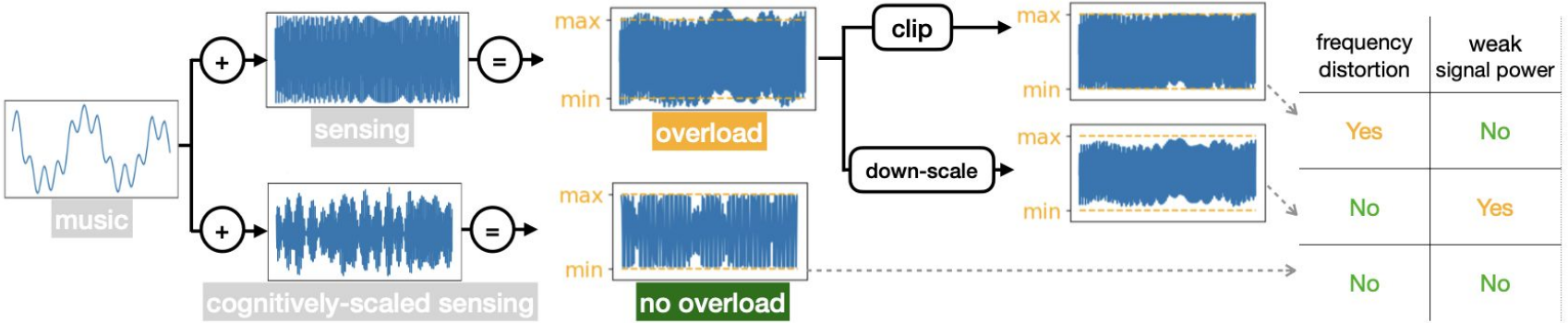
CoPlay: Audio-agnostic Cognitive Scaling for Acoustic Sensing



Outline

1. **Background**: Intro to *acoustic sensing* in smart devices
2. **Challenge**: Playing music and sensing simultaneously in the same speaker
3. **Method**: Cognitive scaling in speaker mixer
4. **Evaluation**: model performance + *user study* result

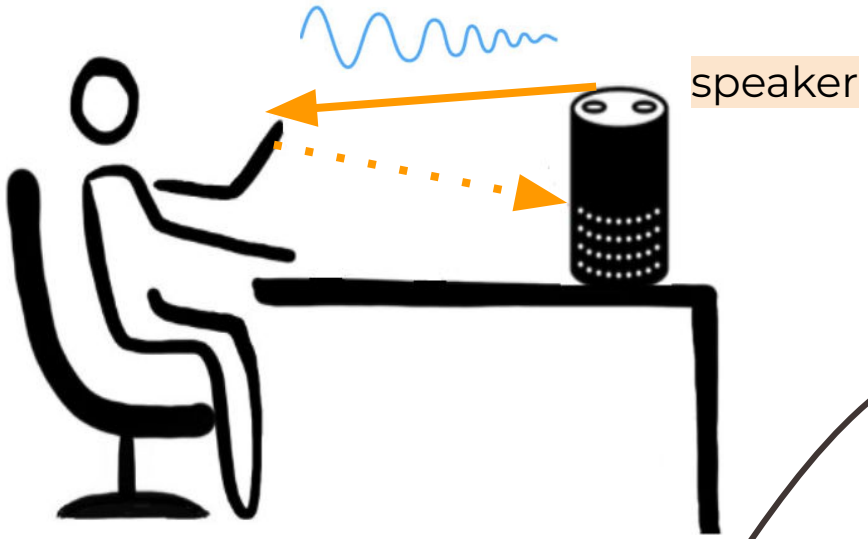
01

Background

Intro to acoustic sensing

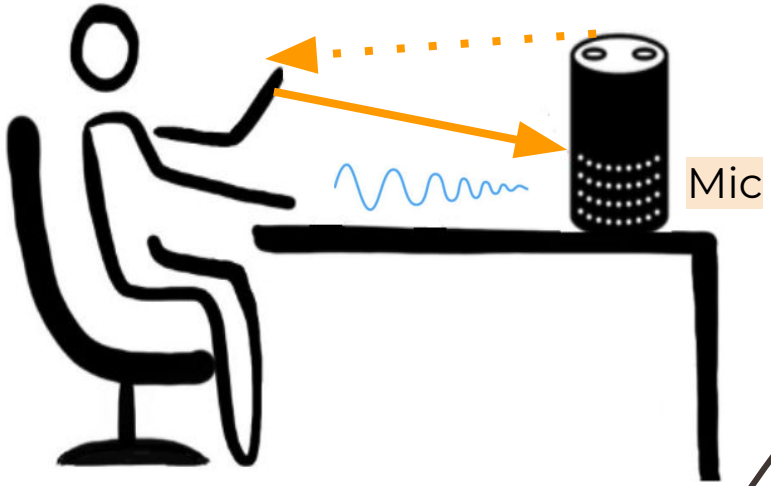
Background
- *1.1 Intro to acoustic sensing*

Transform the smart device into a **SONAR** system by leveraging the **speaker** and **microphone** on device for motion tracking.



Background
- *1.1 Intro to acoustic sensing*

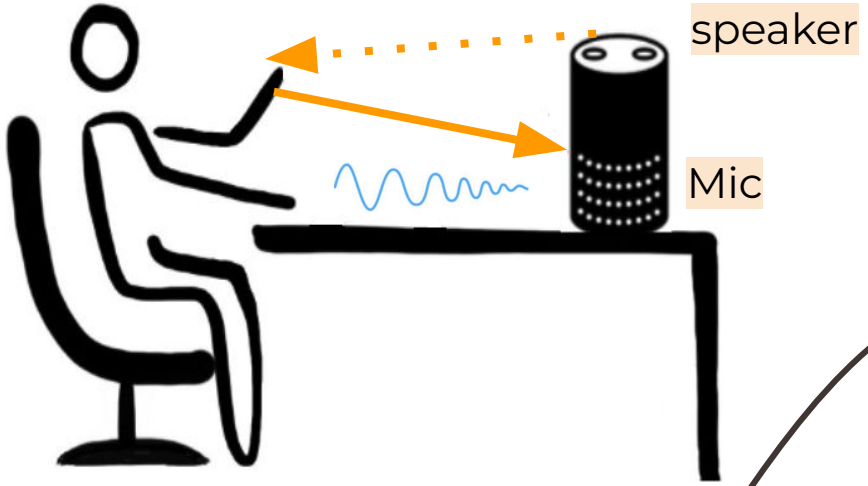
Transform the smart device into a **SONAR** system by leveraging the **speaker** and **microphone** on device for motion tracking.



