SoundSynp: Sound Source Detection from Raw Waveforms with Multi-Scale Synperiodic Filterbanks

Yuhang He
yuhang.he@cs.ox.ac.uk
Department of Computer Science
University of Oxford
United Kingdom

Andrew Markham
andrew.markham@cs.ox.ac.uk
Department of Computer Science
University of Oxford
United Kingdom

Abstract

We propose synperiodic filter banks, a novel multi-scale learnable filter bank construction strategy that all filters are **syn**chronized by their rotating **periodic**ity. By synchronizing in a certain periodicity, we naturally get filters whose temporal length are reduced if they carry higher frequency response, and vice versa. Such filters internally maintain a better time-frequency resolution trade-off. By further alternating the periodicity, we can easily obtain a group of synperiodic filter bank (we call synperiodic filter banks), where filters of same frequency response in different groups differ in temporal length. Convolving these filter banks with sound raw waveform achieves multi-scale perception in time domain. Moreover, applying the same filter banks to recursively process the 2x-downsampled waveform enables multi-scale perception in the frequency domain. Benefiting from the multi-scale perception in both time and frequency domains, our proposed synperiodic filter banks learn multi-scale time-frequency representation in a data-driven way. Experiments on both sound source direction of arrival (DoA) and physical location detection task show the superiority of synperiodic filter banks.

1 Introduction

The fundamental task for an agent to perceive and interact with the 3D environment is to know the location and semantic identity of its nearby objects. The location includes both spatial location and temporal location. Vision-based such environment perception has received large attention in the past decade and we have witnessed huge progress in tasks such as object detection (Liu et al., 2016; Lin et al., 2014), classification (He et al., 2016a) and tracking (Wang et al., 2019). Nevertheless, the research in sound-based counterpart has far lagged behind, despite all the fascinating properties sound signals exhibit. For example, sound is ubiquitous and insensitive to ambient illumination change, it has no field-of-view constraints and is capable of circumventing physical barriers to perceive scene beyond line-of-sight. As a sensing approach complementary to vision, sound-based perception essential for acoustic scene understanding. A fundamental task is the sound source detection from multi-channel sound waveforms.

To detect sound sources, we often deploy a spatially-configured microphone array to record an acoustic environment. The recorded sound is a highly compressed 1D time series. Since different sound sources have different frequency properties, it is essential to convert 1D waveform into 2D time-frequency representation. This is often achieved by projecting the raw waveform onto various frequency bases. A sound source's spatial location clue lies in inter-channel difference among waveforms (*i.e.* phase difference). It is essential to design a neural network that jointly encodes mono-channel time-frequency and inter-channel phase difference from the raw waveforms in a unified, parameter-frugal and computation-efficient manner. The learned representation should have expressive resolution in both time, frequency and space domains so that sound sources can be precisely detected.

However, learning such representation is a tough task. Challenges stem from both theoretical side and practical side. According to the Uncertainty Principle, we cannot achieve the optimal resolution in time and frequency domain at the same time, but instead keep a trade-off between them. Traditional hand-engineered sound feature (Davis and Mermelstein, 1980; Cao et al., 2021; Brandstein and Silverman, 1997) and some recently proposed learnable fil-

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

ter bank (Ravanelli and Bengio, 2018; Zeghidour et al., 2018) empirically set the same length for all filters, resulting in human-biased, unadjustable time-frequency resolution map. Some other works (Zeghidour et al., 2021; He et al., 2021; He and Markham, 2022) correlate filter frequency response and filter length by initializing in mel-scale, but it is neither scalable nor stable. Moreover, all of them process raw waveform with one-scale filter bank, we think such one-scale filter bank easily leads to non-optimal representation, especially when sound sources have spectrum overlap or undergo free spatial motion.

In this paper, we first give theoretical analysis on the filter bank impact on time-frequency representation. Based on the analysis, we propose a simple yet effective synperiodic filter banks construction strategy, in which synperiodic means each filter's temporal length and its carried frequency response are synchronized by rotating periodicity such that each filter's length is inversely proportional to its frequency resolution. The resulting synperiodic filter banks (we call it filter banks as it contains multiple filter bank groups) thus internally maintain a better time-frequency resolution trade-off than traditional fixed-length filter bank. Coupling the filter length with its frequency response helps us to reduce human intervention in filter bank design. By simply alternating the periodicity term, we further construct a group of synperiodic filter banks, with which we achieve multi-scale perception in time domain. At the same time, by applying a synperiodic filter banks to process one raw waveform as well as its consecutively-downsampled versions, we achieve multi-scale perception in frequency domain. The multi-scale perception in both time and frequency domain of synperiodic filter banks enables the neural network to dynamically learn better representation for sound source detection in a data-driven way. It is worth noting that synperiodic filter banks parameter number is just linear to filter number (adds up to less than 1% of the whole parameters) and it can be efficiently implemented as a 1D convolution operator.

Following the learnable synperiodic filter banks, we further design backbone network (a small lite one and a large one) with two paralleling branches with layerwise soft-parameter sharing to learn sound source's semantic and spatial location related representation jointly. Experiment on both direction-of-arrival (DoA) task and physical location estimation task shows that our proposed framework outperforms comparing methods significantly. Replacing existing method's head with our proposed synperiodic filter banks also improves the performance significantly. The source code is `https://github.com/yuhanghe01/SoundSynp`.

2 Related Work

Sound signal processing has been thoroughly studied in traditional digital signal processing area. The preliminary step of sound signal processing is usually to convert raw waveform into 2D time-frequency representation. There are two main realms: Fourier transform based and Wavelet based transform (Sturm, 2007). Traditional sound feature design are motivated influenced by human-auditory system. For example, they often convert frequency bins into mel-scale to imitate human hearing system, like MFCC (Davis and Mermelstein, 1980), LogMel (Cao et al., 2021; Grondin et al., 2019), the filter length is empirically chosen and often a windowing is added to avoid spectrum leakage. For inter-channel phase difference encoding, it is often recommended to encode in frequency domain due to the less-computation advantage. Typical phase difference features include GCC-Phat (Brandstein and Silverman, 1997) and intensity vector (Cao et al., 2021).

Sound source detection has been previously treated as sound structure (Thrun, 2006) estimation, sound object detection (He et al., 2021) and sound event detection and localization (SELD) problem (Adavanne et al., 2018; Thi Ngoc et al., 2021). It involves jointly identifying a sound source's semantic label and predicting its spatial location, the two sub-tasks have been thoroughly studied separately in acoustics (Nandwana and Hasan, 2016; Mohan et al., 2008; Sundar et al., 2020; Vera-Diaz et al., 2018) and computer vision community (He et al., 2016a). Kim *et al.* (Kim et al., 2019) provided a review and discussion for raw-audio based event classification. Benefiting from the success of the traditional hand-engineered sound feature and the large availability of mature image-based deep neural networks (He et al., 2016a), most work (Grondin et al., 2019; Cao et al., 2021; Grondin et al., 2019; Adavanne et al., 2018; Thi Ngoc et al., 2021) tackle the task by first extracting hand-engineered sound feature and then feeding them to mature image-based neural networks. This workflow is straightforward and often guarantees reasonably good results but it is not end-to-end trainable and heavily depend on image side which may not be optimal for sound processing. At the same time, some work (He et al., 2021; Adavanne et al., 2018; Tho Nguyen et al., 2020) have simplified the problem by assuming no two sound sources of the same semantic label but different spatial location happen at the same time. This assumption avoids semantic label and spatial location association issue but may not reflect real scenarios.

