Zhou and Wan
EURASIP Journal on Audio, Speech, and Music Processing (2023) 2023:27
https://doi.org/10.1186/s13636-023-00292-9

EURASIP Journal on Audio, Speech, and Music Processing

METHODOLOGY

Open Access

Dual-branch attention module-based network with parameter sharing for joint sound event detection and localization

Yuting Zhou[1] and Hongjie Wan[1*]

Abstract

The goal of sound event detection and localization (SELD) is to identify each individual sound event class and its activity time from a piece of audio, while estimating its spatial location at the time of activity. Conformer combines the advantages of convolutional layers and Transformer, which is effective in tasks such as speech recognition. However, it achieves high performance relying on complex network structure and a large number of computations. In the SELD task of this paper, we propose to use an encoder with a simpler network structure, called the dual-branch attention module (DBAM). The module is improved based on the conformer using two parallel branches of attention and convolution, which can model both global and local contextual information. We also blend low-level and high-level features of the localization task. In addition, we add soft parameter sharing to the joint SELD network, which can efficiently exploit the potential relationship between the two subtasks, SED and DOA. The proposed method can effectively detect two sound events with overlapping occurrence in the same time period. We experimented with the open dataset DCASE 2020 task 3 proving that the proposed method achieves better SELD performance than the baseline model. Furthermore, we conducted ablation experiments for verifying the effectiveness of the dual-branch attention module and soft parameter sharing.

Keywords Sound event detection and localization, Conformer, Attention mechanism, Multi-task learning, Soft parameter sharing

1 Introduction

Sound event detection (SED) is an important research direction in non-speech signal recognition. In our daily life, we often hear dogs barking, birds chirping, car sirens, sirens, footsteps, broken glass, etc. All these signals can be called sound events. The sound source localization (SSL) technique involves the measurement of sound signals using multiple microphones at different location points in the environment. Since the sound signals arrive at each microphone with different degrees of delay, the algorithm is used to process the measured sound signals and thus obtain the direction of arrival (including azimuth and elevation angle) and the distance relative to the microphone of the sound source point, etc. In this paper, we consider only the relative position of the active sound event, i.e., the direction of arrival (DOA) estimate, without considering its relative distance. Sound event detection and localization (SELD) is the process of identifying the sound event associated with a label from the audio, detecting the activity time (start and offset time of the event), and estimate its spatial location.

The application of audio signals as auxiliary signals is more efficient to work with, compared to some situations where it is not convenient to capture video. Effective SELD methods can describe the temporal and spatial representation of sound events, which have a wide range

*Correspondence:
Hongjie Wan
wanhj@mail.buct.edu.cn
[1] Information Engineering Dept, Beijing University of Chemical Technology, No.15 North Third Ring Road East, Beijing 100029, China

Springer Open

© The Author(s) 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

of applications in many fields. In the field of autonomous driving, it enables smart cars to detect the sound of horns, sirens, etc., in the environment and respond to them correctly [1]. In industry, it can be used as an aid for fault checking of large machinery and equipment [2]. In smart cities and smart homes, it can be used for machine listening and audio monitoring [3, 4]. In addition, it can identify, track, and suppress ambient noise to enhance speech quality for video conferencing or automatic speech recognition (ASR) [5, 6].

SED focuses on the type of sound and its corresponding temporal activity, while DOA estimation focuses on the spatial location information of the sound source. The SELD task serves as a joint task of the SED and DOA estimates. With the rapid development and great breakthroughs in machine learning, SELD have gained more performance improvements. The neural network-based approach is applied to SED, which effectively improves the accuracy of detection. It also overcomes the shortcomings of the traditional parameter-based DOA method, with robustness in reverberation and low signal-to-noise scenarios.

The processing of each frame by CNN [7] is based on a finite temporal frequency range around which the neighborhood information can be captured. RNN [8] can effectively exploit the temporal dependence in pairs of audio sequences, but it is difficult to learn in parallel and capture the interactions of long-range sequences. For the SELD task, the authors used CNNs to jointly predict audio content classes and spatial locations [9]. Adavanne et al. trained the features of SED and DOA together in a CRNN network, consisting of the convolutional module and the BiGRU module [10]. The two-stage strategy for SELD proposed by Cao et al. achieved a significant improvement [11]. This method detects sound event types at first, uses a representation of migration learning in order to extract DOA estimation features, and finally trains the SED mask to predict the direction of arrival of the sound events. These methods still in essence view detection and localization as two separate tasks and do not make good use of the connection between them. Moreover, when two sound events of the same type are in different DOA directions, they will not be detected.

Transformer [12] was first proposed in the field of natural language processing. It has also been introduced to computer vision in recent years and has shown revolutionary performance improvements. The detection accuracy of sound events can be improved by using context-sensitive information [13]. In [14], Transformer model was introduced to the field of speech recognition and Speech-transformer was proposed, which is a repetition-free seq-seq model relying entirely on the attention mechanism to learn positional dependencies. Self-attention mechanisms are very effective for long-range global context modeling. However, local feature information of audio sequences has an indispensable role in sound event detection. Conformer [15] is a variant of Transformer that combines this global and local contextual information into a unified single-branch structural model by incorporating a convolution module. However, the structure of conformer is very complex and a large number of network parameters need to be used for training.

Inspired by the branching architectures of Lite Transformer [16] and Branchformer [17], we propose a dual-branch model combining self-attentive and convolutional modules in the SELD task. The proposed module has a simple structure and can also take full advantage of the two branches, combining global and local contextual information of the audio sequences. In this work, our contributions are as follows.

- First, the dual-branch attention module is proposed in this paper as an attention mechanism for track separation in the joint SELD task. Moreover, by reducing the overall computation and model size through a parallel dual-branch architecture, better SELD performance can be achieved with fewer parameters.
- Second, improved localization performance is achieved by blending the low-level and high-level features in the localization sub-network.
- Third, the parameters of the SED and DOA sub-networks are shared through the cross-stitch module to obtain the best optimization of the joint task.

In Sect.2, the components of the joint SELD network and the detailed structure of proposed DBAM are described in detail. Section 2.2.4 gives the experimental setup, including the dataset, settings, and evaluation metrics. Section 2.5 shows the comparison and analysis of the results of the comparative and ablation experiments. The conclusions and future work are given in Sect. 4.

2 The proposed method

In this section, the joint SELD network, which performs both SED and DOA subtasks, is described in detail. The joint SELD network we proposed consists of two subnetworks, as shown in Fig. 1. One subnetwork learns the features of the SED task, the other subnetwork learns the features of the DOA task. These two subnetworks have independent parts and intersecting parts. The independent parts are the respective conv blocks, the dual-branch attention module, and the fully connected layers. The intersecting parts are the cross-stitch modules in the soft parameter sharing. First, the original audio data

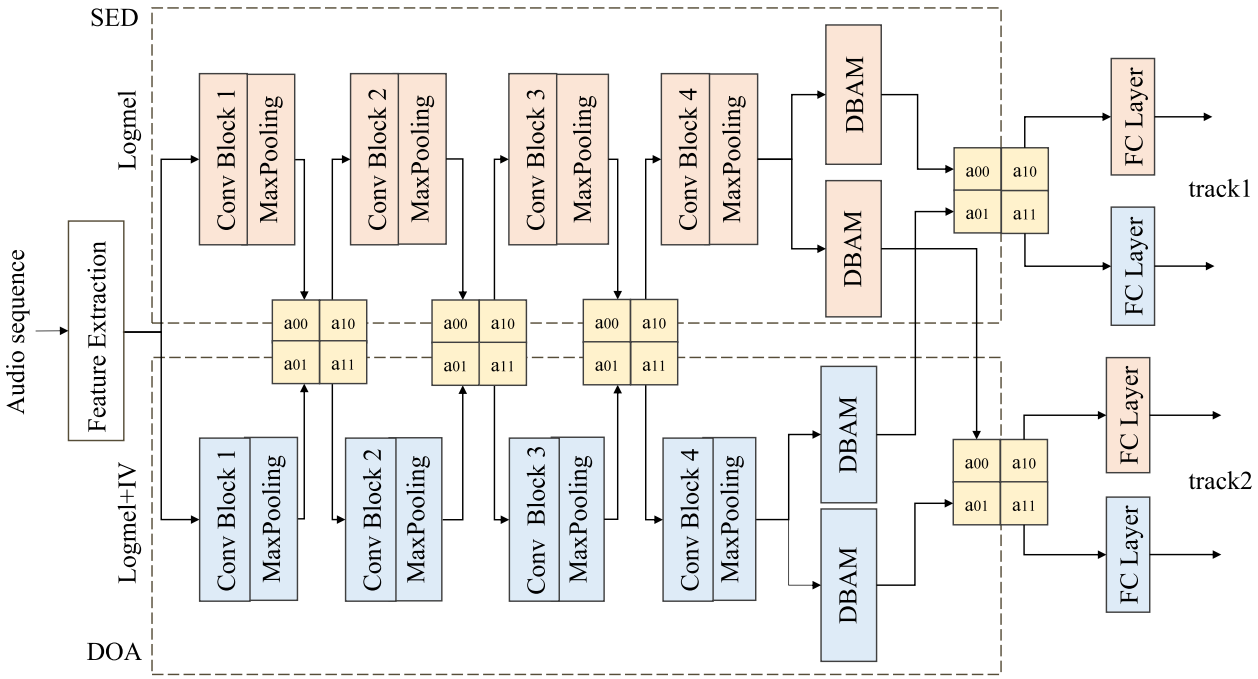

Fig. 1 Overall structure of the proposed joint SELD network. The red rectangle indicates the task of SED, the blue indicates the task of DOA, and the yellow indicates the cross-stitch module for soft parameter sharing