Background
- *1.1 Intro to acoustic sensing*

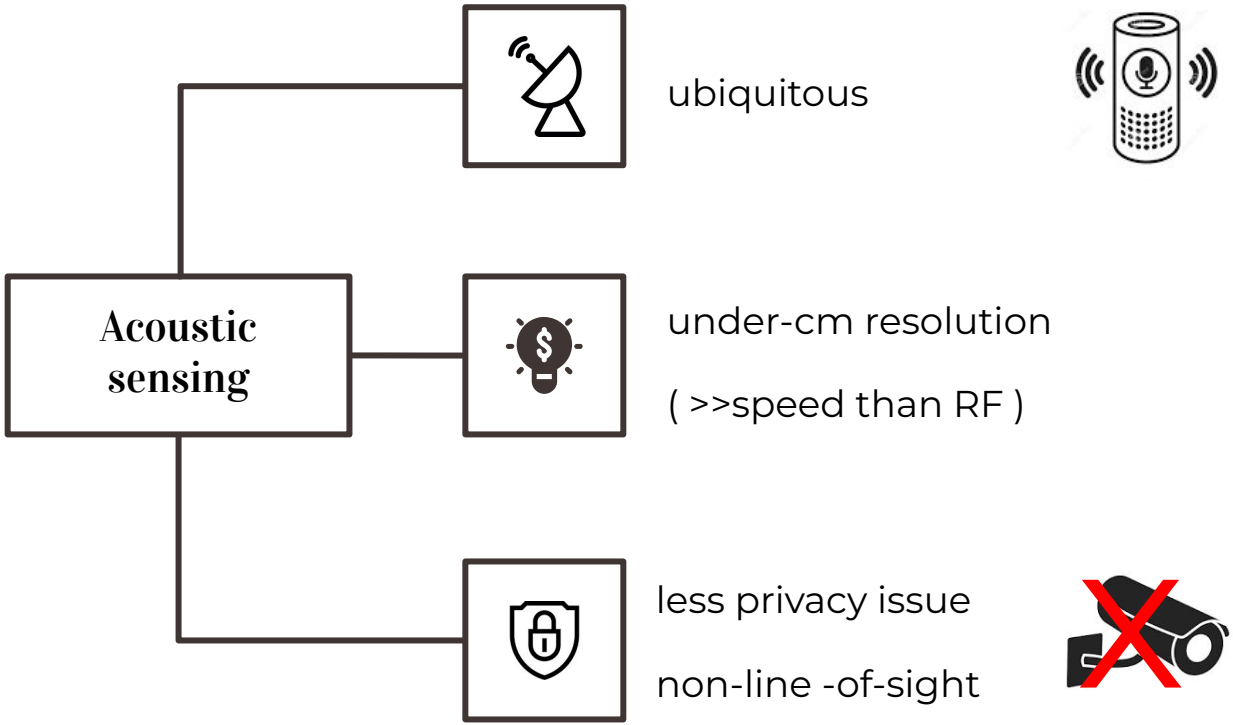
Acoustic/wireless perception **applications**

