Can audio-visual integration strengthen robustness under multimodal attacks?

Yapeng Tian   Chenliang Xu
University of Rochester

陈仁苗

2022.3.11

McGurk Effect

Do

Robustness of Computational Models

- There is now having developed some computational approaches to achieve robust auditory or visual perception by multisensory integration
  - audio-visual speaker recognition, speech recognition, sound separation, event recognition, etc
- Whether these models still exhibit robustness under attacks?
- Inspired by the auditory-visual illusion in human perception, presenting a systematic study on machines' multisensory integration under attacks

Audio-Visual Robustness under Multimodal Attacks

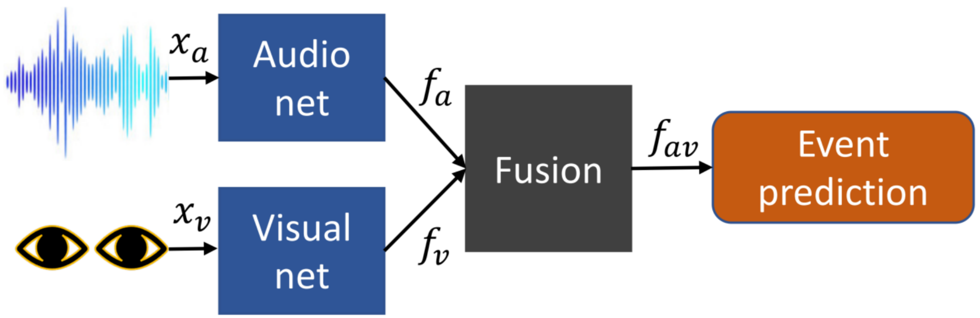
Multimodal attack

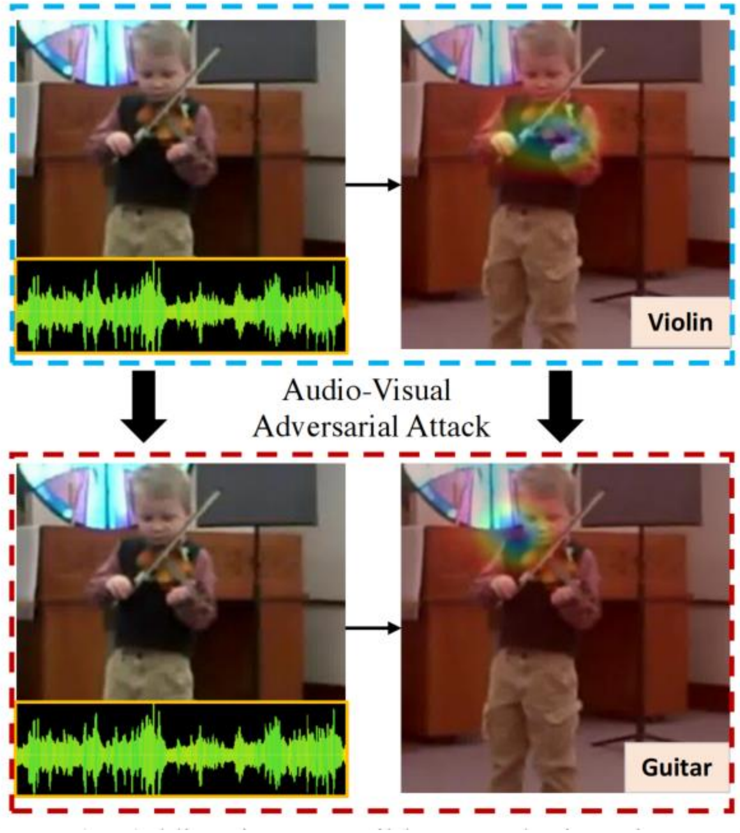
- Goal: to fool the target multimodal model by adding human imperceptible perturbations into its inputs from multiple modalities
- Two types: single-modality attack and audio-visual attack
- Adversarial objective:

$$\begin{aligned} &\underset{x_a^{adv}, x_v^{adv}}{\operatorname{argmax}} \mathcal{L}(x_a^{adv}, x_v^{adv}, y; \theta) \\ &\text{s.t.} \quad ||x_a^{adv} - x_a||_p \leqslant \epsilon_a \\ &\qquad\; ||x_v^{adv} - x_v||_p \leqslant \epsilon_v \end{aligned}$$

Audio-Visual Robustness under Multimodal Attacks

- Audio-visual event recognition as a proxy task

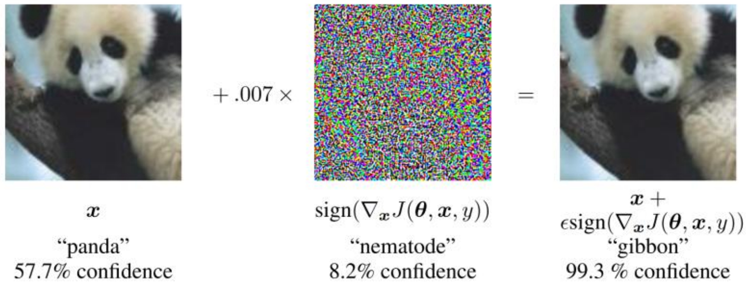




Experiments

- Attack methods
  - Fast Gradient Sign Method (FGSM)

$$x_a^{adv} = x_a + \epsilon_a \cdot \text{sign}(\nabla_{x_a} \mathcal{L}(x_a, x_v, y; \theta))$$