In recent years, a bunch of work tried to design neural work to directly learn from raw sound waveform, ranging from the earlier methods that directly apply stacked layers to process raw waveform (Schneider et al., 2019; Palaz et al., 2013; Jaitly and Hinton, 2011; Sainath et al., 2013) to the recent frequency-sensitive filter bank learning meth-

ods (Zeghidour et al., 2021; Ravanelli and Bengio, 2018; Zeghidour et al., 2018; He et al., 2021; Hoshen et al., 2015; Sainath et al., 2015; Luo and Mesgarani, 2019; He and Markham, 2022). The filter bank parameter is initialized in either mel-scale (Ravanelli and Bengio, 2018; Zeghidour et al., 2021; He et al., 2021; Zeghidour et al., 2018) or as Gammatone filter (Hoshen et al., 2015; Sainath et al., 2015). SoundDet (He et al., 2021) designs MaxCorr filter bank to directly convolve with multi-channel raw waveforms to learn phase difference aware features.

Multi-scale representation has a rich history in computer vision (Liu et al., 2016; Lin et al., 2016; He et al., 2016b), in which multi-scale representation strategy has been proposed to accommodate large object scale variation. The "multi-scale" discussed in this paper is slightly different from their definition, as we mainly indicate the interaction between a filter's temporal support region and frequency response. In sound signal processing, Mallat (Bruna and Mallat, 2013; Mallat, 2012) proposed wavelet scattering to obtain multi-scale sound representation by iteratively treating the proceeding processed sound waveform as new virtual waveform for further process. Won *et al.* (Won et al., 2020) proposed to learn multi-scale harmonic filters in a data-driven way, by exploiting audio signal inherent harmonic structures.

3 Multi-Scale Synperiodic Filter Banks

3.1 Background Knowledge Discussion

Sound source detection task aims at detecting each sound source's start/end time, the semantic identity and spatial location during its occurrence. Semantic identity cues mainly lie in mono-channel sound waveform time-frequency representation, and spatial location cues lie in inter-channel waveform difference (e.g. phase difference). To get the time-frequency representation for each mono-channel waveform, a frequency-selective filter bank $\mathcal{F}$ is often used to project the waveform onto different frequency bases. A general filter bank $\boldsymbol{\mathcal{F}}$ of M filters can be mathematically represented as,

$$\boldsymbol{\mathcal{F}} = \{\mathcal{F}^i\}_{i=1}^{M},\ \mathcal{F}^i_{f_i,\sigma_i}(t) = \phi_{f_i}(t) \cdot \omega_{\sigma_i}(t) \qquad (1)$$

Each filter $\mathcal{F}^i$ is a filter in time domain. It contains a frequency response f_i and filter length σ_i, that are independently controlled by frequency-selective filter $\phi_{f_i}(t)$ (e.g. (Ravanelli and Bengio, 2018)) and a window function $\omega_{\sigma_i}(t)$. An expressive filter bank should have good resolution capability in both time and frequency domains. Frequency resolution indicates the ability of discerning two adjacent frequency bins, time resolution corresponds to the capability of precisely localizing a sound source in time domain. According to the Uncertainty Principle, the frequency resolution Δ_f and time resolution Δ_t satisfies $\Delta_f \cdot \Delta_t \geq C$ (C is a constant, which means we cannot achieve the optimal resolution at time and frequency domains simultaneously, but instead maintain a trade-off between them. Therefore, it is essential to design a filter bank that maximally maintains a good time-frequency resolution.

Existing filter bank differs in their way of choosing $\phi_{f_i}(t)$ and $\omega_{\sigma_i}(t)$. Classic Fourier transform based filter banks, such as short-time Fourier transform (STFT), LogMel and MFCC (Davis and Mermelstein, 1980), decide $\phi_{f_i}(t)$ and $\omega_{\sigma_i}(t)$ independently in arbitrary manner. They usually assign a fixed window length to all filters (where $\omega_{\sigma_i}(t)$ is a constant), so their extracted time-frequency representation resolution is fixed and unadjustable across all frequency bins. We call such filter bank **syndistance** filter bank to emphasize their equal length property across all frequency responses. Wavelet transform (Sturm, 2007) inversely correlates window length with frequency response so that the filter with higher frequency response is naturally associated with shorter window. The resulting time-frequency map theoretically has better resolution than the one extracted by Fourier transform, but it is still fixed and heavily rely on empirical parameter tuning. Some recent work (He et al., 2021; Zeghidour et al., 2018, 2021) relax $\phi_{f_i}(t)$ to be trainable so that they can be further optimized in a data-driven way. They still involve much empirical parameter-tuning work (*i.e.* $\omega_{\sigma_i}(t)$ selection). Moreover, they all process the raw waveform in one-scale manner, which often leads to non-optimal time-frequency representation.

Synperiodic filter banks address these issues from three perspectives: **First**, we inversely correlate $\phi_{f_i}(t)$ and $\omega_{\sigma_i}(t)$ by a *periodicity* term. Therefore, we do not have to explicitly set the window length ($\omega_{\sigma_i}(t)$) for each filter because it is internally decided by the filter's frequency response. In addition, inversely correlating $\phi_{f_i}(t)$ and $\omega_{\sigma_i}(t)$ naturally generates filter bank in which filters with high frequency responses are associated with shorter window length. Such filter bank naturally maintains a better time-frequency resolution. **Second**, By simply alternating the *periodicity* term, we create a group of filter banks that differ in their window length. Applying these filter banks to process the raw waveform helps achieve multi-scale perception in time domain. **Third**, applying the same filter banks to process recursively 2x downsampled waveforms helps achieve multi-scale perception in frequency domain.

4 Synperiodic Filter Banks Construction

In synperiodic filter banks, we inversely correlate ϕ_{f_i} and ω_{σ_i} by setting σ_i to be proportional to the filter's periodic term: rotating ρ periodic circles. Specifically, synperiodic filter banks $\boldsymbol{\mathcal{F}}_{synp} = \{\mathcal{F}^{synp}_i\}_{i=1}^{M}$ can be represented as,

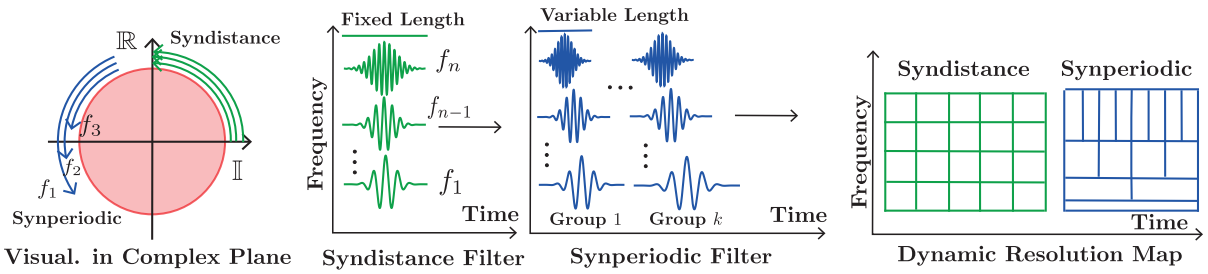

Figure 1: Synperiodic filter banks illustration: Syndistance filter bank (green color) rotates the same distance in complex plane and thus has the same kernel length, regardless of frequency it carries. Its time-frequency dynamic resolution map is thus rectangular. our proposed synperiodic filter banks (blue color) are generated by rotating the same periodicity number. So filter carrying lower frequency has larger kernel size than those with higher frequency response. As a result, synperiodic filter banks' time-frequency dynamic resolution map achieves better trade-off than traditional syndistance filter bank.