- gesture recognition
- breath rate detection
- intrusion detection
- VR/AR



Background

- *1.1 Intro to acoustic sensing*



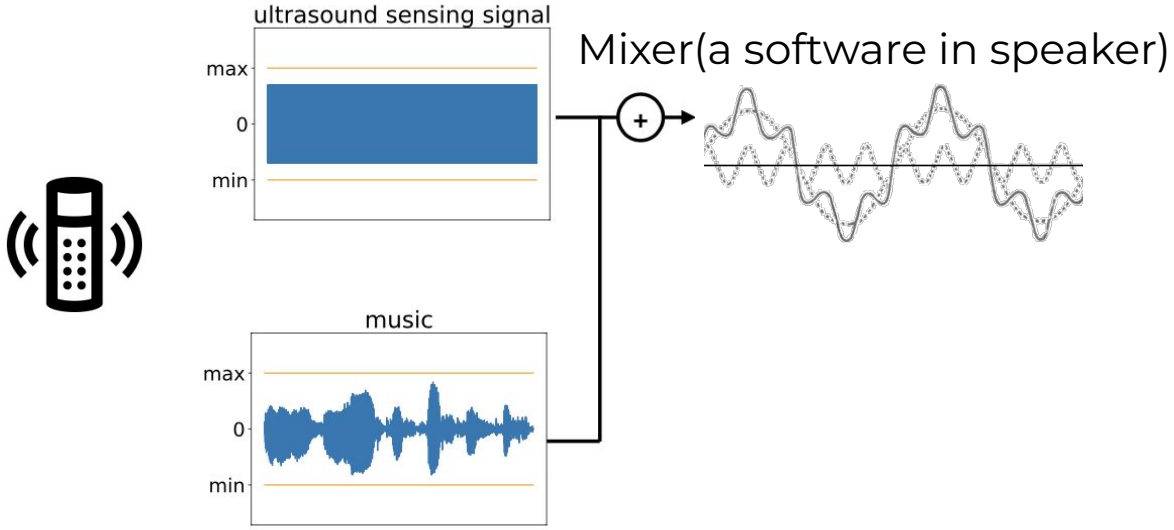
02

Challenge

Overload in mixer when playing music and sensing simultaneously

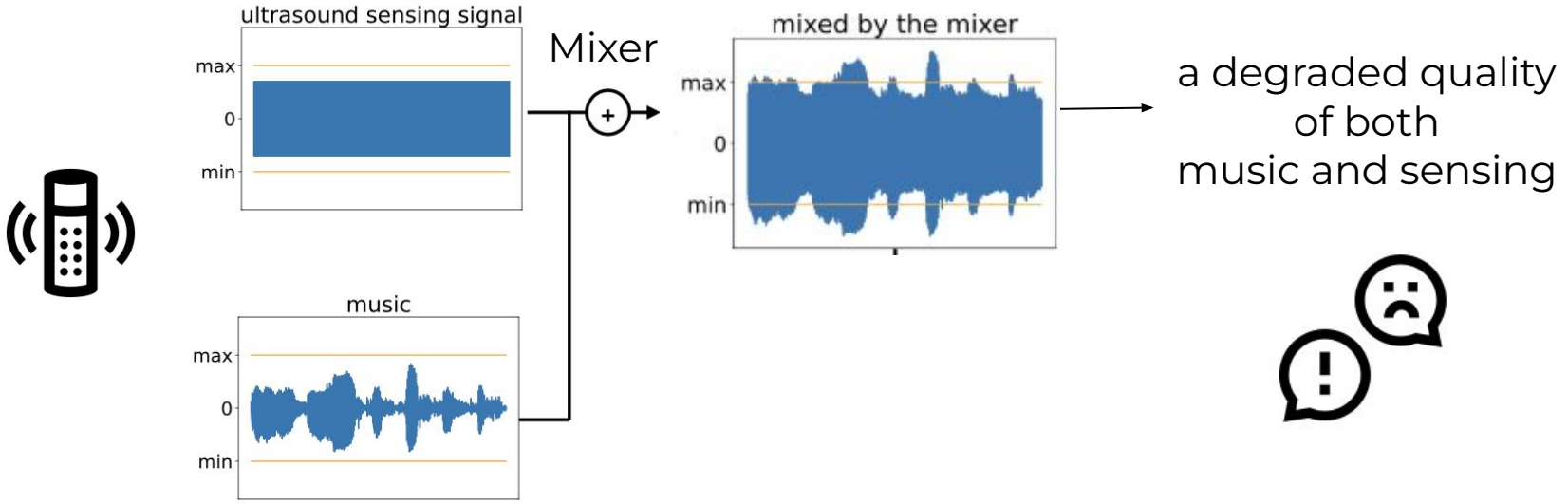
Challenge
- *the **concurrent-music** problem*

Play music and **acoustic sensing signal** on the same speaker



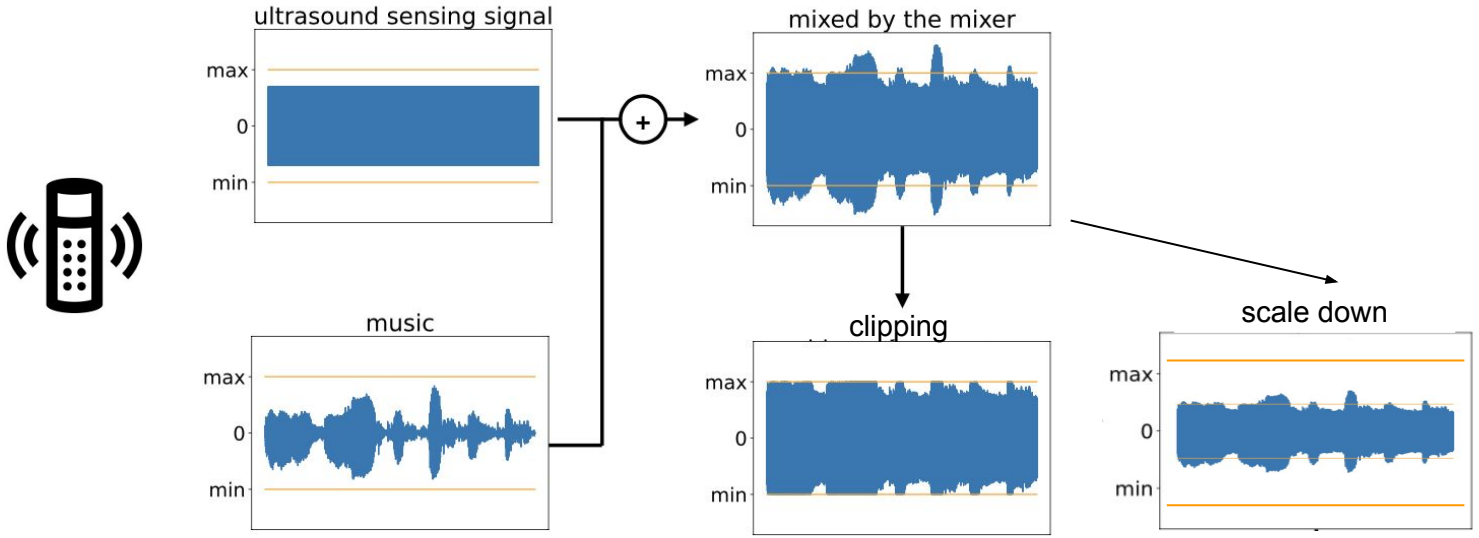
Challenge
- *the **concurrent-music** problem*

Play music and **acoustic sensing signal** on the same speaker



Challenge
- *the **concurrent-music** problem*

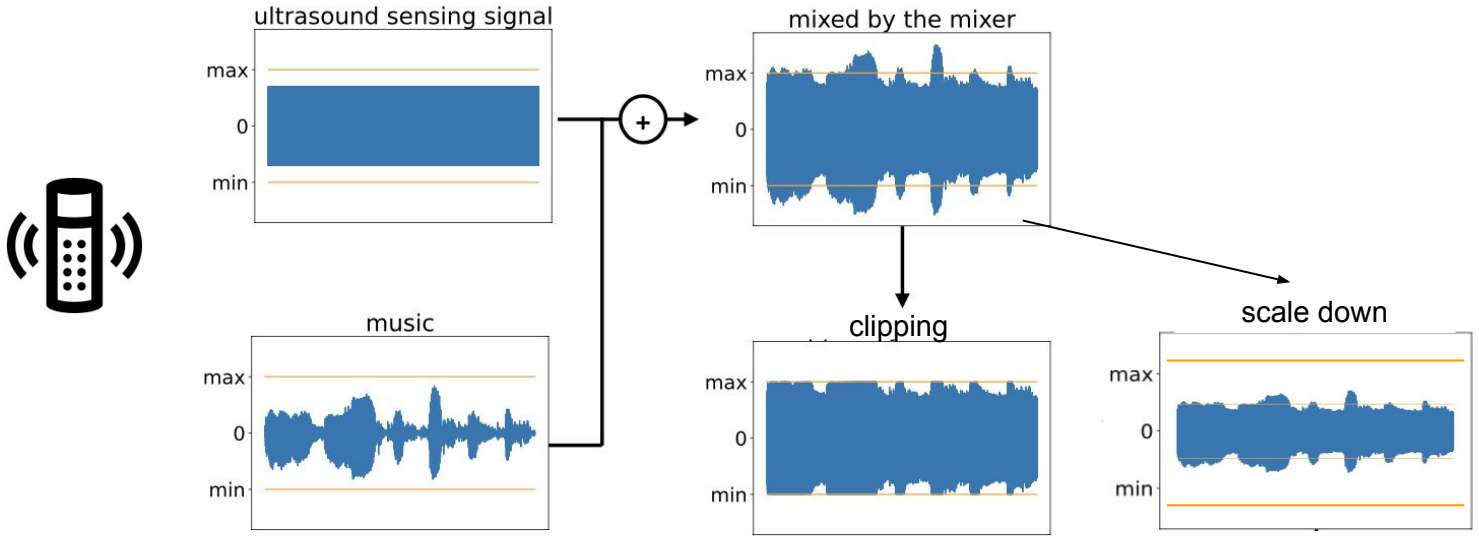
Current solution in OS: clipping or scale down