features are extracted and the respective features are input accordingly to the respective convolutional operations in the conv blocks of the SED and DOA tasks. Next, the global and local contextual correlations in the audio sequences are modeled by the dual-branch attention module. The stacked DBAMs are used to separate the tracks of the two overlapping sound events. In addition, the shared features of two independent subnetworks are linearly modeled using the parameter sharing module after each conv block and DBAM. Finally, separate SED and DOA subtasks are executed through two fully connected layers in parallel to output sound event categories and relative position information. The detailed description of each module in Fig. 1 is given in the following subsections.

2.1 Conv blocks

The audio signal features can be efficiently extracted by convolutional layers and the perceptual field can be expanded by stacked convolutional modules. Each subnetwork uses four conv blocks. The detailed network structure of the proposed conv blocks is shown in Fig. 2. Each conv block has two 2D convolutional layers with a kernel size of 3×3. All convolutional layers are followed by adding batch normalization and ReLU activation layers. Pooling operations are used after the conv blocks. We use maximum pooling for

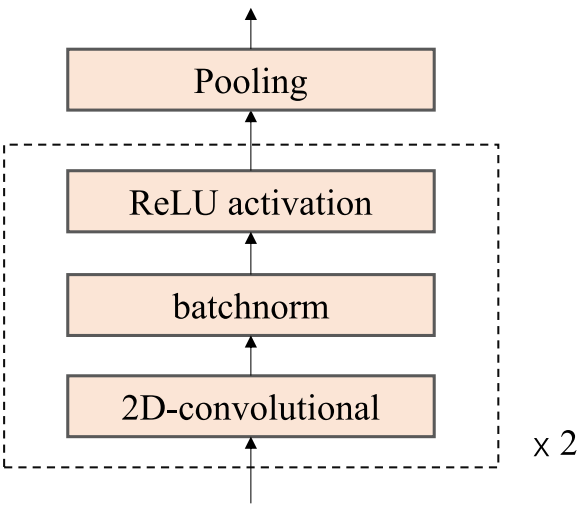

Fig. 2 The structure of the conv block (1–4)

the first three pooling in each subnetwork. The purpose of this is to use maximum pooling to efficiently obtain the most obvious features and remove invalid information. The dilation parameter of the maximum pooling is set to 1, i.e., the element step in the window is 1. The last conv block uses average pooling because it is needed to combine the information of the high-level features deep in the network, which can help the classifier to classify.

2.2 Dual-branch attention module

Transformer and conformer models have successively achieved excellent performance in speech separation and speech recognition tasks. In sound event detection, feature extraction is important to improve audio classification and localization performance. Although Transformer is capable of context-aware modeling of audio sequences, it eschews RNNs and CNNs, which also makes its ability to integrate local features limited. Conformer uses deep separable convolution to enhance the performance of Transformer, but its model structure is complex. We improve on the conformer model by using N stacked dual-branch attention module (DBAM) to model the global and local contextual information of the feature sequence and to separate the tracks of two overlapping sound events. It is described in detail in this section.

Inspired by the paper [16, 17], we use the attention module with two branches, consisting of the attention branch, the gated convolutional branch and a residual connection. The detailed structure is shown in Fig. 3a. The convolutional branch can effectively extract local detail features, while the attentional branch is good at capturing long-range global contexts. Therefore, our proposed model can well distinguish the spectral features of different sound events and thus improve the performance of the network. These two branches share the features of the input, but they respectively focus on global and local information and are combined in the fusion unit. The original input of the DBAM is also added to the final output as a residual connection. The residual connection solves the problem of gradient explosion and gradient disappearance that may arise during training in deep networks. Only the encoder part is used in our proposed method, since our task does not require the recovery of the audio sequence. In addition, we do not use the position encoding part of the encoder because it is not suitable for acoustic sequences.

2.2.1 Attention branch

The structure of the attention branch is shown in the left branch of Fig. 3a. First, we use layer normalization of the input features, then the multi-headed self-attention module [12] can capture global information of the sequence, followed by a dropout operation. The multi-headed attention module MHSA in the conformer is attempt to model both global and local contexts. The MHSA in the proposed model focuses only on the global context in the attention branch, leaving the capture of local information to the convolution branch.

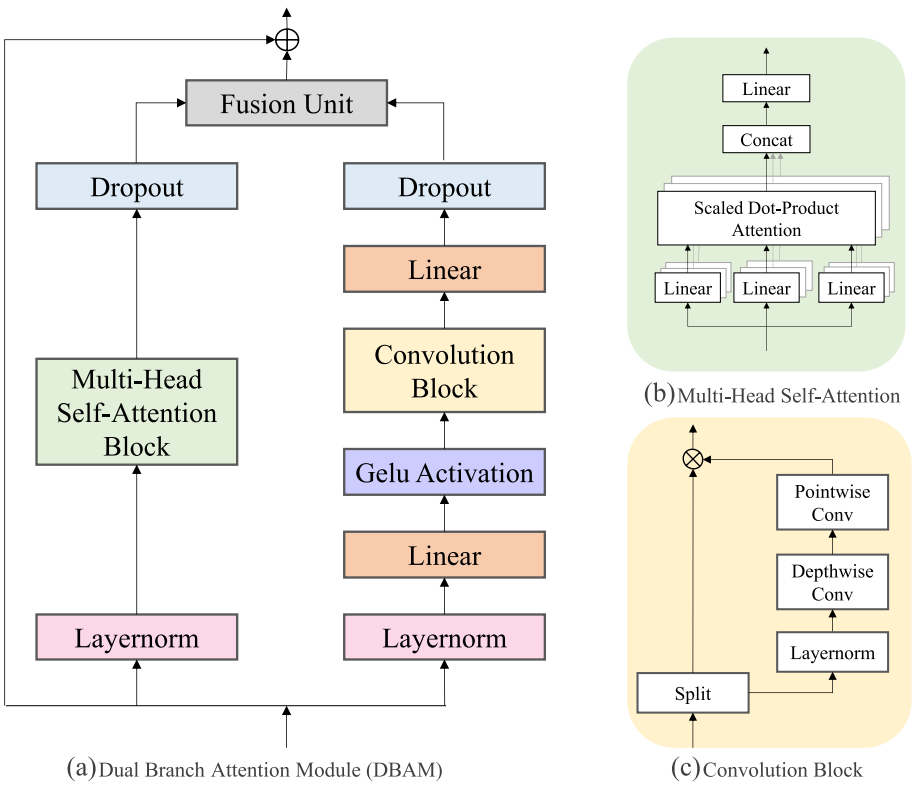

Fig. 3 The overall architecture of the DBAM and the detail of multi-head self-attention and convolution block therein. **a** Dual-branch attention module. **b** Multi-head self-attention. **c** Convolution block

Some sound events can be easily confused. For example, the cat's purr is very similar to baby's cry. It is difficult to distinguish them manually or using spectral information in the frequency domain. The attention mechanism can further distinguish confusable sounds by learning the importance of each element in a sequence and then assigning a series of attention coefficients. Multi-headed self-attention is essentially multiple independent attention computations and connects the features they extract as an integrated effect, as shown in Fig. 3b.