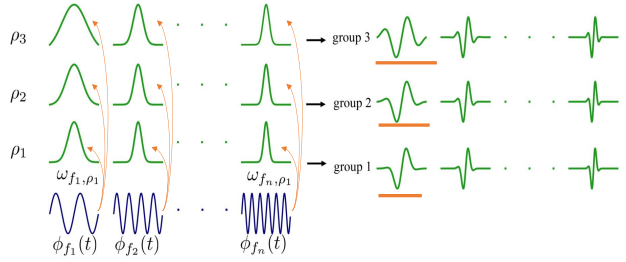

Figure 2: Synperiodic filter construction. Given a set of filters of infinite length $\phi_f(t)$ (dark blue) and predefined periodicity parameters $[\rho_1, \rho_2, \rho_3]$, windowing function $\omega_{f,\rho}$ takes the frequency f and periodicity ρ as input to create windows with locality property (green), and the window's active region length is inversely proportional to its corresponding filter's frequency response f. Multiplying the infinite filter and its associated window results in a group of Synperiodic filter bank. The filters of the same frequency response in different groups have different time length, helping to achieve multi-scale perception in time-scale.

$$\mathcal{F}^{synp}_{i,f_i}(t) = \phi_{f_i}(t) \cdot \omega_{f_i,\rho}(t) \tag{2}$$

where $\omega(f_i, \rho)$ indicates the periodic number the filter rotates. Since filters carrying higher-frequencies have shorter period length, requiring all filters to rotate the same period number naturally results in shorter filter length for high-frequency filters and wider filter length for low-frequency filters. Therefore, $\omega(f_i, \rho)$ defines the filter window length by constraining the period number it rotates. We call our filter banks synperioidc filter banks to emphasize that all filters' window lengths are automatically decided by the period number they rotate. To better understand the difference between syndistance and synperiodic filter banks, we visualize them in complex-valued plane (see Fig. 1, left-most), in which a filter is a complex exponential rotating in the complex plane counter-clockwisely, the rotating speed corresponds to the frequency it carries. In the complex-valued plane, all syndistance filters rotate to the same distance. Synperiodic filters, however, rotate to a predefined periodicity ρ, resulting in narrow window for high-frequency filters and wide window for low-frequency filters.

Synperiodic filter banks lend us three advantages: it **first** avoids us setting window length for each filter separately, which is quite empirical and random; **second**, the constructed filter banks internally maintain a good time-frequency resolution trade-off; **third**, by simply varying the periodicity term ρ, we can easily obtain a group of synperiodic filter banks to process the raw waveforms in multi-scale manner. Our synperiodic construction strategy shares similar idea with Wavelet transform (Sturm, 2007) where it adopts a time shift and "squeezing ratio" to achieve multi-scale perception. The difference is that we omit the time shift but instantiate the squeezing ratio with our proposed synperiodicity strategy. Moreover, synperiodic filter banks are multi-scale both time and frequency domain, and self-adjustable in a data-driven way.

There are many ways to instantiate $\omega(w_i, \rho)$, as long as we guarantee the window length gradually reduces as the frequency response increases. The simplest choice is to treat $\omega(w_i, \rho)$ as a constant, but we find it either results in too wide window for low-frequency filters or too narrow window for high-frequency filters. To mitigate this dilemma, we use logarithmic window function,

$$\omega(f_i, \rho) = 27 \cdot \log_{10}(f_i) - \rho, \ \rho = \{-6, -11, -16\} \tag{3}$$

We set ρ as $[-6, -11, -16]$ respectively to construct three synperiodic filter banks. The design of this window function is motivated by mel-scale frequency initialization strategy. By roughly setting a filter's bank width to be equal to the distance between its preceding and next frequency location in frequency domain, converting to time domain we can roughly get a logarithmic scale frequency-periodicity relationship (see Fig. 1 in Appendix).

Figure 2 visualizes synperiodic filter banks construction. In

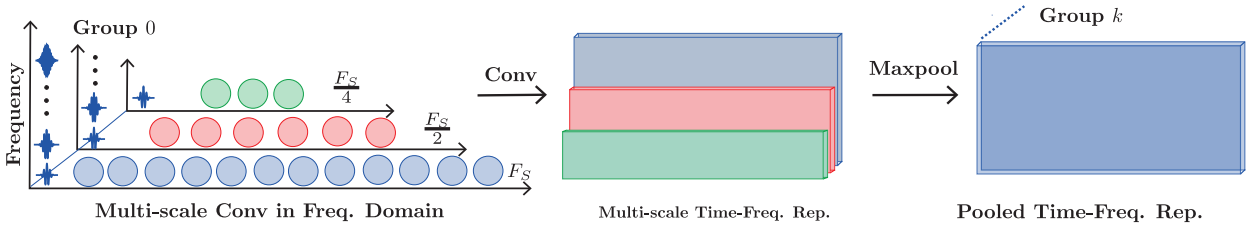

Figure 3: Multi-scale learning in frequency domain. Given the raw one channel sound waveform and pre-constructed synperiodic filter banks, we consecutively downsample the waveform by factor 2x, the newly downsampled waveform is processed by low-half filter bank from the proceeding filter bank. We can obtain time-frequency representation on each frequency scale. These time-frequency representations share the same time length by adjusting step size. The final time-frequency representation is obtained by max-pooling them together.

our implementation, synperiodic filter is created by multiplying a sinusoidal basis ($\phi(f_i)$ instantiation) with learnable frequency response initialized in mel-scale by a Gaussian window ($\omega(f_i, \rho)$ instantiation) with learnable width initialized through the windowing function by Eqn. (3). The initial synperiodic filter extracted features in different channels are in a complex format, we further encode cross-spectrum feature as spatial location relevant feature (see Sec. 1 in supp. material) and concatenate them with synperiodic filter extracted together as the overall sound source feature. Synperiodic filter banks are initialized with independent learnable frequencies and window length, they are independently updated during training stage.

4.1 Multi-Scale Perception in Time Domain

We use the previously constructed synperiodic filter banks to convolve with mono-channel sound waveform with the same step size and padding strategy, resulting in the same size output for each single synperiodic filter banks. Since different filter group has different window length, we achieve multi-scale learning in time domain (see the third figure in Fig. 1). It maximally avoids us empirically selecting one window scale ρ, but instead uses a group of filter banks to enforce the neural network to strike a better time-frequency resolution trade-off in a data-driven way.

4.2 Multi-Scale Perception in Frequency Domain

Multi-scale perception in frequency domain is hierarchical: given a raw sound waveform with sampling frequency F_S, synperiodic filter banks' frequency is initialized within the range $[0, \frac{F_S}{2}]$ under Nyquist sampling theorem. If we downsample the sound waveform by a factor of 2, the resulting waveform can be processed by the lower-half filters in each group whose frequency response lies in $[0, \frac{F_S}{4}]$. Please note that merely using the lower-half filters to process the 2x-downsampled waveform helps us to avoid aliasing issue. This process that 2x-downsampling the waveform further process the downsampled waveform with filters with lower-half frequency response can potentially iterate a couple of times (in our case three times), resulting in multi-scale perception in frequency domain. Figure 3 illustrates how it works. Multi-scale learning in frequency domain brings us two extra benefits: 1) from data augmentation perspective, 2x downsampling a waveform creates new low-quality waveform, equivalently we have extra dataset. 2) from the perception field perspective, the adjacent 2x-downsampling strategy leads to dilated convolution for lower-frequency filters, because applying a filter to convolve with a downsampled waveform equals to convolve on the original waveform with dilated convolution (skip-2 connection). The resulting wider or dilated perception field for lower frequency filters enables to learn better sound representation along the time axis (see Fig. 2 in supp. material). In sum, by using learnable synperiodic filter bank group to process the raw waveforms in multi-scale manner, we achieve a dynamic time-frequency resolution that naturally maintains a better time-frequency resolution fitting for sound source detection in a data-driven way.