Challenge
- *the **concurrent-music** problem*

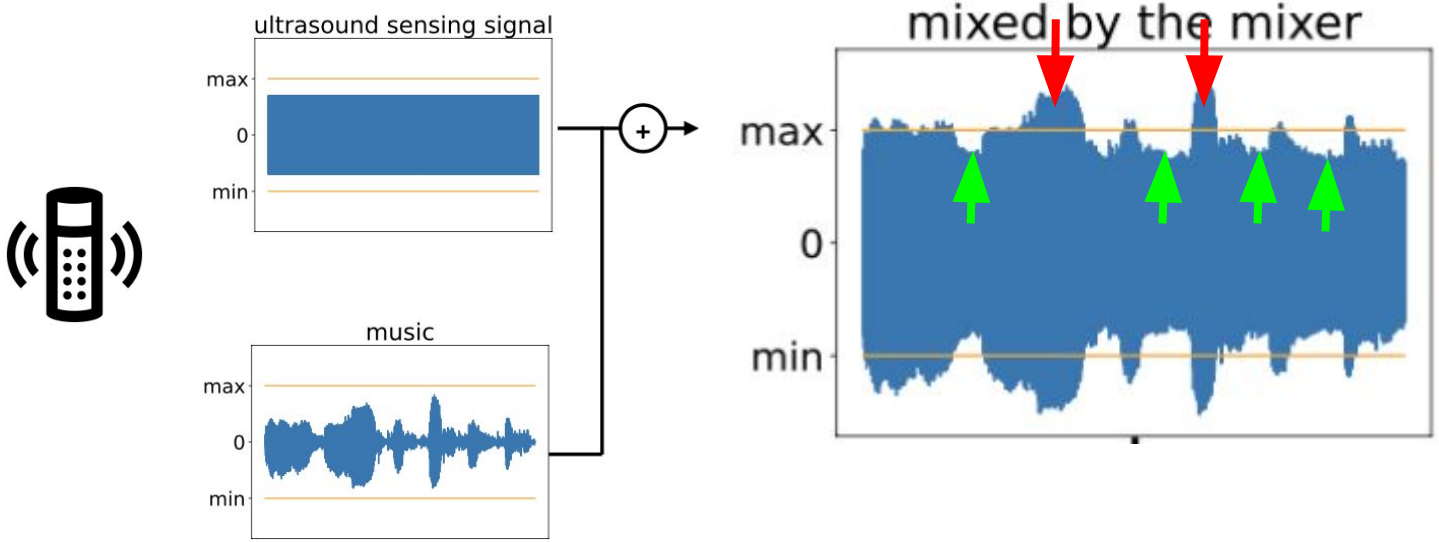
clipping scale down

Most acoustic sensing systems rely on ~~**accurate FFT** at~~ the receiver end & ~~**large power** f~~or long detection range



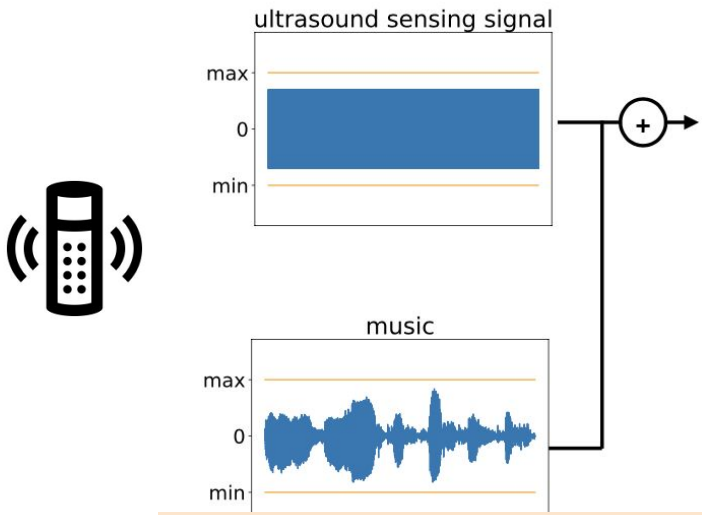
Challenge
- *the **concurrent-music** problem*