In the MHSA block, the input matrix (X) is mapped into queries (Q), keys (K), and values (V) by linear transformation h times, as described in Eq. (1). Then, the dot product between each query and key is calculated and divided by a constant. Normalization is performed using the Softmax function to obtain the weights of the values. As shown in Eq. (2), the attention of each head is a weighted combination of values. Finally, the attentions of all heads are concatenated and linear projection is performed to obtain the final output.

$$Q_i = XW_i^Q, K_i = XW_i^K, V_i = XW_i^V \tag{1}$$

$$H_i = f(\frac{Q_i K_i^T}{\sqrt{d}})V_i \tag{2}$$

$$M(Q,K,V) = C(H_1,...,H_h)W^O \tag{3}$$

where the input $X \in R^{L\times d}$ is a sequence of length L, feature dimension d. $I \in R^{d\times d}$ is the linear transformation matrix ($i=1, 2, \ldots h$), and h is the number of attention heads. $Q_i, K_i, V_i \in R^{L\times d/h}$ are the queries, keys, and values of the mapping, respectively. $W_i^Q W_i^K W_i^V \in R^{d\times d/h}$ denote the *ith* linear transformation matrices of the Q, K, and V, respectively. In addition, f denotes the Softmax function, C denotes the concatenation operation, and d is the number of columns of the Q, K matrix, i.e., the vector dimension. H represents the output of single-headed attention, and M is the output of multi-headed attention.

2.2.2 Convolutional branch

The convolutional branch structure in the proposed model is shown in the left branch of Fig. 3a to model local contextual information. This branch is firstly layer-normalized as the attention branch. Referring to the Macaron structure used in the conformer model, we use two linear layers before and after the convolution block. A linear layer and the Gaussian error linear unit (GeLU) [18] activation function are used to act as the feedforward part. Then comes a convolutional block with gating and another linear layer. Finally, the dropout operation is used as the last layer of the convolutional branch, which helps to regularize the network. Suppose X represents the input to the convolutional branch, the output Y of this branch can be calculated by the intermediate $\widehat{Z}$, Z, and Z' as

$$\widehat{Z} = \mathrm{M}_{\mathrm{layernorm}}(X) \tag{4}$$

$$Z = GeLU(\widehat{Z}P) \tag{5}$$

$$Z' = \mathrm{M}_{\mathrm{conv}}(Z) \tag{6}$$

$$Y = Z'Q \tag{7}$$

where GeLU represents the layer normalization process, the GeLU activation, and the convolution block, respectively. In addition, $P \in R^{d\times d_{hidden}}$, $Q \in R^{(d_{hidden}/2)\times d}$ indicates the linear transformation of the linear layer, and d_{hidden} indicates the hidden layer dimension.

As shown in Fig. 3c, the convolution block in DBAM consists of a linear gating containing layer normalization, a depthwise convolutional layer, and a pointwise convolutional layer. Such a convolution block with linear gating is *more simply* structured *if* compared to the convolution module in conformer, and no nonlinear activation function is used. Referring to depthwise separable convolution, we use the combination of depthwise convolution and pointwise convolution, which has a lower number of parameters and computational cost compared to traditional convolutional layers. Suppose the number of input channels is C_{in}, the number of output channels is C_{out}, and the size of feature map is $M \times N$. Then, the computation of ordinary convolution is $k^2 \times C_{in} \times M \times N \times C_{out}$, and the computation of depth-separable convolution is $C_{in} \times M \times N \times (k^2 + C_{out})$. It can be seen that the depth-separable convolution can reduce the computation.

The split operation is a mean splitting of the input sequence along the feature dimension. The Z_1 sequence resulting from the split is layer normalized and convolved with the layer, and then multiplied with the elements of the Z_2 sequence.

$$\overline{Z_1} = \mathrm{M}_{\mathrm{DWConv}}(\mathrm{M}_{\mathrm{Layernorm}}(Z_1)) \tag{8}$$

$$\widetilde{Z_1} = \mathrm{M}_{\mathrm{PWConv}}(\overline{Z_1}) \tag{9}$$

$$Z' = \widetilde{Z_1} \otimes Z_2 \tag{10}$$

where $\mathrm{M}_{\mathrm{DWConv}}$ denotes the process of deep convolution, $\mathrm{M}_{\mathrm{PWConv}}$ denotes the process of pointwise convolution, and $\otimes$ denotes the multiplication of elements.

2.2.3 Fusion unit

We use weighted averaging in the fusion unit to combine the attention branch and the convolution branch. The weighted averaging method is more efficient and easier to train for the combination of the two branches than the normally used direct tandem connection. The weights of the two branches can indicate how the global and local contextual information is used in training. The output sequences of the attention and convolution branches are respectively multiplied by the dynamically generated weights of the model, then added together. The output Y of the fusion module is a weighted average expressed by the Eq. (11).

$$\mathrm{Y} = \omega_{att}\mathrm{Y}_{att} + \omega_{conv}\mathrm{Y}_{conv} \tag{11}$$

where $\mathrm{Y_{att}}$,$\mathrm{Y_{conv}} \in R^{l\times d}$ are the output sequences of attentional and convolutional branches, respectively. Y captures the global and local dependencies. And ω_{att},$\omega_{\mathrm{conv}} \in R^{1\times d}$ represent the branch weights obtained by normalization using Softmax.

$$\omega_{att}, \omega_{conv} = \mathrm{softmax}(W_{att}Y_{att}, W_{conv}Y_{conv}) \tag{12}$$

2.2.4 Computational complexity analysis

We analyze the computational complexity of each component in DBAM in this section. Assume that the input sequence length is L and the feature dimension is d. The parallel computation of the two branches reduces the complexity of the proposed DBAM. The number of model parameters is compared in Sect. 4.2.

In the attention branch, the complexity of the linear mapping of Q, K and V is O (Ld^2). In the scaled dot-product part, the computation of QK^T in Eq. (2) obtains an $L\times L$ matrix, which determines the complexity of the self-attentive module is O(L^2d). Since the attention of multiple heads is computed in parallel, the complexity can be equivalent to that of self-attention.

In the convolution branch, the split operation reduces the dimension of the sequence Z_1, Z_2 by half to $d_{hidden}/2$. d_{hidden} is the dimension of the hidden layer, which is usually larger than d (e.g., in our experiments $d=512$ and $d_{hidden}=2048$). This approach reduces the computational cost of the convolutional branch. The complexity of the first linear layer is $\mathrm{O}(L \times d \times d_{hidden})$, and the second linear layer is $\mathrm{O}(L \times d \times d_{hidden}/2)$. In addition, the complexity of the convolution block is $\mathrm{O}(L \times k^2 \times d^2_{hidden}/2)$, where k is the size of the convolution kernel.

2.2.5 The connection of DBAM stacked in DOA subnetwork

In conventional neural networks, most networks send only the features output from the last layer to the pooling layer. This approach ended up utilizing only high-level features but lacked a description of the low-level texture features of the audio signal. It has been shown in a number of studies that high-level semantic features are crucial for classification, but for localization, texture features extracted by low-level networks are essential (Fig. 4).

2.3 Soft parameter sharing

Multi-task learning (MTL) is a joint machine learning method that learns by multiple tasks in parallel so that the results affect each other. MTL can alleviate the overfitting of the model to some extent and enhance the generalization ability of the model, which leads to better results. SELD has two parallel sub-tasks SED and DOA, while the performance of both is influenced by each other, so it can be considered as an MTL problem. Hard parameter sharing and soft parameter sharing are two methods to implement MTL. Hard parameter sharing can be applied to all hidden layers of all tasks, while retaining the output layers associated with the task. Due to its sharing for most parameters, it reduces the risk of model overfitting. Soft parameter sharing, in which each subtask consists of its own model and parameters and there are links between different feature layers, is the method favored by modern research priorities.

SELDnet [10] is a hard parameter sharing model as a baseline method for SELD. The features of SED and DOA are trained together using the CRNN network as an advanced feature representation module. Then, two fully connected parallel branches are used to predict the sound event type and spatial location respectively. Different from previous challenges DCASE 2020 uses a joint metric of SED and DOA [20], i.e., location-dependent detection and class-dependent localization. Soft parameter sharing can take advantage of the association between SED and DOA to learn better models. Better SELD performance will be achieved if the advantages of soft parameter sharing and Transformer structure are fully utilized.

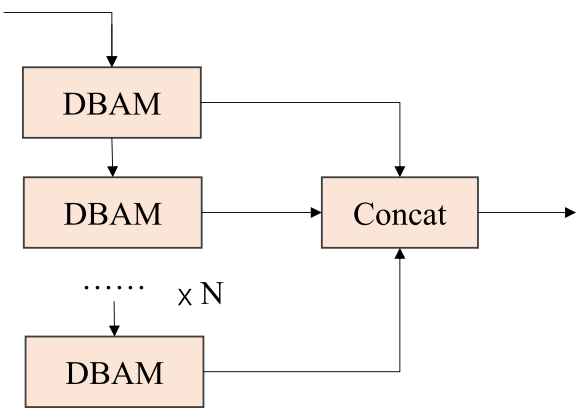

Fig. 4 The connection method of DBAM stacked in DOA subnetwork

There are some effective multi-task learning mechanisms in DNNs, such as fully adaptive feature sharing [21], cross-stitch networks [22], a joint many-task model [23], weighting losses with uncertainty [24], and sluice networks [25].