Computational Analysis Synperiodic filter bank group introduces very few parameters (less than 1%) because they are parameterized filters. The trainable parameter number increases linearly w.r.t. synperiodic filter bank number. Their convolution with raw waveforms can also be efficiently implemented with 1D convolution.

4.3 Backbone Neural Network

Following the Synperiodic filter banks, we further design a backbone network to further learn sound source representation. Jointly learning sound source semantic label and spatial location representation is a multi-task problem (Kendall et al., 2018; Misra et al., 2016), we follow (Cao et al., 2021) to propose a backbone with two paralleling and identical branches to learn each sub-task separately. To enforce information communication, we add a layerwise information exchange module: for the intermediate semantic label feature f_s^i and spatial location feature f_g^i learned by the i-th block, a learnable weight W_i is introduced to linearly com-

bine them together to get an updated f_s^i and f_g^i before feeding them to the next layer, $[f_s^i, f_g^i] = W_i \cdot [f_s^i, f_g^i]$. On top of the representation, we add trackwise permutation-invariant training (PIT) strategy to train the whole neural network in an end-to-end manner. We have designed two backbone versions: a lite version of $24M$ parameters and a large version of $60M$ parameters (see Table 8, 9 in Appendix for network architecture).

5 Experiments

We focus on two tasks: direction of arrival (DoA) and physical location estimation. For DoA estimation task, we use DCASE2020 sound event detection and localization dataset (Politis et al., 2020), in which the sampling rate is 24 k. It contains 14 sound sources with azimuth range $[-180°, 180°]$ and elevation range $[-45°, 45°]$. Two recording formats: FOA and MIC-array (refer Sec. 1 in Appendix). More details are in (Politis et al., 2020). For physical location, we use L3DAS22-SELD dataset (Guizzo et al., 2022).

5.1 Comparing Methods

For DoA, we compare with five latest methods:

1. **SELDNet** (Adavanne et al., 2018), SELDNet is the baseline model and it jointly trains sound source's semantic label and spatial location with a convolutional recurrent neural network (CRNN) (Chung et al., 2014).

2. **EIN-v2** (Cao et al., 2021), EIN-v2 (Cao et al., 2021) is a very recent work. It adopts multi-heads self-attention (Vaswani et al., 2017) (MHSA) to model temporal dependency and trackwise permutation-invariant training to train the model.

3. **SoundDet** (He et al., 2021), SoundDet directly learns from raw waveforms with MaxCorr kernels, followed by an encoder-decoder neural network to learn frame-wise representation.

4. **SoundDoA** (He and Markham, 2022) SoundDoA also learns from raw waveform by a Gabor-like filter bank with an *Enhance* module. A backbone neural network is associated with the Gabor-like filter bank to learn time-frequency representation.

5. **Utsc-Iflytek** (Wang et al., 2020). Utsc-Iflytek is ranked first in DECASE2020 challenge leaderboard [1], it combines MIC and FOA features and ensembles different models like ResNet (He et al., 2016a) and Xception (Chollet, 2017) to detect sound sources.

For sound source physical location estimation task, we additionally compare with Conf-EIN (Hu et al., 2022), which is based on EIN-v2 (Cao et al., 2021) and additionally contains conformer and dense blocks. We call our framework SoundSynp (the one with small backbone SoundSynp_lite, large backbone SoundSynp_large). All methods' comparison is in Table 4.

[1] see link for leaderboard report.

5.2 Implementation Detail

To train the neural network, we clip all 4s short snippets from the original one minute long four-channel raw waveforms so that we have the largest train dataset. The raw waveforms are normalized to $[-1, 1]$. We adopt Adam optimizer (Kingma and Ba, 2015) with an initial learning rate 0.0002 in the first 100 epochs and 0.0001 in the following 50 epochs. The loss combination weight between classification head and regression head is $1 : 2$. During training, data augmentation method SpecAugment (Park et al., 2019) is applied. For DoA task, we regress direction of arrival angle in Cartesian coordinates $[x, y, z]$. In synperiodic filter banks, the filter length is 1025, each group's filter number is 256 and the step size is 600. Particularly, we have observed the initialized learnable synperiodic filter banks update its parameters intensively during the early several epochs, and then gradually becomes stable. We train each model with Pytorch (Paszke et al., 2019) **five times independently and report the average score**. The standard deviation is within 0.04 (for recall) and $0.17°$ for angle, 0.002 for mAP and mAR, we do not report the standard deviation in tables for succinct report.

5.3 Direction of Arrival (DoA) Estimation Result

Evaluation Metrics: We use two metrics. **Segment-based** metric is a widely adopted evaluation metric (Adavanne et al., 2018; Cao et al., 2021; He and Markham, 2022; He et al., 2021), it couples semantic label and spatial location together: a semantic-correctly detected sound source needs to be spatially close enough to its ground truth location in order to be regarded as a true positive detection. **Event-based** based metric is newly proposed by (He et al., 2021) to comprehensively evaluate under different confidence scores. Like object detection from images (Lin et al., 2014), it computes mean average precision (mAP) and mean average recall (mAR).

The result is given in Table 1, from which we see that SoundSynp (both the lite and large versions) achieves the best performance over all comparing methods significantly. Both EIN-v2 and UTSC-Iflytek use pre-extracted hand-engineered sound features, such as Logmel, GCC-Phat and Intensity Vector. SELDNet (Adavanne et al., 2018) uses phase and spectrum. SoundDet (He et al., 2021), SoundDoA (He and Markham, 2022) and SoundSynp are the only three methods that directly learn from raw waveforms. At the same time, SoundSynp obtains better performance on FOA than MIC format, the same phenomena has been ob-

Table 1: Result on DoA task. For Segment-based eval., we report detection error ER_{20°, F-measure F_{20° under DoA threshold 20°, and classification dependent localization error LE_{CD} and localization recall LR_{CD}. For event-based eval., we report mAP/mAR. The "Input" column labels are: 0. Raw waveforms, 1. Log-Mel, 2. GCC-Phat, 3. Intensity Vector. Top three performances are respectively highlighted by **red**, **green**, and **blue** color.

