Karen wear a mask? The gendering of COVID-19 masking rhetoric," *International Journal of Sociology and Social Policy*, 2020.

[459] Y. LeCun, Y. Bengio, and G. Hinton, "An attentive survey of attention models," *ACM Trans. Intell. Syst. Technol.*, vol. 1, no. 1, 2021. 33 pages.

[460] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.

[461] S. P. Kaur and V. Gupta, "COVID-19 vaccine: A comprehensive status report," *Virus research*, vol. 288, p. 198114, 2020.

[462] S. Loomba, A. de Figueiredo, S. J. Piatek, K. de Graaf, and H. J. Larson, "Measuring the impact of COVID-19 vaccine misinformation on vaccination intent in the UK and USA," *Nature human behaviour*, vol. 5, no. 3, pp. 337–348, 2021.

[463] J. P. Moore and P. A. Offit, "SARS-CoV-2 vaccines and the growing threat of viral variants," *JAMA*, vol. 325, no. 9, pp. 821–822, 2021.

[464] R. Rubin, "COVID-19 vaccines vs variants determining how much immunity is enough," *JAMA*, vol. 325, no. 13, pp. 1241–1243, 2021.

[465] A. A. Dawood, "Mutated COVID-19 may foretell a great risk for mankind in the future," *New Microbes and New Infections*, vol. 35, p. 100673, 2020.

[466] L. Bandiera, G. Pavar, G. Pisetta, S. Otomo, E. Mangano, J. R. Seckl, P. Digard, E. Molinari, F. Menolascina, and I. M. Viola, "Face coverings and respiratory tract droplet dispersion," *Royal Society open science*, vol. 7, no. 12, p. 201663, 2020.

[467] J. Howard, A. Huang, Z. Li, Z. Tufekci, V. Zdimal, H.-M. van der Westhuizen, A. von Delft, A. Price, L. Fridman, L.-H. Tang, V. Tang, G. L. Watson, C. E. Bax, R. Shaikh, F. Questier, D. Hernandez, L. F. Chu, C. M. Ramirez, and A. W. Rimoin, "An evidence review of face masks against COVID-19," *Proceedings of the National Academy of Sciences*, vol. 118, no. 4, 2021.

[468] I. M. Viola, B. Peterson, G. Pisetta, G. Pavar, H. Akhtar, F. Menoloascina, E. Mangano, K. E. Dunn, R. Gabl, A. Nila, *et al.*, "Face coverings, aerosol dispersion and mitigation of virus transmission risk," *IEEE Open Journal of Engineering in Medicine and Biology*, vol. 2, pp. 26–35, 2021.

[469] D. K. Chu, E. A. Akl, S. Duda, K. Solo, S. Yaacoub, H. J. Schünemann, A. El-harakeh, A. Bognanni, T. Lotfi, M. Loeb, *et al.*, "Physical distancing, face masks, and eye protection to prevent person-to-person transmission of SARS-CoV-2 and COVID-19: A systematic review and meta-analysis," *The lancet*, vol. 395, no. 10242, pp. 1973–1987, 2020.

[470] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. Weiss, and K. Wilson, "CNN architectures for large-scale audio classification," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 131–135, 2017.

[471] B. Wang, D. Zhao, C. Lioma, Q. Li, P. Zhang, and J. G. Simonsen, "Encoding word order in complex embeddings," in *Proc İCLR*, (Addis Ababa, Ethiopia), 2020.

[472] R. Turner, "Covid-19 and aerosol transmission: up in the air," *BMJ*, vol. 372, 2021.

[473] L. Morawska, J. W. Tang, W. Bahnfleth, P. M. Bluyssen, A. Boerstra, G. Buonanno, J. Cao, S. Dancer, A. Floto, F. Franchimon, C. Haworth, J. Hogeling, C. Isaxon, J. L. Jimenez, J. Kurnitski, Y. Li, M. Loomans, G. Marks, L. C. Marr, L. Mazzarella, A. K. Melikov, S. Miller, D. K. Milton, W. Nazaroff, P. V. Nielsen, C. Noakes, J. Peccia, X. Querol, C. Sekhar, O. Seppänen, S. ichi Tanabe, R. Tellier, K. W. Tham, P. Wargocki, A. Wierzbicka, and M. Yao, "How can airborne transmission of COVID-19 indoors be minimised?," *Environment International*, vol. 142, p. 105832, 2020.

[474] R. Zhang, Y. Li, A. L. Zhang, Y. Wang, and M. J. Molina, "Identifying airborne transmission as the dominant route for the spread of COVID-19," *Proceedings of the National Academy of Sciences*, vol. 117, no. 26, pp. 14857–14863, 2020.

[475] T. Greenhalgh, J. Jimenez, K. Prather, Z. Tufekci, D. Fisman, and R. Schooley, "Ten scientific reasons in support of airborne transmission of SARS-CoV-2," *The Lancet*, vol. 397, pp. 1603–1605, 2021.