In [22], two independent networks are connected with a soft parameter sharing like the cross-stitch network. The cross-stitch module uses linear combinations of shared features for the purpose of coming to learn the best linear combinations for multiple tasks. We train two separate sub-networks for SED and DOA respectively and connect them by cross-stitch module. This way the tasks can supervise each other and decide how much to share. The cross-stitch module exchanges shared features of useful information to learn the optimal linear combination of multiple tasks. The performance loss of two subtasks can be avoided. In this paper, soft parameter sharing is used after pooling for convolutional layers and after linear normalization for DBAM (Fig. 5).

The equation for the cross-stitch module of SED and DOA is described as

$$\begin{bmatrix} \hat{x}^{SED} \\ \hat{x}^{DOA} \end{bmatrix} = \alpha \begin{bmatrix} x^{SED} \\ x^{DOA} \end{bmatrix} = \begin{bmatrix} \alpha_{00} & \alpha_{01} \\ \alpha_{10} & \alpha_{11} \end{bmatrix} \begin{bmatrix} x^{SED} \\ x^{DOA} \end{bmatrix} \tag{13}$$

where α is a 2×2 matrix and α_{ij} denotes the learnable parameters.

The parameters of cross-stitch are set between [0,1] in order to ensure the stability of learning and that the values of the inputs and outputs are on the same order of magnitude.

2.4 Track output format

The output format of SELDnet is to predict the probability of all sound events and the corresponding locations at frame t. Only one position per event can be predicted and the same event with multiple positions cannot be detected.

The track output format [26] is at t frames and detects only one event and the corresponding position per track and reduces the model capacity. In the absence of overlapping sound events, only one track is used. However, the track output format becomes very useful in case of overlap. Multiple tracks can output the same overlapping event in separate locations without being affected by the sound event type. Therefore, it can solve the problem of isomorphic overlap.

Affected by the above factors, we use track prediction to solve overlapping event scenarios. In this paper, the number of tracks is set to 2 because we use the dataset containing at most two cases of overlapping sound events. In addition, the track output format estimates only M positions, not the positions of all K events (whether they are active or not). $M \ll K$, reducing the need for network parameters and required data size.

The track output format gives rise to track alignment problem, which can be solved by using permutation-invariant training (PIT) [27]. In the SELD task, the frame-level PIT is used to assign the possible event trajectories in each frame during training. The lowest PIT loss is then selected for backpropagation. Optimal matching of sound event type labels to tracks improves SED and DOA frame-by-frame prediction performance.

2.5 Loss function

We need to detect and classify multiple sound event classes in audio, and therefore the SED task is considered as a multi-label classification problem. A cross-entropy (CE) loss is used as the loss function for SED. The calculation equation is as follows.

$$loss_{SED} = -\frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T} y_{nt} \cdot \log(p(\hat{y}_{nt})) \tag{14}$$

where y_{nt} and $\hat{y}_{nt}$ are the probability reference and prediction of the nth sound event being active in the tth frame, respectively.

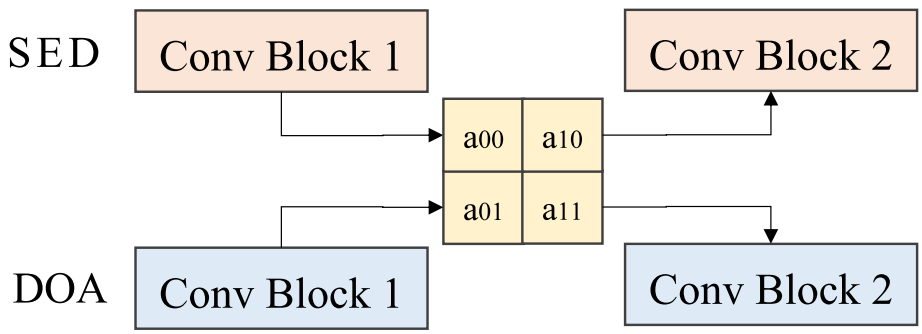

Fig. 5 The detailed structure of a cross-stitch module applied between the two sub-networks

DOA estimation is usually considered as a multiple output regression task with mean square error (MSE) loss. The equation is described as

$$loss_{DOA} = \sum_{n=1}^{N} \sum_{t=1}^{T} \left\| \hat{d}_{nt} - d_{nt} \right\|^2 \quad (15)$$

where d_{nt} and $\hat{d}_{nt}$ are the DOA reference and prediction of the nth sound event at frame t, respectively.

The joint SELD network in this paper uses a joint loss function, which is a combination of SED loss and DOA loss. The joint optimization is performed during training using weights to obtain more accurate classification and localization performance. The joint loss function can be described by Eq. (16).

$$loss_{SELD} = \omega loss_{SED} + (1 - \omega) loss_{DOA} \quad (16)$$

where ω is an adjustable parameter and was set to 0.5 in this experiment.

3 Experimental setup

3.1 Datasets

The proposed method is experimented on the TAU-NIGENS Spatial Sound Event 2020 [28]. The dataset contains two different spatial sound formats, first-order Ambisonics (FOA) and tetrahedral microphone array (MIC), both of which are four channels. We performed separate feature extraction for the FOA and MIC datasets. For the MIC dataset, log Mel features and generalized cross-correlation (GCC-PATH) features were extracted. For the FOA dataset, log Mel features and 3-channel acoustic intensity vector (IV) features were extracted.

The sound signal in the microphone array arrives at different microphones at different times, and a time delay is generated. The target signals received by each microphone of the microphone array come from the same sound source, and there is a strong correlation between the signals of each channel. By calculating the generalized mutual correlation function between two channels, the time delay estimation difference between two microphones is obtained, and thus the arrival direction of the sound signal is estimated. Therefore, the generalized cross-correlation feature is applicable to the audio in microphone array format. The first-order Ambisonic (FOA) format is a multi-channel surround sound that accurately records information about the location of sound in space. FOA format audio has four channels representing four different directions: center, left and right, front and back, and top and bottom in a three-dimensional 360-degree range. The same Ambisonic channels have the same spatial characteristics, independent of the recording settings. For FOA format audio cannot be located using the time delay estimation method. The GCC feature is not applicable in the audio in FOA format. We choose sound intensity vectors which contain spatial phase information of the sound in FOA format.

Each dataset contains 600 60-s multichannel audio divided into six folds and sampled at 24 kHz. The spatialized sound event categories are 14 in total, e.g., alarm, crying baby, running engine, female scream, footsteps, and ringing phone. The azimuthal angle $\phi \in [-180, 180)$ and the elevation angle $\theta \in [-45, 45]$, both in degrees. There may be at most two overlapping sound events in time and space. In addition, the audio contains both static and moving sound events. Moving sound events have three possible angular velocities: slow (10 degrees/sec), medium (20 degrees/sec), or fast (40 degrees/sec). The signal-to-noise ratio (SNR) of ambient noise varies from noiseless (30 dB) to noisy (6 dB).

3.2 Training setup

First, FFT was performed using a 1024-point Hann window with a 600-point frame shift. For both the training and test sets, the audio is split into segments of 4 s in length, without overlapping parts. The Adam optimizer was used. The initial learning rate is set to 0.0005, and the learning rate is adjusted at 80 epoch intervals by a gamma factor of 0.1. The batch size is set to 32 for training and 64 for prediction. The threshold of SED was set to 0.5 for binarization of

Table 1 The SELD performance of the proposed method and comparison models for FOA dataset

Models	$ER_{20^\circ}\downarrow$	$F_{20^\circ}\uparrow$	$LE_{CD}\downarrow$	$LR_{CD}\uparrow$	SELD↓
DCASE2020task3	0.580	51.3%	18.3°	69.9%	0.367
SELDnet	0.720	37.4%	22.8°	60.7%	0.466
Two-stage network	0.399	67.5%	14.8°	73.8%	0.267
Cao_Surrey	0.363	71.2%	13.3°	81.1%	0.229
Nguyen_NTU	0.360	71.9%	12.1°	**82.7%**	0.220
Proposed method	**0.347**	**74.0%**	**9.30°**	80.2%	**0.214**

The "↓" after the evaluation metric in the table indicates that the lower the metric is better, and "↑" indicates that the higher the metric is better

Table 2 The SELD performance of the proposed method and comparison models for MIC dataset

Models	$ER_{20^\circ}\downarrow$	$F_{20^\circ}\uparrow$	$LE_{CD}\downarrow$	$LR_{CD}\uparrow$	SELD↓
DCASE2020task3	0.690	41.3%	23.1°	62.4%	0.445
SELDnet	0.780	31.4%	27.3°	59.0%	0.503
Two-stage network	0.518	59.5%	14.2°	69.6%	0.371
Cao_Surrey	0.471	61.5%	16.7°	75.4%	0.298
Nguyen_NTU	0.360	71.4%	12.1°	**82.0%**	**0.223**
Proposed method	**0.364**	**71.9%**	**11.5°**	80.3%	0.226

the predictions. There were at most two overlapping events in the audio, so the number of tracks M was set to 2. The experiments in Sect. 2.5 are trained on the dev dataset and tested on the eval dataset.