Methods	Input	Segment-Based Eval.				Event-based Eval.	
		$ER_{20^\circ}(\downarrow)$	$F_{20^\circ}(\uparrow)$	$LE(\downarrow)$	$LR(\uparrow)$	$mAP(\uparrow)$	$mAR(\uparrow)$
SELDNet(foa) (Adavanne et al., 2018)	1,3	0.63	0.46	23.1	0.69	0.087	0.152
SELDNet(mic) (Adavanne et al., 2018)	1,2	0.66	0.43	24.2	0.66	0.079	0.140
EIN-v2(foa) (Cao et al., 2021)	1,2	0.30	0.77	8.9	0.84	0.134	0.256
SoundDet(foa) (He et al., 2021)	0	0.25	0.81	8.3	0.82	0.197	0.294
SoundDoA(foa) (He and Markham, 2022)	0	0.23	0.85	7.9	0.87	0.220	0.301
UTSC-Iflytek(foa+mic) (Wang et al., 2020)	1,2,3	0.20	0.85	6.0	0.89	-	-
SoundSynp_lite(mic)	0	0.21	0.83	6.2	0.87	0.199	0.303
SoundSynp_large(mic)	0	0.19	0.86	5.5	0.91	0.210	0.313
SoundSynp_lite(foa)	0	0.20	0.85	5.6	0.89	0.205	0.309
SoundSynp_large(foa)	0	**0.15**	**0.89**	**4.3**	**0.94**	**0.232**	**0.327**

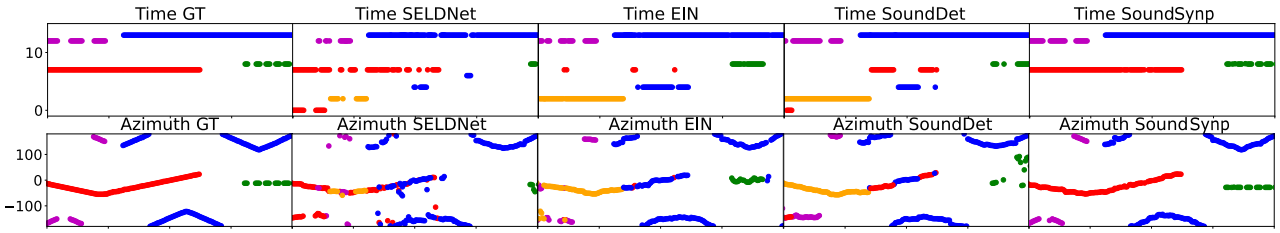

Figure 4: **DoA result visualization**. We show detected sound source temporal location (top row) and azimuth (bottom row). The horizontal axis is time, the vertical axis is semantic label (top) and azimuth in degree(bottom). Different color indicates different sound source class.

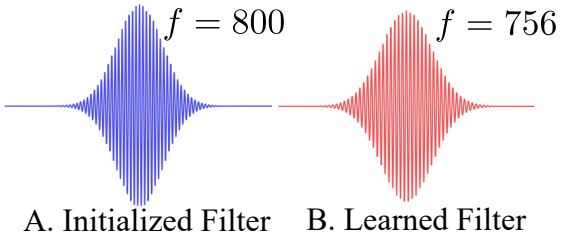

Figure 5: Learned Synperiodic Filterbank Visualization.

served by all other methods. It thus shows FOA better fits for sound source detection than MIC format. It is worth noting that Utsc-Iflytek (Wang et al., 2020) ensembles different powerful image-based 2D models to detect sound sources. However, our proposed SoundSynp still outperforms Utsc-Iflytek by a large margin. SoundSynp_lite achieves comparable performance with Utsc-Iflytek with much smaller parameter size (24M). It thus shows our proposed synperiodic filter banks are capable of learning expressive representation for sound source detection. We do not report mAP/mAR value for Utsc-Iflytek because it is a complex system and no detail about their system is available.

We show one learned synperiodic filter in Fig. 5, in which the filter's temporal support region and frequency response is updated to achieve better time-frequency resolution.

5.4 Ablation Study

1. **Existing Methods with Synperiodic Frontend**. We replace either fixed TF feature extractor front-end of SELDNet, EIN-v2, or learnable TF extractor front-end of SoundDet (He et al., 2021) and SoundDoA (He and Markham, 2022) with our proposed Synperiodic filter banks front-end to test their performance. It removes the influence of the backbone neural network of different models and thus helps to get direct comparison of synperiodic filter banks with other fixed time-frequency features. The result is in Table 2, from which we can see using synperiodic filter banks as a replacement of existing filter bank dramatically improves the performance. Synperiodic filter banks can be used as a general plug-and-play front-end by existing methods.

2. **Replace Synperiodic with Classic TF Feature**. In SoundSynp, we replace the synperiodic filter bank group with MFCC (Davis and Mermelstein, 1980) and Log-Mel (used by SELDNet (Adavanne et al., 2018) and EIN-v2 (Cao et al., 2021)), STFT and Wavelet (Sturm, 2007) like filters (we use the typical Gabor filter bank, we call SSynp_Gabor). It helps us to understand the performance with/without synperiodic filter banks. The result is given in Table 5, from which we can see replacing SoundSynp's

Table 2: Existing Methods with Synperiodic Frontend

Method	ER ↓	F ↑	LE↓	LR↑
SELDNet (Grondin et al., 2019)	0.63	0.46	23.1	0.69
SELDNet_Synp	**0.50**	**0.53**	**21.0**	**0.78**
EIN-v2 (Cao et al., 2021)	0.30	0.77	8.9	0.84
EIN-v2_Synp	**0.22**	**0.84**	**6.7**	**0.89**
SoundDet (He et al., 2021)	0.25	0.81	8.3	0.82
SoundDet_Synp	**0.21**	**0.85**	**7.5**	**0.87**
SoundDoA (He and Markham, 2022)	0.23	0.85	7.9	0.87
SoundDoA_Synp	**0.21**	**0.87**	**7.2**	**0.90**

Table 3: Various SoundSynp Variants Results

Variants	ER ↓	F ↑	LE↓	LR↑
SSynp_MSFreq	0.20	0.84	7.3	0.86
SSynp_MSTime	0.23	0.83	7.0	0.88
SSynp_Linear	0.22	0.83	8.4	0.84
SSynp_SScale	0.25	0.81	7.8	0.84
SSynp_Sinc	0.21	0.83	6.3	0.87
SSynp_LEAF	0.22	0.84	5.7	0.86
SoundSynp	**0.15**	**0.89**	**4.3**	**0.94**

Table 4: Network Architecture Highlight. MHSA: multi-head self-attention.

Variants	Network Blocks
SELDNet (Grondin et al.)	Conv2D, biGRU
EIN-v2 (Cao et al.)	Conv2D, MHSA
SoundDet (He et al.)	Conv1D LSTM
SoundDoA (He and Markham)	Conv1D/2D MHSA
SoundSynp	Conv1/2D MHSA

Table 5: Replace Synperiodic Learnable Frontend with traditional TF feature.

Method	ER ↓	F ↑	LE↓	LR↑
SSynp_MFCC	0.22	0.81	6.7	0.86
SSynp_LogMel	0.23	0.84	6.6	0.86
SSynp_STFT	0.23	0.82	7.0	0.84
SSynp_Gabor	0.22	0.82	7.3	0.85
SoundSynp	**0.15**	**0.89**	**4.3**	**0.94**

synperiodic filter banks with classic hand-engineered TF features inevitably reduces the performance under all evaluation metrics. It thus shows learning from either fixed or single-scale TF feature leads to worse performance than our proposed multi-scale synperiodic filter banks on sound source detection task.