Our solution: **cognitive** scaling



Challenge
- *the **concurrent-music** problem*

Our solution: **cognitive** scaling



Maximal power
Minimal distortion

With music untouched

03

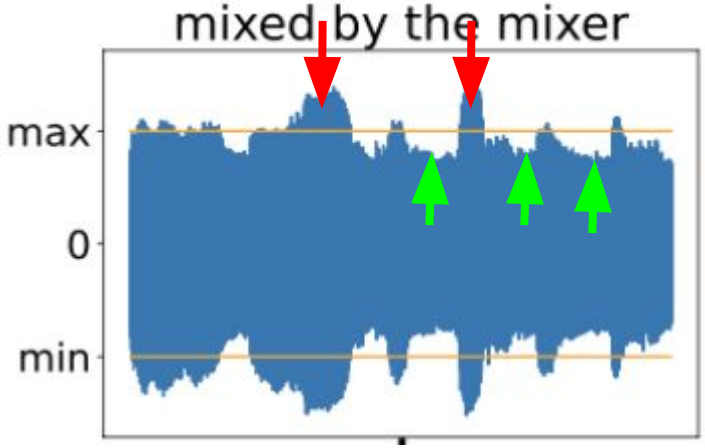
Method

Learning-based cognitive scaling in mixer

Method

Learning-based cognitive scaling in mixer

Our solution: **cognitive** scaling



Maximal power
Minimal distortion

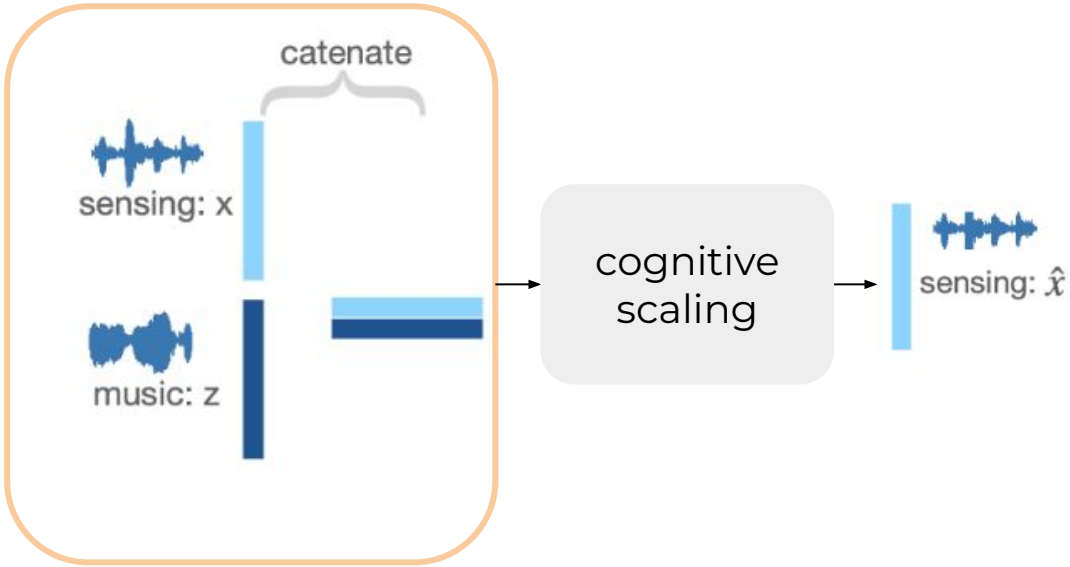
→ **Learning-base**

Method

Learning-based cognitive scaling in mixer

Problem definition:

- Input:
  - sensing signal *x*
  - concurrent music *z*
  - numerical range *m*
- Output:
  - Cognitively scaled $\hat{\mathbf{x}}$

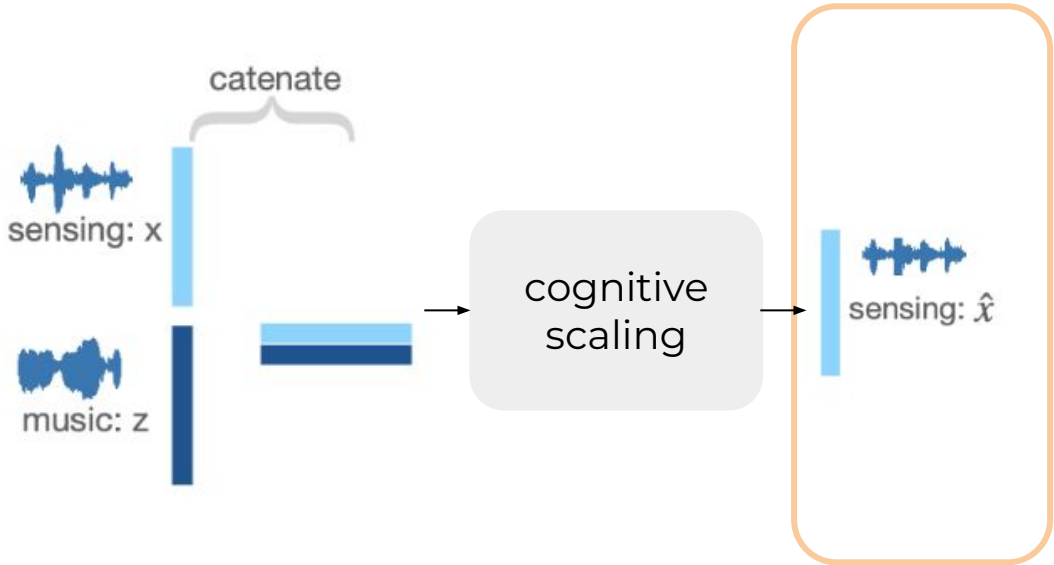


Method

Learning-based cognitive scaling in mixer

Problem definition:

- Input:
  - sensing signal *x*
  - concurrent music *z*
  - numerical range *m*
- Output:
  - Cognitively scaled $\hat{x}$



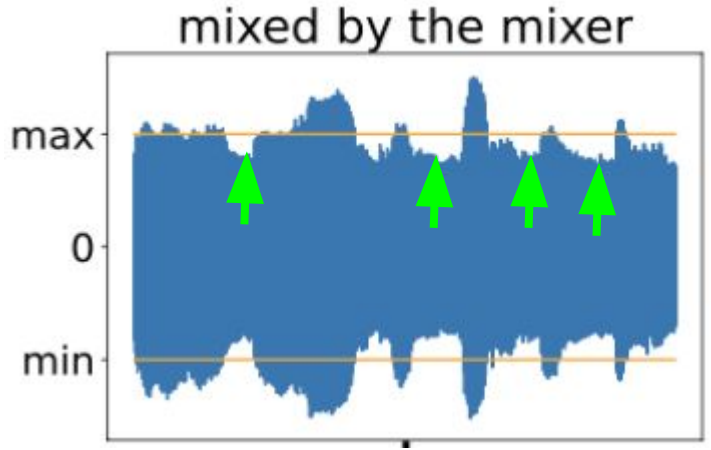
Method

Learning-based cognitive scaling in mixer

Problem definition:

- Loss:
  - $q(\hat{x} + z, m)$ (the difference of the magnitude) ← **Max**imal power **Min**imal distortion
  - $p(x, \hat{x})$ (the difference in frequency domain)

- Input:
  - sensing signal *x*
  - concurrent music *z*
  - numerical range *m*
- Output:
  - Cognitively scaled $\hat{\mathbf{x}}$



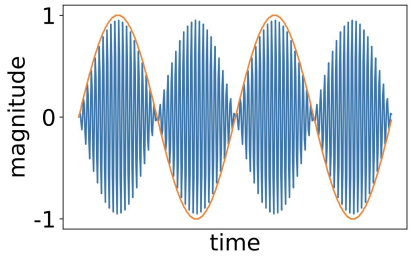
Method

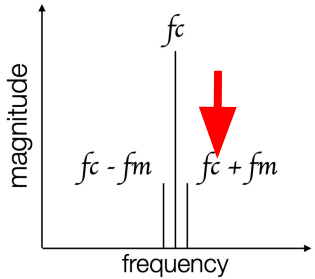
Learning-based cognitive scaling in mixer

Problem definition:

- Loss:
  - $q(\hat{x} + z, m)$ (the difference of the magnitude)
  - $p(x, \hat{x})$ (the difference in frequency domain) ← **Max**imal power **Min**imal distortion

- Input:
  - sensing signal *x*
  - concurrent music *z*
  - numerical range *m*
- Output:
  - Cognitively scaled **X**





Method

Learning-based cognitive scaling in mixer

Problem definition:

- Loss:
  - $q(\hat{x}+z,m)$ (the difference of the magnitude)
  - $p(x,\hat{x})$ (the difference in frequency domain) ← **Max**imal power **Min**imal distortion
  - $s(\hat{x})$ (the variance of all frequency) (FMCW only)
- Input:
  - sensing signal *x*
  - concurrent music *z*
  - numerical range *m*
- Output:
  - Cognitively scaled $\hat{\mathbf{X}}$

Method

Learning-based cognitive scaling in mixer

- Loss:
  - $q(\hat{x}+z,m)$ (the difference of the magnitude)
  - $p(x,\hat{x})$ (the difference in frequency domain)
  - $s(\hat{x})$ (the variance of all frequency) (FMCW only)

$$loss_{sine} = \alpha * p(x,\hat{x}) + \beta * q(\hat{x}+z,m)$$

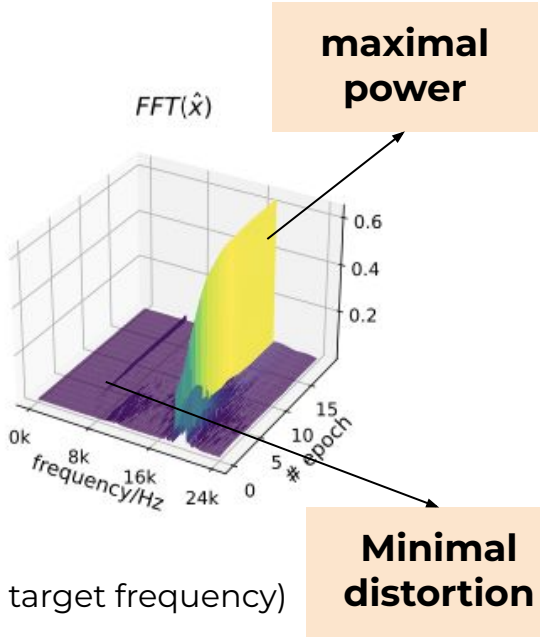
$$loss_{chirp} = \alpha * p(x,\hat{x}) + \beta * q(\hat{x}+z,m) + \gamma * s(\hat{x})$$

Method

Learning-based cognitive scaling in mixer

Problem definition:

- Loss:
  - $q(\hat{x}+z,m)$ (the difference of the magnitude)
  - $p(x,\hat{x})$ (the difference in frequency domain)



l2-norm of the mixed magnitude normalized by *m, N* is window siz

$$q(\hat{x}+z,m)=1-\frac{1}{2N}l^2(\frac{\hat{x}+z}{m})$$

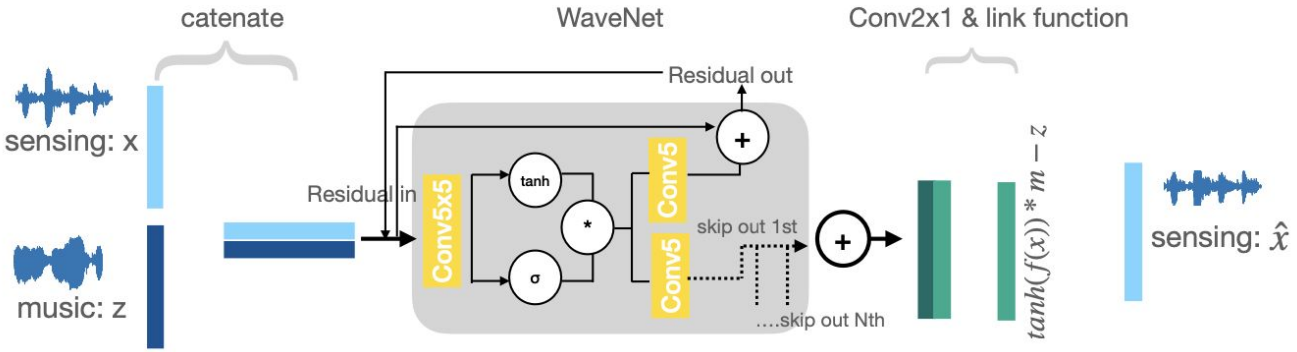
l2-norm of FFT coefficients, (further customize it for sine wave, f_c is the target frequency)

$$p(x,\hat{x})=1-l^2(c_{\hat{f}_c})+\frac{1}{N-1}\sum_{i\in -f_c} l^2(c_i,\hat{c}_i)$$

Method

Learning-based cognitive scaling in mixer

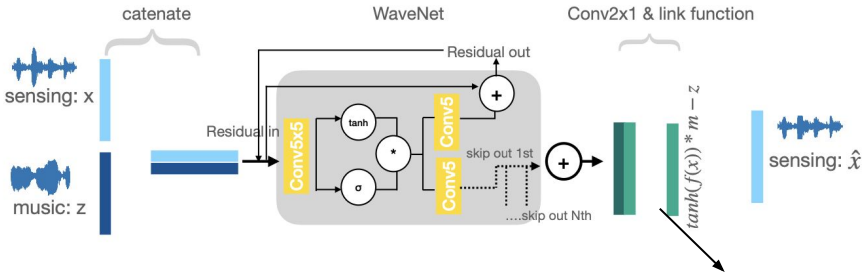
Model structure



Method

Learning-based cognitive scaling in mixer

Link function

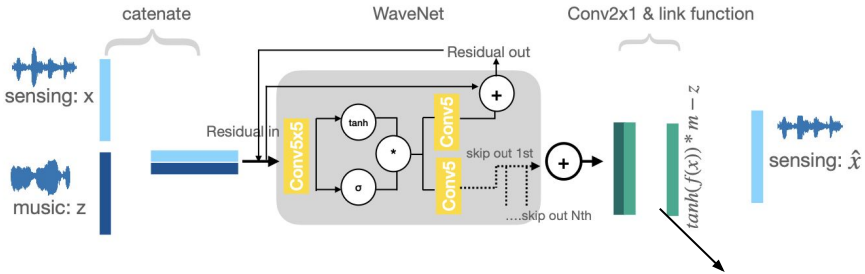


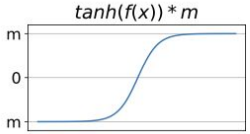
Create windowed-sinc impulse response for given cutoff frequencies.