3.3 Evaluation metrics

The evaluation metrics we used were those of the DCASE 2020 SELD Challenge [20]. This differs from the previous evaluation metrics by adding a correlation between the two tasks. Four evaluation metrics are used: the SED-related error rate (ER_{20°), the F_1 score of the SED (F_{20°), the SED-dependent location error (LE_{CD}), and the frame recall of the location (LR_{CD}). The effective SELD network should have lower ER_{20° and LE_{CD}, higher F_{20° score and LR_{CD}.

F_1 score and error rate are classical SED metrics, but they were added with a condition related to location. If the sound event is considered correctly detected, it means that its category is correctly predicted and the deviation of the predicted DOA from the reference value is less than T°. In the evaluation, this threshold was set to T = 20°. Thus, these two evaluation metrics are denoted as F_{20° and ER_{20°.

$$F_{20^\circ} = \frac{2PR}{P+R} = \frac{2TP}{2TP+FP+FN} \tag{17}$$

Table 3 Test with overlapping sound events

	$ER_{20^\circ}\downarrow$	$F_{20^\circ}\uparrow$	$LE_{CD}\downarrow$	$LR_{CD}\uparrow$	SELD↓
overlap = 1	0.273	79.4%	7.097°	81.7%	0.175
overlap = 2	0.515	62.5%	12.362°	70.6%	0.313
overlap = 1&2	0.347	74.0%	9.304°	80.2%	0.214

$$ER_{20^\circ} = \frac{D+I+S}{N} \tag{18}$$

where

$$S = \min(FN, FP) \tag{19}$$

$$D = \max(0, FN - FP) \tag{20}$$

$$I = \max(0, FP - FN) \tag{21}$$

The true positive (TP) is when both reference and prediction events are active. The false positive (FP or insert I) is when the reference is inactive and the prediction is active. The false negative (FN or delete D) is when the reference is active but the prediction is inactive. A TP and a TN occurring at the same time are counted as a substitution error (S), and N is the total number of sound events in the ground truth.

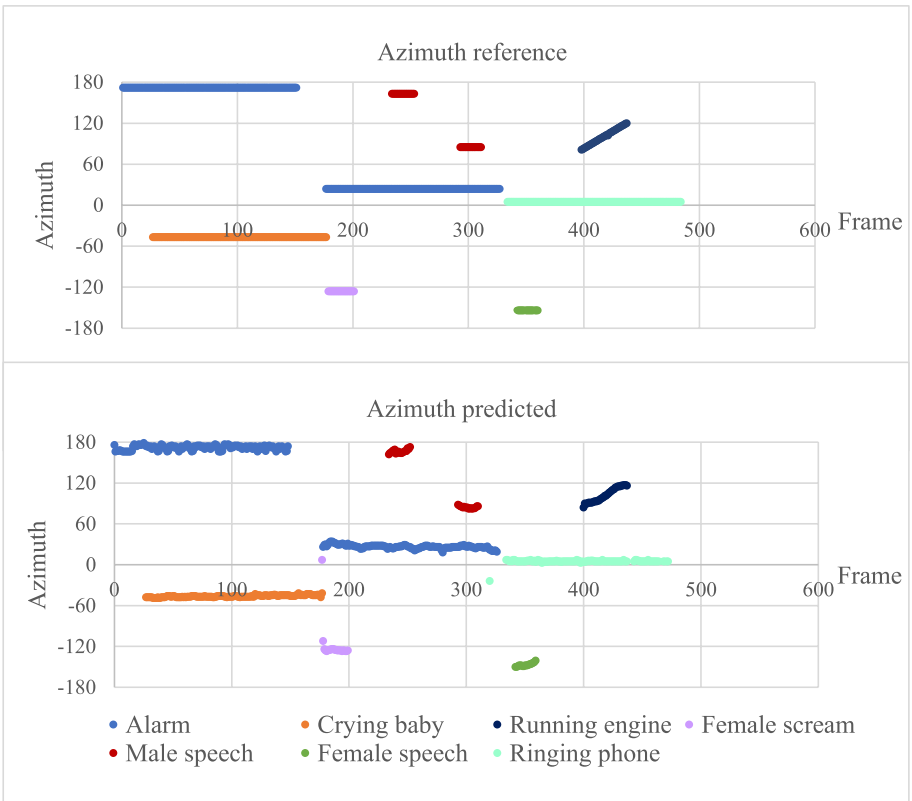

Fig. 6 The reference and prediction of azimuth in audio Mix146 for eval dataset

The other two metrics focus on the localization component and are dependent on the classification, which means they are calculated only in each class and not in all outputs. The localization error (LE_{CD}) represents the average angular distance between prediction and reference for the same class. The locality recall metric (LR_{CD}) indicates represents the number of true positives detected for location estimates in a class as a percentage of the total class instances. No threshold is used for above two localization metrics.

$$LE_{CD} = \arccos(u_{ref} \cdot u_{pre}) \tag{22}$$

where u_{ref} and u_{pre} denote the position vectors of the reference and predicted sound event, respectively. The subscripts are the classification correlations.

In addition, $SELD_{score}$ were calculated to aggregate all four metrics, which were calculated as

$$SELD_{score} = \frac{ER_{20^\circ} + (1 - F_{20^\circ}) + LE_{CD}/\pi + (1 - LR_{CD})}{4} \tag{23}$$

$SELD_{score}$ is the overall performance metric of the SELD task. The model with the smallest $SELD_{score}$ in the evaluation is selected as the best model.

4 Experimental results and analysis

4.1 Comparison with other methods

To effectively validate the joint SLED network we used, we compared the proposed network model with the approaches of SELDnet, two-stage networks, the official baseline model in DCASE2020task3, Cao_Surrey, and Nguyen_NTU based on datasets in both FOA and MIC formats. For the proposed method, we used one DBAM module and two DBAM modules, respectively. In addition, the number of headers in the multi-head self-attention module is set to 8. All models were trained and tested under the same conditions.

In Table 1, "DCASE2020task3" represents the official baseline model under challenge. Cao_Surrey [29] and Nguyen_NTU [30] are the two methods that rank higher in the challenge. As can be seen in Table 1, the proposed method on the FOA format dataset has better results than the official baseline model of DCASE2020task3, the SELDnet model, the two-stage network, and Cao_Surrey in all metrics. The method of Nguyen_NTU divides the orientation estimation into azimuth and elevation angles before using a sequence matching network. The Nguyen_NTU method has a higher localization recall and more false positives. However, for the sound event localization task, we mainly focus on improving the localization