3. **Necessity of Each SoundSynp Component**. We internally test six synperiodic filter banks variants: (1) synperiodic filter banks with just multi-scale in frequency domain (SSynp_MSFreq); (2) just multi-scale in time domain (SSynp_MSTime); (3) Synperiodic filter banks with frequency responses linearly initialized in Nyquist frequency range (SSynp_Linear, compare with our mel-scale initialization); (4) just one synperiodic filter bank without multi-scale perception neither in time nor frequency domain (SSynp_SScale); (5) with rectangular band-pass frequency response initialization (SSynp_Sinc), like SincNet (Ravanelli and Bengio, 2018) does; (6) with LEAF (Zeghidour et al., 2021) learnable filter bank (SSynp_LEAF). The result is given in Table 3. We can observe that he absence of multi-scale perception in either frequency domain or time domain inevitably reduces the performance. We find semantic label detection suffers more in single-scale perception in time domain than in frequency domain (see better performance on ER_{20°, and F_{20° score), which shows frequency domain multi-scale perception is vital for semantic estimation. Similarly, we can observe that multi-scale perception in time domain is vital for sound source spatial location estimation (see better performance on LE and LR score). Linearly initialized filter bank frequency reduces the performance as well, which shows assigning more filters to the lower frequency range is important. But this conclusion might be data-biased because we find DCASE dataset contains many low-frequency sound events like burning fire. Moreover, reducing the synperiodic filter bank groups to one group with just single-scale perception leads to the worst performance, it thus shows multi-scale perception in both time and frequency domain is essential for DoA-based sound source detection. Moreover, SoundSynp_Sinc leads to slightly inferior performance than our used mel-scale initialization strategy, it attests synperiodic filter framework is general enough to be adopted by various initialization strategy.

One qualitative comparison is shown in Fig. 4. We can clearly see that SELDNet (Grondin et al., 2019) generates mixed prediction at different time steps and DoA locations. SoundDet (He et al., 2021) and EIN-v2 (Cao et al., 2021) give non-existing sound sources (orange color). When multiple sound sources happen at the same time (polyphonicity), SoundDet and EIN are easily failed to predict the right spatial location (discretized blue and red color). Our method (SoundSynp_large) predicts more spatially and temporally consistent sound sources by maximally keeping sound source's continuity and completeness.

5.5 Physical Location Estimation Result

We run experiment on L3DAS22-SELD dataset (Guizzo et al., 2022), whose target is to predict sound source's 3D physical location $[x, y, z]$ in indoor room environment. The room is of size $6m \times 5m \times 3m$. It involves 14 seed sounds from FSD50K (Fonseca et al., 2022). Up to 3 sources are co-emitting sound. The dataset contains 600/150/150 30s-clips for train/val/test. We report the result on val-

Table 6: Physical Location Detection Result. The top three performances are labelled by red, green and blue color.

Method	$F_{\leq 1m} \uparrow$	$F_{\leq 2m} \uparrow$
EIN-v2 (Cao et al., 2021)	0.621	0.636
SoundDet (He et al., 2021)	0.640	0.672
SoundDoA (He and Markham, 2022)	0.652	0.688
Conf-EIN (Hu et al., 2022)	0.685	0.715
SoundSynp_lite	0.644	0.681
SoundSynp_large	**0.722**	**0.733**

Table 7: Inference Time and Param. Size (M: million).

Method	Infer. Time	Param. Size
SELDNet (Adavanne et al.)	1.20 s	0.5 M
EIN-v2 (Cao et al.)	2.20 s	26 M
SoundDet (He et al.)	1.25 s	13 M
SoundDoA (He and Markham)	2.10 s	27 M
Conf-EIN (Hu et al.)	4.0 s	83 M
SoundSynp_lite	1.80 s	24 M
SoundSynp_large	3.10 s	60 M

idation set because the test set is held-out for the challenge and not publicly available. The evaluation metric used here is F-score (Mesaros et al., 2019; Guizzo et al., 2022). In addition to EIN-v2 (Cao et al., 2021), SoundDet (He et al., 2021) and SoundDoA (He and Markham, 2022) (SELDNet model does not converge in training), we also compare with the champion method Conf-EIN (Hu et al., 2022), it ranks the first in L3DAS22-SELD challenge. The result is in Table 6, we can see SoundSynp_large outperforms Conf-EIN (Hu et al., 2022) by a large margin with smaller parameter size and inference time (see Table 7). SoundSynp_lite achieves comparable performance with Conf-EIN (Hu et al., 2022). It thus shows SoundSynp framework is capable of accurately detecting sound sources 3D physical location.

The inference time (Intel(R) Core(TM) i9-7920X CPU, 100 independent tests, report the average time) and model parameters of all methods are given in Table 7, from which we can see that SoundSynp_lite has comparable parameters and smaller inference time than EIN-v2 (Cao et al., 2021). SoundSynp_large has fewer parameters and less inference time than Conf-EIN (Hu et al., 2022), and it outperforms Conf-EIN (Hu et al., 2022) in physical location based sound source detection.

5.6 Limitation Discussion

One limitation is the lack of experiment of testing synperiodic filter banks on highly-polyphonic sound scenes, where multiple sound sources (more than 3) are co-emitting sound. Another limitation is the lack of testing synperiodic filter banks in tasks outside sound source detection (e.g. speech and other vibro-acoustics area). Moreover, the instantiation of synperiodic filter banks in Eqn. 3 is just one appropriate instantiation (no proof to show it is optimal, we show it outperforms SSynp_Sinc, see Table 3). There are other potential instantiations that await to be discussed.

Acknowledgements

We thank Prof. Niki Trigoni from department of computer science, Prof. Xiaowen Dong from department of engineering science, University of Oxford for constructive suggestions. We also appreciate reviewers for useful feedback and comments. Moreover, Yuhang He expresses his gratitude to all pioneers for their great work on signal processing, either on Fourier transform kingdom or on Wavelet transform kingdom, which inspired him on developing this work.

References

Sharath Adavanne, Archontis Politis, Joonas Nikunen, and Tuomas Virtanen. Sound Event Localization and Detection of Overlapping Sources Using Convolutional Recurrent Neural Networks. In *IEEE Journal of Selected Topics in Signal Processing*, 2018.

M. S. Brandstein and H. F. Silverman. A Robust Method for Speech Signal Time-delay Estimation in Reverberant Rooms. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1997.

J. Bruna and S. Mallat. Invariant Scattering Convolution Networks. *IEEE Transactions on Pattern Analysis Machine Intelligence (TPAMI)*, 2013.

Yin Cao, Turab Iqbal, Qiuqiang Kong, Fengyan An, Wenwu Wang, and Mark D Plumbley. An Improved Event-Independent Network for Polyphonic Sound Event Localization and Detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

François Chollet. Xception: Deep Learning with Depthwise Separable Convolutions. In *IEEE/CVF Computer Vision and Pattern Recognition (CVPR)*, 2017.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical Evaluation of Gated Recurrent Neural Ntworks on Sequence Modelling. In *Advances Neural Information Processing System (NeurIPS)*, 2014.

Steven Davis and Paul Mermelstein. Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. In *IEEE Transactions on Acoustics, Speech, and Signal Processing (TASSP)*, 1980.

Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra. FSD50K: An Open Dataset of Human-Labeled Sound Events. *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, 2022.

Francois Grondin, James Glass, Iwona Sobieraj, and Plumbley Mark D. A Study of the Complexity and Accuracy of Direction of Arrival Estimation Methods Based on GCC-PHAT for a Pair of Close Microphones. In *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, 2019.

Eric Guizzo, Christian Marinoni, Marco Pennese Pennese, Xinlei Ren, Xiguang Zheng, Chen Zheng, Bruno Masiero, Aurelio Uncini, and Danilo Comminiello. L3DAS22 Challenge: Learning 3D Audio Sources in a Real Office Environment. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Sun Jian. Identity Mappings in Deep Residual Networks. In *European Conference on Computer Vision (ECCV)*, 2016a.