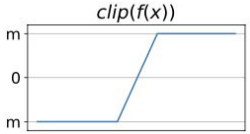
Method

Learning-based cognitive scaling in mixer

Link function







$$\hat{x} = tanh(a) * m - z = \frac{e^{a} - e^{-a}}{e^{a} + e^{-a}} * m - z$$

- match the domain
- encourage large amplitude

03

Evaluation

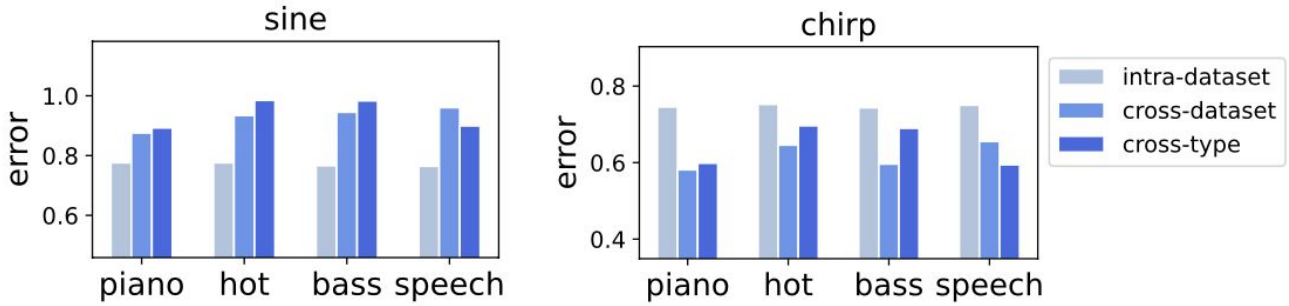
Model performance + User study

Evaluation - part 1: model performance

- **Dataset**
  - **Piano**
    - Beethoven dataset
      - a benchmark music dataset for audio generation.
      - music for a total duration of 10 hours
    - YouTubeMix dataset
      - music dataset with higher-quality recordings than Beethoven
      - a total duration of 4 hours.
  - **Hot songs**: top-ranking Billboard songs from 2019 and 2022. 2h+
  - **Bass**: bass-centric playlists for low-frequency audio. 1h ~ 2h
  - **Speech**: podcast of conversational speech. 1h+
  - (We interpolate all the data to a sample rate of 48k Hz and cast the 8-bit quantization into 16-bit signed integer.)

Evaluation - part 1: model performance

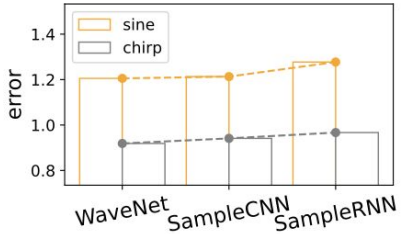
- **Generalization across datasets and music types**



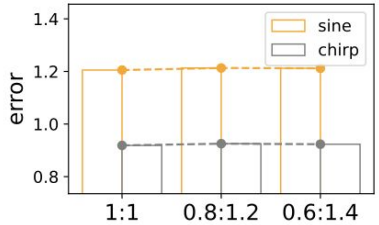
The model generalizes well to unseen datasets and new types of music or speech.

Evaluation - part 1: model performance

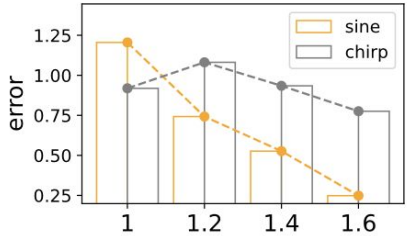
- **Ablation study results**



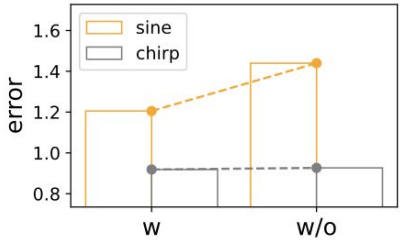
(a) model architecture



(b) sensing volume: music volume



(c) weight of recovery-loss

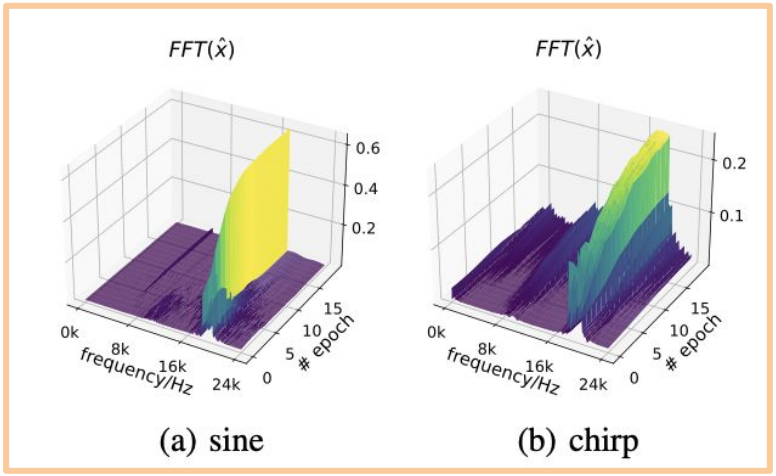


(d) with or without sinc kernel

Evaluation - visualization

Visualization in frequency domain:

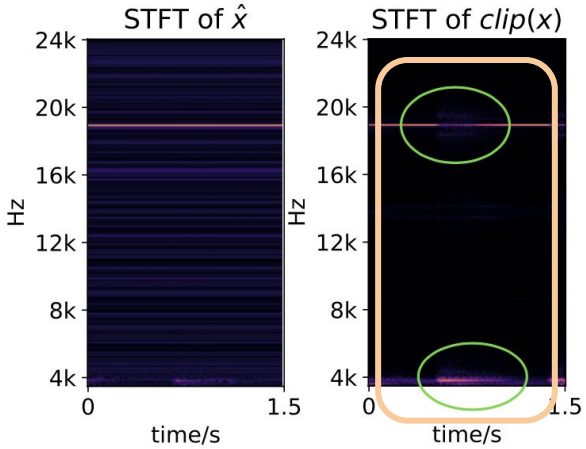
- **Peak gradually converges to target frequency** with neglectable sidebands
- While the clipped(x) has worse distortion as highlighted with the circles