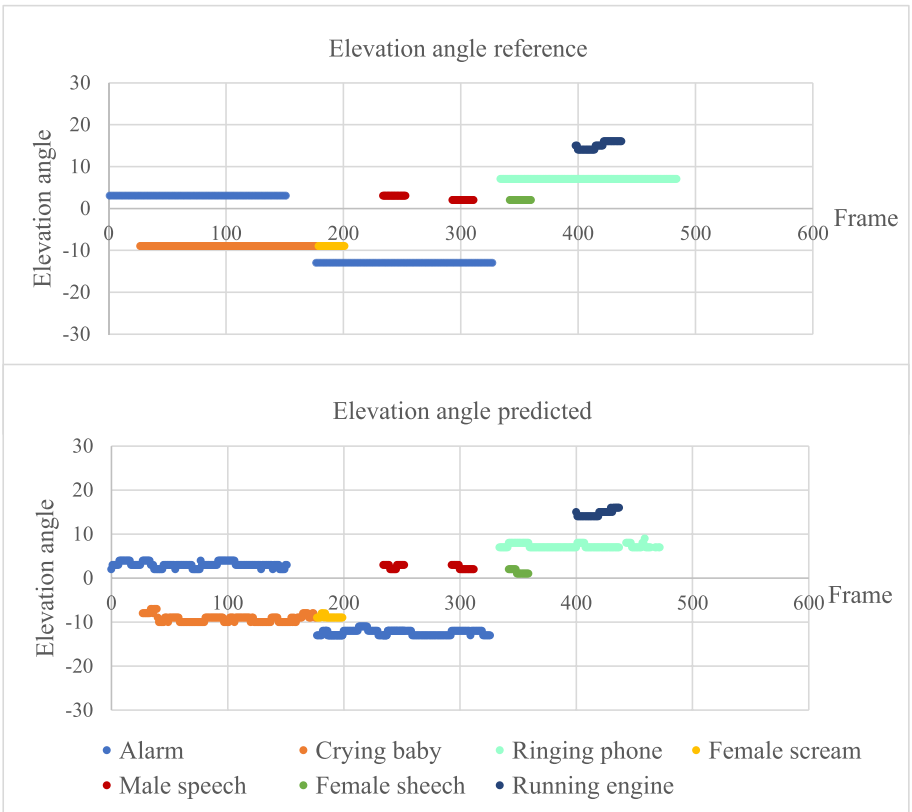

Fig. 7 The reference and prediction of elevation angle in audio Mix146 for eval dataset

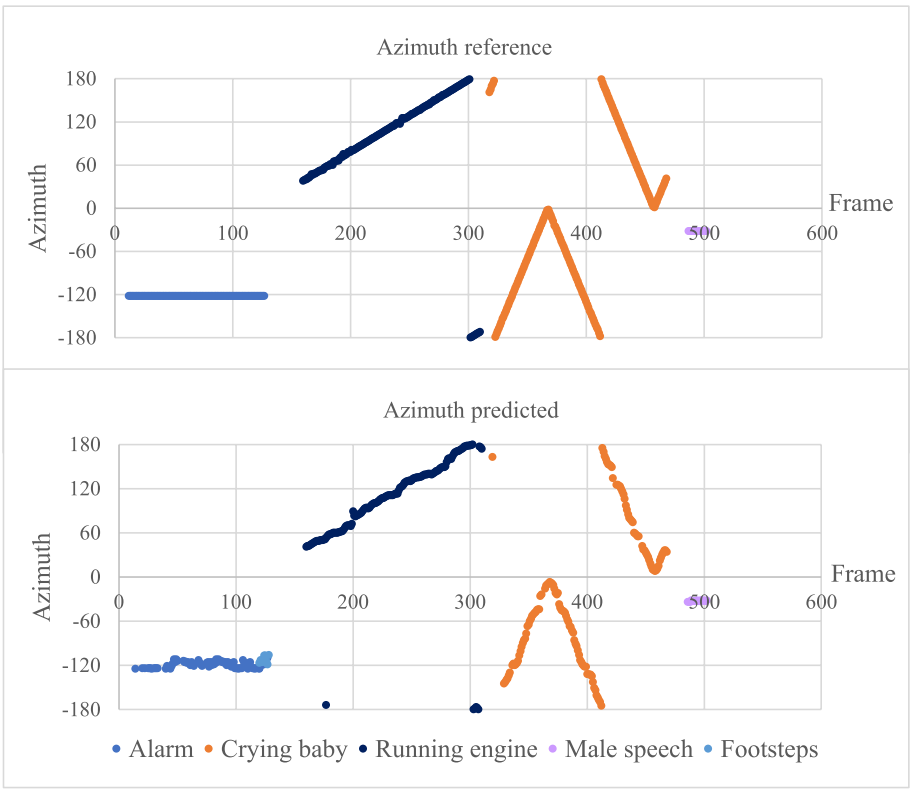

Fig. 8 The reference and prediction of azimuth in audio Mix80 for eval dataset

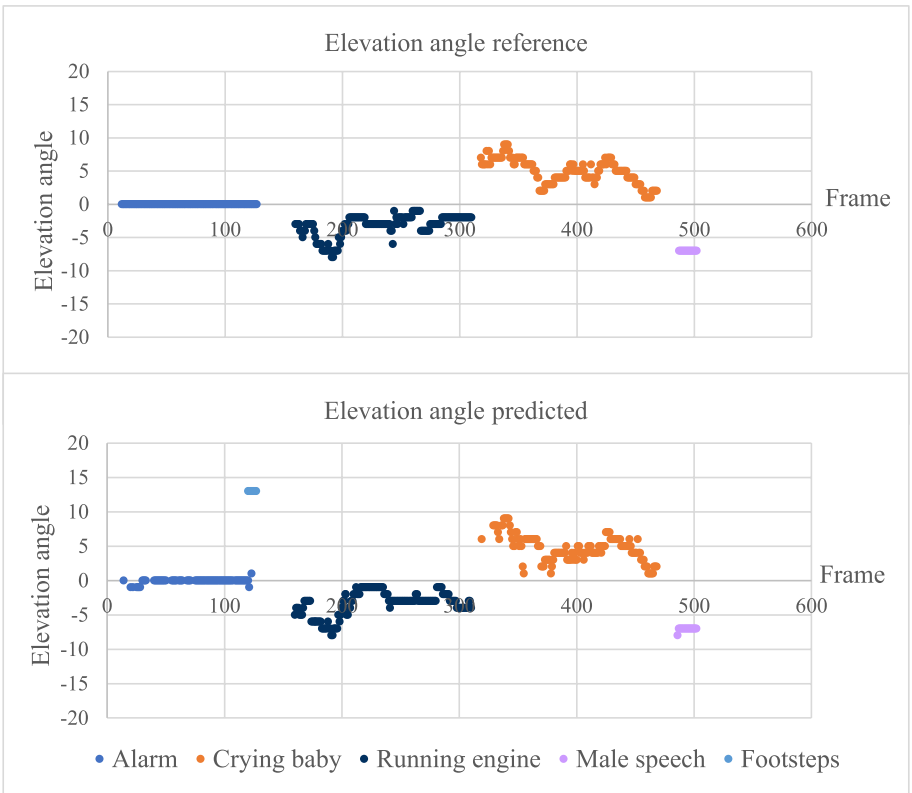

Fig. 9 The reference and prediction of elevation angle in audio Mix80 for eval dataset

accuracy rate and reducing false positives. Therefore, although our proposed method is slightly lower than Nguyen_NTU in terms of LR_{CD} metrics, the overall performance is the best.

Table 2 shows the results of the experiments performed on the MIC format dataset using the above method. As can be seen in Table 2, the model decreases in all metrics on the MIC dataset compared to the FOA format dataset, but the proposed method is still optimal. Similarly, the proposed method is worse on LR_{CD} metrics compared to Nguyen_NTU. In addition, the SELD score of our proposed method is 0.226, which is close to that of the Nguyen_NTU method. Furthermore, the F-score stays above 70%, the localization recall stays above 80%, and the localization error stays around 10 degrees on the FOA and MIC datasets, proving the effectiveness of our proposed method.

We respectively tested sound events with no overlap (i.e., the number of overlapping sounds is 1), sound events with two overlaps, and sound events with both the number of overlapping sounds 1 and 2 included. The data in Table 3 enable to prove that the proposed method is effective for the detection of two overlapping sound events.

We randomly select two audio segments, mix146 and mix80, from the FOA eval dataset as examples. The sound event categories and azimuth angles in the audio are drawn so that the SELD performance of the proposed method can be visually verified. Figures 6 and 7 show the reference and predicted of azimuth and elevation angles in audio mix 146, respectively. The horizontal coordinate in the figures is the frame, which corresponds to a temporal resolution of 100 ms. The vertical coordinate is the azimuth or elevation angle in degrees.

There are overlapping sound events in mix146. Except for the existence of a few points with large errors, the categories of sound events are accurately predicted. The azimuth and elevation angles are generally consistent with the reference. This example also demonstrates that the proposed method can distinguish between two sound events that overlap in time.

Mix80 contains multiple segments of moving audio. As can be seen in the Fig. 8 and Fig. 9, the proposed method predicts the moving spatial trajectories of the events in general agreement with the reference. The error is the prediction of one more sound type of footsteps, which is not present in the ground truth. This is also allowed within the error range.

4.2 Ablation experiments

The results in the previous subsection show that the proposed joint SELD method enhances the performance of sound event detection and localization compared to other methods. Two parts of ablation experiments were performed to further verify the effectiveness of the submodule of the proposed method. First, we validated the necessity of DBAM. Second, we verified the superiority of soft parameter sharing relative to not using parameter sharing networks. The experiments in Sect. 3.1 were evaluated on FOA eval dataset.