Yuhang He and Andrew Markham. SoundDoA: Learn Sound Source Direction of Arrival and Semantics from Sound Raw Waveforms. In *Interspeech*, 2022.

Yuhang He, Long Chen, and Jianda Chen. Multi-task Relative Attribute Prediction by Incorporating Local Context and Global Style Information. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2016b.

Yuhang He, Andrew Markham, and Niki Trigoni. SoundDet: Polyphonic Moving Sound Event Detection and Localization from Raw Waveform. In *International Conference on Machine Learning (ICML)*, 2021.

Yedid Hoshen, Ron J. Weiss, and Kevin W. Wilson. Speech Acoustic Modeling from Raw Multichannel Waveforms. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

Jinbo Hu, Yin Cao, Ming Wu, Qiuqiang Kong, Feiran Yang, Mark D. Plumbley, and Jun Yang. A Track-wise Ensemble Event Independent Network for Polyphonic Sound Event Localization and Detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

Navdeep Jaitly and Geoffrey Hinton. Learning a Better Representation of Speech Soundwaves Using Restricted Boltzmann Machines. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In *IEEE/CVF Computer Vision and Pattern Recognition (CVPR)*, 2018.

Taejun Kim, Jongpil Lee, and Juhan Nam. Comparison and Analysis of SampleCNN Architectures for Audio Classification. *IEEE Journal of Selected Topics in Signal Processing*, 2019.

Diederik Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representation (ICLR)*, 2015.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *European Conference on Computer Vision (ECCV)*, 2014.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature Pyramid Networks for Object Detection. In *IEEE/CVF Computer Vision and Pattern Recognition (CVPR)*, 2016.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander Berg. SSD: Single Shot MultiBox Detector. In *European Conference on Computer Vision (ECCV)*, 2016.

Yi Luo and Nima Mesgarani. A Sequence Matching Network for Polyphonic Sound Event Localization and Detection. In *IEEE Transactions on Audio, Speech and Language Processing (TASLP)*, 2019.

Stéphane Mallat. Group Invariant Scattering. In *Communications in Pure and Applied Mathematics*, 2012.

Annamaria Mesaros, Sharath Adavanne, Archontis Politis, Toni Heittola, and Tuomas Virtanen. Joint Measurement of Localization and of Detection of Sound Event. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2019.

Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-Stitch Networks for Multi-task Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Satish Mohan, Michael Lockwood, Michael L. Kramer, and Douglas L. Jones. Localization of multiple acoustic sources with small arrays using a coherence test. In *The Journal of Acoustical Society of America*, 2008.

Mahesh Kumar Nandwana and T. Hasan. Towards SmartCars That Can Listen: Abnormal Acoustic Event Detection on the Road. In *Interspeech*, 2016.

Dimitri Palaz, Ronan Collobert, and Mathew Magimai-Doss. Estimating Phoneme Class Conditional Probabilities from Raw Speech Signal Using Convolutional Neural Networks. In *Interspeech*, 2013.

Daniel Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin Cubuk, and Quoc V. Le. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. In *Interspeech*, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems (NeurIPS)*. 2019.

Archontis Politis, Sharath Adavanne, and Tuomas Virtanen. A dataset of reverberant spatial sound scenes with moving sources for sound event Localization and Detection. In *Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, 2020.

Mirco Ravanelli and Yoshua Bengio. Speaker Recognition from Raw Waveform with SincNet. In *In IEEE Workshop on Spoken Language Technology (SLT)*, 2018.

Tara N. Sainath, Brian Kingsbury, Abdel-rahman Mohamed, and Bhuvana Ramabhadran. Learning filter banks within a deep neural network framework. In *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013.

Tara N. Sainath, Ron J. Weiss, Andrew Senior, Kevin W. Wilson, and Oriol Vinyals. Learning the Speech Frontend With Raw Waveform CLDNNs. In *Interspeech*, 2015.

Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. Wav2Vec: Unsupervised Pre-training for Speech Recognition. In *Interspeech*, 2019.

Bob L Sturm. Stéphane mallat: A wavelet tour of signal processing, 2nd edition. *Computer music journal*, 2007.

Harshavardhan Sundar, Weiran Wang, Ming Sun, and Chao Wang. Raw Waveform Based End-to-end Deep Convolutional Network for Spatial Localization of Multiple Acoustic Sources. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Tho Nguyen Thi Ngoc, Ngoc Khanh Nguyen, Huy Phan, Lam Pham, Kenneth Ooi, Douglas Jones, and Woon-Seng Gan. A General Network Architecture for Sound Event Localization and Detection Using Transfer Learning and Recurrent Neural Network. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2021.

T. N. Tho Nguyen, D. L. Jones, and W. Gan. A Sequence Matching Network for Polyphonic Sound Event Localization and Detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Sebastian Thrun. Affine Structure From Sound. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 18, pages 1353–1360. MIT Press, 2006.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jacob Uszkoreit, Llion Jones Jone, Aidan N. Gomez, and Lukasz Kaiser. Attention is All You Need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Juan Manuel Vera-Diaz, Daniel Pizarro, and Javier Macias-Guarasa. Towards End-to-End Acoustic Localization Using Deep Learning: From Audio Signals to Source Position Coordinates. *Sensors*, 2018.

Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip H.S. Torr. Fast Online Object Tracking and Segmentation: A Unifying Approach. In *IEEE/CVF Computer Vision and Pattern Recognition (CVPR)*, 2019.

Qing Wang, Huaxin Wu, Zijun Jing, Feng Ma, Yi Fang, Yuxuan Wang, Tairan Chen, Jia Pan, Jun Du, and Chin-Hui Lee. The USTC-Iflytek System for Sound Event Localization and Detection of Dcase2020 challenge. In *Tech. Report of SELD Decase2020 Challenge*, 2020.

M. Won, S. Chun, O. Nieto, and X. Serrc. Data-Driven Harmonic Filters for Audio Representation Learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Neil Zeghidour, Nicolas Usunier, Iasonas Kokkinos, Thomas Schatz, Gabriel Synnaeve, and Emmanuel Dupoux. Learning Filterbanks from Raw Speech for Phone Recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

Neil Zeghidour, Olivier Teboul, Félix de Chaumont Quitry, and Marco Tagliasacchi. LEAF: A Learnable Frontend for Audio Classification. *International Conference on Learning Representations (ICLR)*, 2021.

A Appendix

A.1 Spatial Location Encoding in Frequency Domain and Recording Format Discussion

We discuss the detailed spatial location encoding for FOA and MIC sound waveforms recording format. The spatial location encoding is based on 2D feature learned by the Gabor filter bank for each sound waveform channel, which can be represented as $\{F_i = (R_i, I_i)\}_{i=1}^{4}$. R_i and I_i are real part and imaginary part feature of the i-th channel sound waveform, respectively.

FOA format is well-known as first-order Ambisonics (B-format). It contains four channels: omni-directional, x-directional, y-directional and z-directional components, respectively. The instantaneous sound intensity vector is often used as spatial location (or phase difference) feature, which can be computed through the cross-spectrum between the omni-directional channel to the remaining x, y, z- directional. As a result, we have obtained 3 channel spatial location encoding feature.

$$IV_x = F_0^* \cdot F_1, \;\; IV_y = F_0^* \cdot F_2, \;\; IV_z = F_0^* \cdot F_3 \tag{4}$$

where F_0^* indicates the conjugate of the omni-directional feature. The three cross-spectrum feature IV_x, IV_y and IV_z are stacked together and further normalized before serving as the spatial encoding feature.