Evaluation - visualization

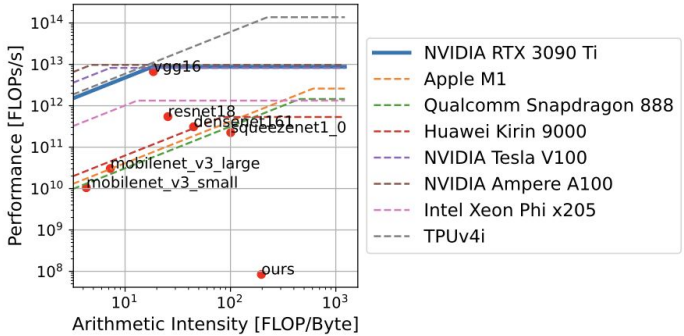
Evaluation/Visualization in frequency domain:

- Peak gradually converges to target frequency with neglectable sidebands
- **While the clipped(x) has worse distortion** as highlighted with the circles



Evaluation - computation efficiency

- Roofline plot of computation cost of our deep learning model



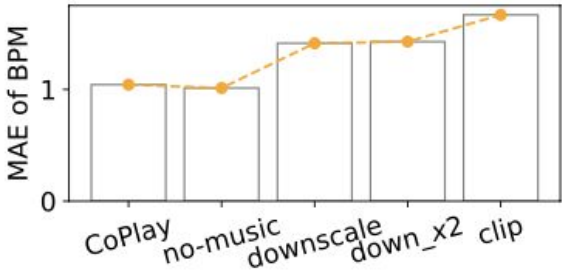
Evaluation - field study

Field study on a downstream tasks

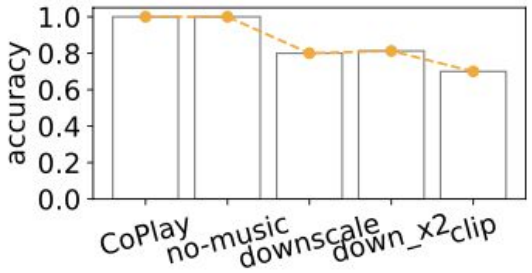
- **Task**: respiration rate detection and gesture classification
- 12 participants, 12 sessions per participant, 1 minute per session
- **Ground truth**: Venier belt with pressure sensor for breath rate
- **Setup**: Android phones and iPhone
- **Signal:** sinewave and FMCW (18K-20KHz)
- **Baseline:** no-concurrent-music, clipping, downscaling (by 1X and 2X)

Evaluation - field study

- Compared with sensing with no music, the baselines degrade sensing performance, while CoPlay does not.



(a) breath rate detection

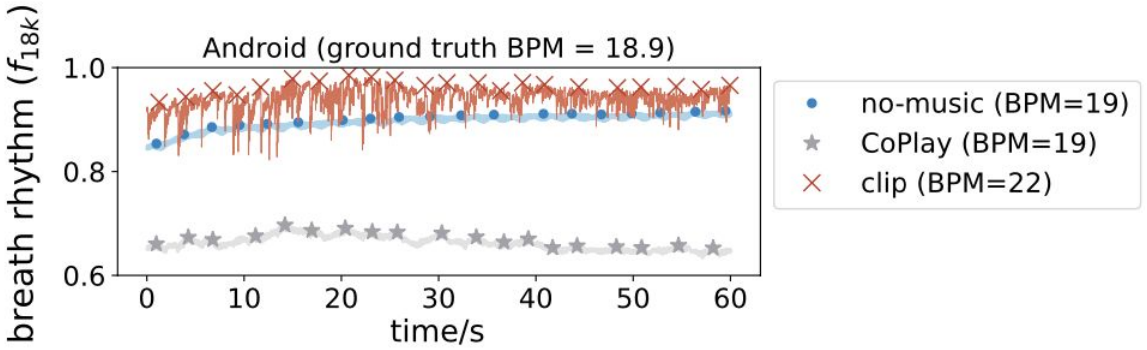


(b) gesture recognition

Evaluation - field study

Break down the results:

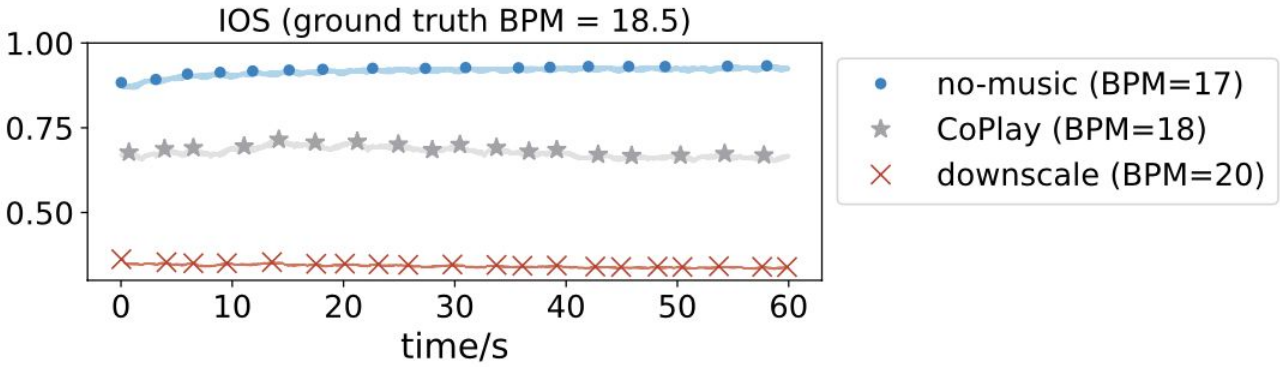
- CoPlay detects the breath rhythm with less noisy and minimal power loss



Evaluation - field study

Break down the results:

- CoPlay detects the breath rhythm with less noisy and minimal power loss



Evaluation - field study

Qualitative study:

- Q1) Did you hear a buzzing noise?
- Q2) Did you hear the music at adequate volume?
- Q3) Did you perceive delay or discontinuity?

	clipping	downscale	CoPlay
Q.1 (5 = buzzing)	4.72	1.13	1.22
Q.2 (5 = loud)	4.14	1.12	4.49
Q.3 (5 = delay)	-	-	1

Future work

- More extensive tests:
  - More downstream tasks, sensing signals, diverse hardwares, etc
- Validation:
  - Computation escalation using multi-channels
  - Explore acceleration methods like model compression and acceleration

Thanks

Q&A

Codebase is to be published on https://github.com/lydhr/CoPlay

Business icon pack