4.2.1 The necessity of DBAM

In this paper, we improved the conformer module to obtain the DBAM, which combines both local and global information. To verify the effectiveness of the DBAM, we designed experiments for comparison to determine whether the presence of the DBAM in the network is necessary. The model without DBAM, using the multi-headed attention module of the encoding part of the transformer, is shown in Fig. 10. The first part of the connection structure performs a normalized layer of residual connections after the multi-headed self-attention. The other part is a feed-forward FC layer followed by a normalized layer and a residual connection.

In addition, two layers of multi-headed attention modules are used to connect. In addition, experiments were conducted using the number of DBAM modules as 1, 2, and 4, respectively. The experimental results are shown in Table 4.

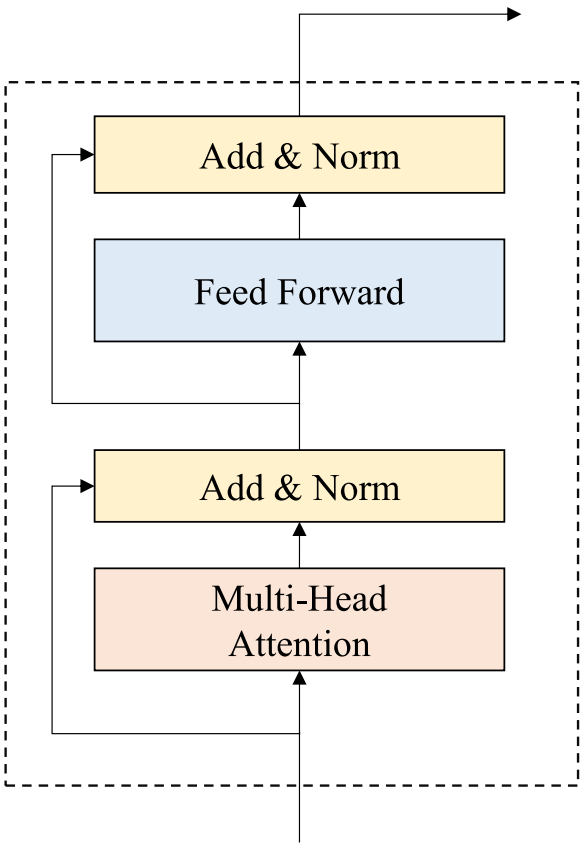

Fig. 10 Structure of the coding part without DBAM

Table 4 The SELD performance of the proposed method and without DBAM method

Models	$ER_{20°}\downarrow$	$F_{20°}\uparrow$	$LE_{CD}\downarrow$	$LR_{CD}\uparrow$	SELD↓
Without DBAM	0.416	68.7%	12.492°	74.7%	0.263
DBAM 1	0.367	71.0%	11.248°	79.6%	0.230
DBAM 2	0.347	74.0%	10.818°	80.2%	0.216
DBAM 4	0.376	71.5%	9.304°	80.6%	0.227

The experimental results are shown in Fig. 11, where the angular value of the positioning error LE is divided by 10 so that it is expressed using the same vertical axis as the other indicators.

It can be seen from Fig. 11 and Table 4 that the proposed DBAM is better than the method without DBAM in all metrics. This is a good indication of the effectiveness of the improved DBAM in terms of detection and localization. It can be seen from both LE_{CD} and LR_{CD} metrics that the DBAM makes the proposed model improve in terms of localization. However, the DBAM with more than 2 layers makes the SED metrics a little worse. Therefore, we finally use the DBAM with the best result of 2 layers.

In addition, we compared the number of parameters of the models in Table 5. To facilitate the comparison, all models are set to two layers.

As can be seen in Table 5, the network parameters of our proposed DBAM are much reduced than those of the conformer model. Therefore, it is possible to achieve a reduction in model size.

Table 5 The numbers of model parameters

Model	Conformer	DBAM	Without DBAM
Parameters	55,986,402	34,687,714	26,217,186

4.2.2 The effectiveness of soft parameter sharing

The method proposed in this paper uses soft parameter sharing to connect two subtask networks, SED and DOA, and optimizes parameter sharing through cross-stitch modules to achieve better results. To verify whether soft parameter sharing is advantageous, this paper uses the network without parameter sharing to learn SED and DOA respectively and compares the results of the two models. In Fig. 12, no PS indicates that no soft parameter sharing is performed in the two sub-networks.

5 Conclusion

In this paper, we propose a dual-branch attention module-based network for the SELD task. The module uses two parallel attention branches and convolution to fully integrate global and local information about the environment. In addition, we blend the low-level and high-level features of the localization sub-network to improve the localization performance. The DBAM is used as the attention mechanism for track separation in the joint SELD network. The parameters of SED and DOA are shared between the two sub-networks by the cross-stitch module. The effectiveness of the proposed method was proved by experiments on the DCASE 2020 task 3

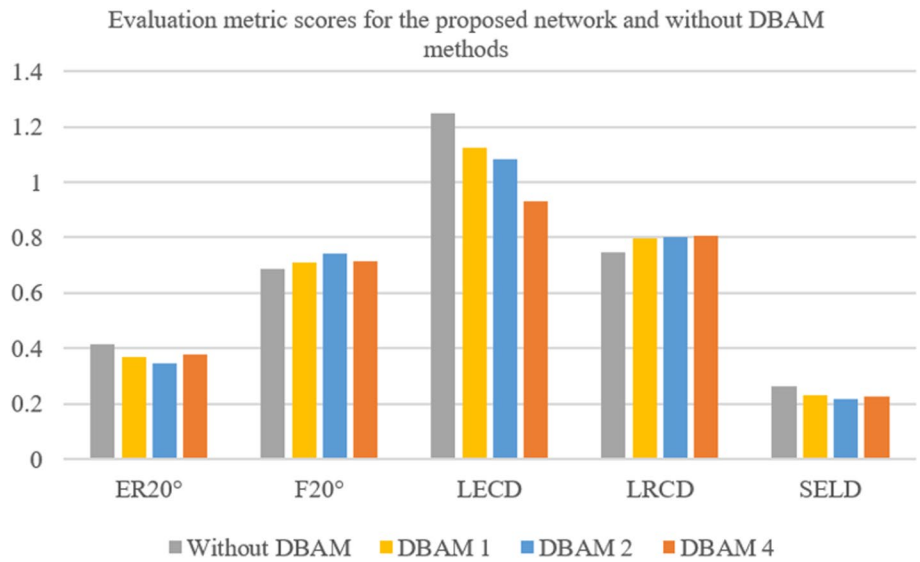

Fig. 11 The SELD performance of the proposed method and without DBAM method

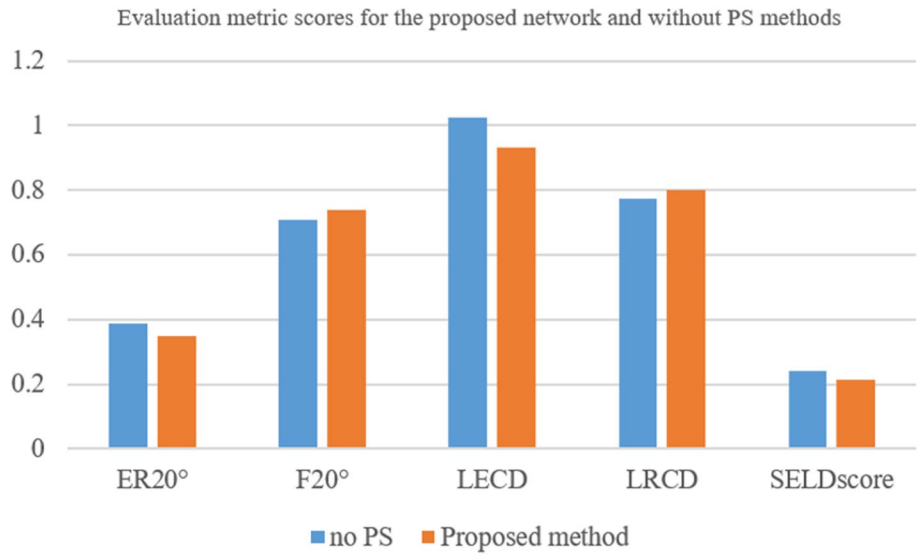

Fig. 12 The performance of the proposed method and no soft parameter sharing network

dataset. The results show that the proposed method has better detection and localization performance compared to the baseline model. Furthermore, the ablation experiments validate the need for DBAM and soft parameter sharing.