MIC format is well-known as tetrahedral microphone array. The four microphones are mounted in spherical coordinates with four distinct orientations. We treat the four microphones equally and compute the phase difference between any two microphones. Thus a total of six channels spatial location feature can be constructed. Specifically, we choose to compute GCC-PHAT (Brandstein and Silverman, 1997) like cross-spectrum feature. For any two channel m and n, we compute the angle between the real part and imaginary part of the cross-spectrum.

$$SL = angle(F_m^* \cdot F_n), \; m \neq n, m = 1, 2, 3, 4; n = 1, 2, 3, 4 \tag{5}$$

SL indicates the spatial location feature computed by the sound waveforms channel m and n. The $angle(\cdot)$ equals to a frequency amplitude normalization operation, like the GCC-PHAT (Brandstein and Silverman, 1997) does. Please note that all the spatial location feature computation operations are differentiable so the whole neural network becomes end-to-end trainable.

A.2 Synperiodic Filter Bank Frequency-Periodicity Relationship Determination

Mel-scale time-frequency representation has been widely used in both traditional sound feature like MFCC (Davis and Mermelstein, 1980), LogMel and learnable filter bank (Zeghidour et al., 2021). It initializes the filter bank in frequency domain, in which high-frequency filter has wider window length. We transform the filter bank into time domain and can naturally get a roughly logarithmic-scale frequency-periodicity relationship, in which narrower window width is associated with high-frequency filters. We thus set $\omega(w_i, \rho) = 27 \cdot \log_{10}(w_i) - \rho$. We plot our synperiodic filter bank window function and the mel-scale initialized windowing function in Fig. 6, it shows our proposed windowing function naturally approximates the mel-scale windowing function.

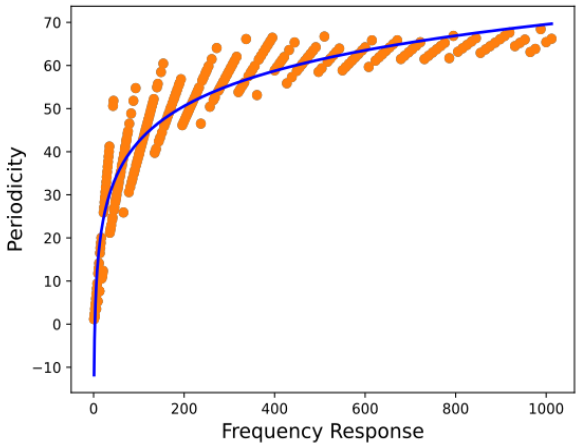

Figure 6: The relationship between filter frequency response and the periodicity. Green curve: our proposed windowing function. Light orange dots: mel-scale initialized frequency-periodicity relationship.

Table 8: SoundSynp-large neural network architecture. The layer follow $name@kernelsize, stride$ format, and synperiodic filter bank follow $name@kernelsize, stride, groups$ format. FC is fully connection layer, AvgPool is the average pooling layer, MaxPool is max-pooling layer. B is the batchsize, T is input waveforms time-length. All convolution layers are followed by a batch normalization layer and Relu activation layer. Please note that since the backbone neural network has two identical branches, we just show one branch here.

layer	filter num	output size
Input: [B,4,T]		
Synperiodic Filter Bank Groups		
SynperiodicFilterBank@1024,600,3	256	[B, 256, T/600, 21]
Backbone Conv block1		
Conv2d@3,1	128	[B, 256, T/600, 128]
Conv2d@3,1	128	[B, 256, T/600, 128]
AvgPool@2,1	None	[B, 128, T/600, 128]
Backbone Conv block2		
Conv2d@3,1	256	[B, 128, T/600, 256]
Conv2d@3,1	256	[B, 128, T/600, 256]
AvgPool@2,1	None	[B, 64, T/600, 256]
Backbone Conv block3		
Conv2d@3,1	256	[B, 64, T/600, 256]
Conv2d@3,1	256	[B, 64, T/600, 256]
AvgPool@2,1	None	[B, 32, T/600, 256]
Backbone Conv block4		
Conv2d@3,1	512	[B, 32, T/600, 512]
Conv2d@3,1	512	[B, 32, T/600, 512]
AvgPool@2,1	None	[B, 16, T/600, 512]
Backbone Conv block5		
Conv2d@3,1	512	[B, 16, T/600, 512]
Conv2d@3,1	512	[B, 16, T/600, 512]
AvgPool@16,1	None	[B, T/600, 512]
Backbone MHSA block1		
MHSA@8,1024	512	[B, T/600, 512]
AvgPool@2,1	None	[B, T/1200, 512]
Backbone MHSA block2		
MHSA@8,1024	512	[B, T/1200, 512]
AvgPool@2,1	None	[B, T/2400, 512]
Backbone MHSA block3		
MHSA@8,1024	512	[B, T/2400, 512]
FC	class num	[B, T/2400, class num]
FC	class num x 3	[B, T/2400, class num x 3]
Trackwise Permutation Invariant Head		
Multi-label Classification	None	scalar
Location Regression	None	scalar

Table 9: SoundSynp-lite neural network architecture. The layer follow $name@kernelsize, stride$ format, and synperiodic filter bank follow $name@kernelsize, stride, groups$ format. FC is fully connection layer, AvgPool is the average pooling layer, MaxPool is max-pooling layer. B is the batchsize, T is input waveforms time-length. All convolution layers are followed by a batch normalization layer and Relu activation layer. It can be easily adjusted to fit other cases. Please note that since the backbone neural network has two identical branches, we just show one branch here.

layer	filter num	output size
Input: [B,4,T]		
Synperiodic Filter Bank Groups		
SynperiodicFilterBank@1024,600,3	256	[B, 256, T/600, 21]
Backbone Conv block1		
Conv2d@3,1	128	[B, 256, T/600, 128]
Conv2d@3,1	128	[B, 256, T/600, 128]
AvgPool@2,1	None	[B, 128, T/600, 128]
Backbone Conv block2		
Conv2d@3,1	256	[B, 128, T/600, 256]
Conv2d@3,1	256	[B, 128, T/600, 256]
AvgPool@2,1	None	[B, 64, T/600, 256]
Backbone Conv block3		
Conv2d@3,1	512	[B, 64, T/600, 512]
Conv2d@3,1	512	[B, 64, T/600, 512]
AvgPool@2,1	None	[B, 32, T/600, 512]
Backbone Conv block4		
Conv2d@3,1	256	[B, 32, T/600, 256]
Conv2d@3,1	256	[B, 32, T/600, 256]
AvgPool@2,1	None	[B, 16, T/600, 256]
Backbone Conv block5		
Conv2d@3,1	256	[B, 16, T/600, 256]
Conv2d@3,1	256	[B, 16, T/600, 256]
AvgPool@16,1	None	[B, T/600, 256]
Backbone MHSA block1		
MHSA@8,1024	256	[B, T/600, 256]
AvgPool@2,1	None	[B, T/1200, 256]
Backbone MHSA block2		
MHSA@8,1024	256	[B, T/1200, 256]
AvgPool@2,1	None	[B, T/2400, 256]
Backbone MHSA block3		
MHSA@8,1024	256	[B, T/2400, 256]
FC	class num	[B, T/2400, class num]
FC	class num x 3	[B, T/2400, class num x 3]
Trackwise Permutation Invariant Head		
Multi-label Classification	None	scalar
Location Regression	None	scalar