[476] M. Klompas, M. A. Baker, and C. Rhee, "Airborne transmission of SARS-CoV-2: Theoretical considerations and available evidence," *JAMA*, vol. 324, no. 5, pp. 441–442, 2020.

[477] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic modeling," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, pp. 1469–1477, 2015.

[478] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[479] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014.

[480] G. Douzas and F. Bacao, "Effective data generation for imbalanced learning using conditional generative adversarial networks," *Expert Systems with applications*, vol. 91, pp. 464–471, 2018.

[481] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Proc. INTERSPEECH*, (Dresden, Germany), pp. 3586–3589, 2015.

[482] S. G. Koolagudi and K. S. Rao, "Emotion recognition from speech: a review," *International Journal of Speech Technology*, vol. 15, no. 2, pp. 99–117, 2012.

[483] M. El Ayadi, M. S. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognition*, vol. 44, no. 3, pp. 572–587, 2011.

[484] E. Kramer, "Elimination of verbal cues in judgments of emotion from voice," *The Journal of Abnormal and Social Psychology*, vol. 68, no. 4, pp. 390–396, 1964.

[485] G. Fairbanks and W. Pronovost, "Vocal pitch during simulated emotion," *Science*, vol. 88, pp. 382–383, 1938.

[486] C. N. Anagnostopoulos, T. Iliou, and I. Giannoukos, "Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011," *Artificial Intelligence Review*, vol. 43, no. 2, pp. 155–177, 2015.

[487] M. E. Ayadi, M. S. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognition*, vol. 44, pp. 572–587, 2011.

[488] Z. Ling, S. Kang, H. Zen, A. Senior, M. Schuster, X. Qian, H. Meng, and L. Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 35–52, 2015.

[489] J. Han, Z. Zhang, G. Keren, and B. W. Schuller, "Emotion recognition in speech with latent discriminative representations learning," *Acta Acustica united with Acustica*, vol. 104, pp. 737–740, 2018.

[490] H. Gunes and B. W. Schuller, "Categorical and dimensional affect analysis in continuous input: Current trends and future directions," *Image Vision Computing*, vol. 31, no. 2, pp. 120–136, 2013.

[491] B. W. Schuller, G. Rigoll, and M. Lang, "Hidden markov model-based speech emotion recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (Hong Kong, China), pp. 401–404, 2003.

[492] F. Eyben, K. R. Scherer, B. W. Schuller, J. Sundberg, E. André, C. Busso, L. Y. Devillers, J. Epps, P. Laukka, S. S. Narayanan, and K. P. Truong, "The Geneva minimalistic acoustic parameter set (GeMAPS) for voice research and affective computing," *IEEE Transactions on Affective Computing*, vol. 7, no. 2, pp. 190–202, 2016.

[493] L. Tian, J. D. Moore, and C. Lai, "Emotion recognition in spontaneous and acted dialogues," in *Proc. International Conference on Affective Computing and Intelligent Interaction (ACII)*, (Xian, China), pp. 698–704, 2015.

[494] G. Keren, F. Ringeval, E. Marchi, and B. W. Schuller, "End-to-end learning for dimensional emotion recognition from physiological signals," in *Proc. IEEE International Conference on Multimedia and Expo (ICME)*, (Hong Kong), pp. 985–990, 2017.

[495] K. Han, D. Yu, and I. Tashev, "Speech emotion recognition using deep neural network and extreme learning machine," in *Proc. INTERSPEECH*, (Singapore, Singapore), pp. 223–227, 2014.

[496] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen, J. Chen, J. Chen, Z. Chen, M. Chrzanowski, A. Coates, G. Diamos, K. Ding, N. Du, E. Elsen, J. Engel, W. Fang, L. Fan, C. Fougner, L. Gao, C. Gong, A. Hannun, T. Han, L. V. Johannes, B. Jiang, C. Ju, B. Jun, P. LeGresley, L. Lin, J. Liu, Y. Liu, W. Li, X. Li, D. Ma, S. Narang, A. Ng, S. Ozair, Y. Peng, R. Prenger, S. Qian, Z. Quan, J. Raiman, V. Rao, S. Satheesh, D. Seetapun, S. Sengupta, K. Srinet, A. Sriram, H. Tang, L. Tang, C. Wang, J. Wang, K. Wang, Y. Wang, Z. Wang, Z. Wang, S. Wu, L. Wei, B. Xiao, W. Xie, Y. Xie, D. Yogatama, B. Yuan, J. Zhan, and Z. Zhu, "Deep speech 2: End-to-end speech recognition in English and Mandarin," in *Proc. ICML*, (New York, NY, USA), p. 173–182, 2016.

[497] T. Meenpal, A. Balakrishnan, and A. Verma, "Facial mask detection using semantic segmentation," in *Proc. ICCCS*, (Singapore, Singapore), pp. 1–5, 2019.

[498] S. Ruder, “An overview of multi-task learning in deep neural networks,” *arXiv preprint arXiv:1706.05098*, 2017.