Compared to CNN or Transformer-based approaches, the proposed DBAM can efficiently model global and local contextual information, which is important for audio sequence processing tasks. The two-branch structure of DBAM makes the structure of the method simpler and also reduces the model parameters compared to the conformer. We hope that in future work we can try to replace the original attention module with the variant of self-attention. In addition, we will try other multi-task learning methods to obtain more accurate sound event detection and localization performance.

Abbreviations

SELD	Sound event detection and localization
SED	Sound event detection
SSL	Sound source localization
DOA	Direction of arrival
DBAM	Dual-branch attention module
ASR	Automatic speech recognition
DNN	Deep neural networks
CNN	Convolutional neural networks
RNN	Recurrent neural networks
CRNN	Convolutional recurrent neural networks
BiGRU	Bidirectional gated recurrent units
ReLU	Rectified linear unit
MHSA	Multi-headed self-attention
GeLU	Gaussian error linear unit
MTL	Multi-task learning
PIT	Permutation-invariant training
BCE	Binary cross-entropy
MSE	Mean square error
FOA	First-order Ambisonics
MIC	Microphone array
IV	Intensity vector
GCC-PATH	Generalized cross-correlation
SNR	Signal-to-noise ratio
FFT	Fast Fourier transform

Acknowledgements
Not applicable.

Authors' contributions
Y. Zhou conceived the research, conducted the experiments, and wrote the manuscript. H. Wan provided guidance on the structure and English modification of the paper. All authors reviewed and approved the final manuscript.

Funding
Not applicable.

Availability of data and materials
The DCASE 2020 dataset used in the experiments of this paper is available at https://dcase.community/challenge2020/task-sound-event-localization-and-detection#download.

Declarations

Ethics approval and consent to participate
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 4 February 2023 Accepted: 13 June 2023
Published online: 30 June 2023

References

1. P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, M. Vento, Audio surveillance of roads: a system for detecting anomalous sounds. J IEEE Transactions on Intelligent Transportation Systems. **17**(1), 279–288 (2016)

2. C. Grobler, C. Kruger, B. Silva, G. Hancke, *IECON 2017–43rd Annual Conference of the IEEE Industrial Electronics Society. Sound based localization and identification in industrial environments* (Beijing, IEEE, 2017), pp.6119–6124
3. G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci, A. Sarti, *2007 Conference on Advanced Video and Signal Based Surveillance. Scream and gunshot detection and localization for audio-surveillance systems* (IEEE, London, 2007), pp.21–26
4. C. Busso, S. Hernanz, C. W.Chu et al., IEEE International Conference on Acoustics, Speech, and Signal Processing. Smart room: participant and speaker localization and identification. (ICASSP, Philadelphia, PA, USA, 2005), pp. ii/1117-ii/1120, Vol. 2
5. D. Barchiesi, D. Giannoulis, D. Stowell, M.D. Plumbley, Acoustic scene classification: classifying environments from the sounds they produce. J IEEE Signal Processing Magazine. 32(3), 16–34 (2015)
6. H. Wang, P. Chu, *1997 IEEE InternationalConference on Acoustics, Speech, and Signal Processing. Voice sourcelocalization for automatic camera pointing system in videoconferencing* (ICASSP, Munich, 1997), pp.187–190
7. K.J. Piczak, *2015IEEE 25th International Workshop on Machine Learning for Signal Processing. Environmental sound classification with convolutional neural networks.* (MLSP, Boston, 2015), pp.1–6
8. G. Parascandolo, H. Huttunen, T. Virtanen, *2016 IEEE International Conference on Acoustics Speech and Signal Processing. Recurrent neural networks for polyphonic sound event detection in real life recordings* (ICASSP, Shanghai, 2016), pp.6440–6444
9. T. Hirvonen, *Audio Engineering Society 138th Convention, Classification of spatial audio location and content using convolutional neural networks* (AES, Warsaw, 2015)
10. S. Adavanne, P. Pertilä, T. Virtanen, *2017 IEEE International Conference on Acoustics Speech and Signal Processing. Sound event detection using spatial features and convolutional recurrent neural network* (ICASSP, New Orleans, 2017), pp.771–775
11. Y. Cao, Q. Kong, T. Iqbal, et al., polyphonic sound event detection and localization using a two-stage strategy. (2019). ArXiv Preprint arXiv:1905.00268
12. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., Attention is all you need. Advances in neural information processing systems, 2017, pp. 5998–6008. ArXiv Preprint arXiv: 1706.03762
13. Heittola, T., Mesaros, A., Eronen, A. et al. Context-dependent sound event detection. J EURASIP Journal on Audio, Speech, and Music Processing. (2013). https://doi.org/10.1186/1687-4722-2013-1
14. L. Dong, S. Xu, B. Xu, *2018 IEEE InternationalConference on Acoustics, Speech and Signal Processing. Speech-transformer: ano-recurrence sequence-to-sequence model for speech recognition* (ICASSP, Calgary, 2018), pp.5884–5888
15. Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., and Pang, R. Conformer: Convolution-augmented Transformer for Speech Recognition. In Proceedings of INTERSPEECH, 2020. ArXiv Preprint arXiv:2005.08100
16. Wu, Z., Liu, Z., Lin, J., Lin, Y., and Han, S. Lite Transformer with long-short range attention. In Proceedings of ICLR, 2020. ArXiv Preprint arXiv: 2004.11886
17. Yifan Peng, Siddharth Dalmia, Ian Lane, Shinji Watanabe, Branchformer: Parallel MLP-Attention architectures to capture local and global context for speech recognition and understanding. ICML 2022. ArXiv Preprint arXiv:2207.02971.
18. Hendrycks, D. and Gimpel, K. Gaussian error linear units (GELUs). (2016) ArXiv preprint arXiv:1606.08415
19. Yang Zhang, Zhiqiang Lv, Haibin Wu, Shanshan Zhang, Pengfei Hu, MFA-Conformer: Multi-scale feature aggregation conformer for automatic speaker verification. INTERSPEECH 2022. ArXiv preprint arXiv: 2203.15249
20. A. Mesaros, S. Adavanne, A. Politis, T. Heittola, T. Virtanen, *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. Joint measurement of localization and detection of sound events* (WASPAA, New Paltz, 2019), pp.333–337
21. Y. Lu, A. Kumar, S. Zhai, Y. Cheng, T. Javidi, R. Feris, *2017 IEEE Conference on Computer Vision and Pattern Recognition. Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification* (CVPR, Honolulu, 2017), pp.1131–1140
22. I. Mistra, A. Shrivastava, A. Gupta, M. Hebert, *2016 IEEE Conference on Computer Vision and Pattern Recognition. Cross-stitch networks for multi-task learning* (CVPR, Las Vegas, 2016), pp.3994–4003
23. K. Hashimoto, C. Xiong, Y. Tsuruoka, R. Socher, *2017 Conference on Empirical Methods in Natural Language Processing. a joint many-task model: growing a neural network for multiple NLP tasks* (EMNLP, Copenhagen, 2017), pp.1923–1933
24. R. Cipolla, Y. Gal, A. Kendall, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics* (CVPR, Salt Lake City, 2018), pp.7482–7491
25. Ruder, Sebastian, Joachim Bingel, Isabelle Augenstein and Anders Søgaard, Sluice networks: learning what to share between loosely related tasks. (2017). ArXiv preprint arXiv: 1705.08142
26. Y. Cao, T. Iqbal, Q. Kong, F. An, W. Wang, M. D. Plumbley, *2021 IEEE International Conference on Acoustics, Speech and Signal Processing. An improved event-independent network for polyphonic sound event localization and detection* (ICASSP, Toronto, 2021), pp.885–889
27. D. Yu, M. Kolbæk, Z.- H. Tan, J. Jensen, *2017 IEEE International Conference on Acoustics, Speech and Signal Processing. Permutation invariant training of deep models for speaker-independent multi-talker speech separation* (ICASSP, New Orleans, 2017), pp.241–245
28. A. Politis, S. Adavanne and T. Virtanen, A dataset of reverberant spatial sound scenes with moving sources for sound event localization and detection. Proc. DCASE 2020 Workshop, 2020, pp. 165–169. ArXiv preprint arXiv: 2006.01919
29. Y Cao T Iqbal Q Kong Y Zhong W Wang MD Plumbley Event-independent network for polyphonic sound event localization and detection. Proc. DCASE, 2020 Workshop, 2020 ArXiv preprint arXiv 2010 00140
30. T. N. T. Nguyen, Douglas L. Jones, W. Gan, DCASE 2020 TASK 3: Ensemble of sequence matching networks for dynamic sound event localization, detection and tracking. Proc. DCASE 2020 Workshop, 2020, pp. 120–124

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ► Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com