  - Projected Gradient Descent (PGD)
    - Iterative variant of FGSM

  - Momentum-based Iterative Method (MIM)
    - integrates a momentum term into the iterative process to further stabilize update directions and mitigate local minima

Experiments

- Datasets
  - MIT-MUSIC
    - 520 videos in 11 instrument categories
    - Clean audio-visual synchronized musical recordings

  - Kinetics-Sound
    - 15,000+ 10s YouTube Videos in 27 human action categories
    - More diverse events rather than only musical instruments
    - More noisy (audio and visual content inside some videos might not be related)

  - AVE
    - contains 4143 videos covering 28 event categories and video
    - temporally labeled with audio-visual event boundaries

- Metric
  - Recognition accuracy

Audio-Visual Robustness under Multimodal Attacks

Dataset	Attack	✓AV	✗A	✗V	✗AV	Avg.	Unimodal ✓A	Unimodal ✓V
MM	FGSM [30]	88.46	50.00	25.00	15.38	30.12	59.62	81.73
	PGD [45]		13.46	**1.92**	**0.00**	5.09		
	MIM [17]		**6.73**	**1.92**	**0.00**	**2.88**		
KS	FGSM [30]	72.42	33.38	15.08	8.18	18.88	35.99	66.08
	PGD [45]		6.22	1.90	0.77	2.96		
	MIM [17]		**3.87**	**1.55**	**0.32**	**1.91**		

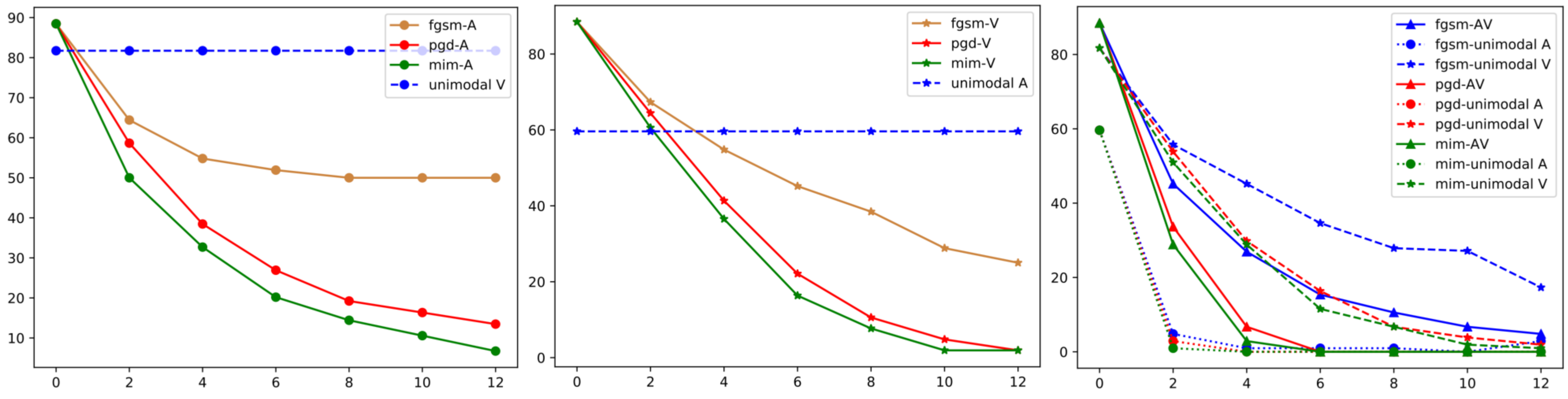
Observations:

- Clean AV models are better than both clean A and V models
- AV models under single-modality attacks might achieve worse performance than unimodal models.
- AV attacks make models even worse

Conclusion:

- A joint perception is not always better than individual perceptions under attacks

Adversarial robustness against multimodal attacks on the MIT-MUSIC. The x-axis denotes the attack strength.



(a) Audio Attack (b) Visual Attack (c) Audio-Visual Attack

- An unreliable modality could weaken perception by the other modality in audio-visual models

Attacked Audio-Visual Event Recognition Results

helicopter

violin

violin

helicopter

chainsaw

dog barking

Different Fusions under Attacks

Method	✓AV	✗A	✗V	✗AV	Avg.
Sum	88.46	35.58	45.19	3.85	43.27
Concat	88.46	**51.92**	**45.19**	**15.38**	**50.24**
FiLM [57]	83.65	28.85	39.42	3.85	38.95
Gated-Sum [39]	**89.42**	33.65	44.23	4.81	43.03
Gated-Concat [39]	**89.42**	45.19	43.27	13.46	47.84

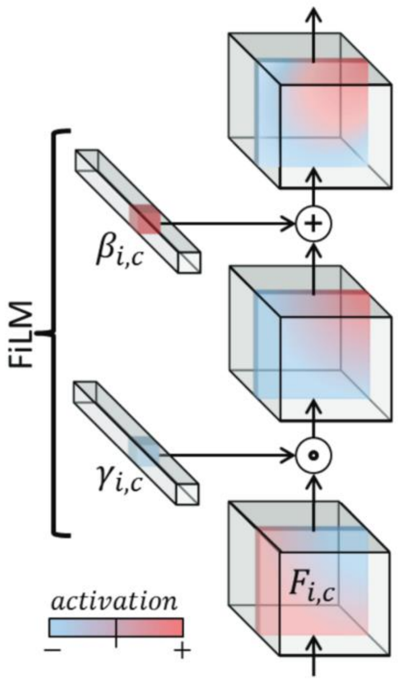
- FiLM $f_{av} = \alpha(f_a) \cdot f_v + \beta(f_a)$

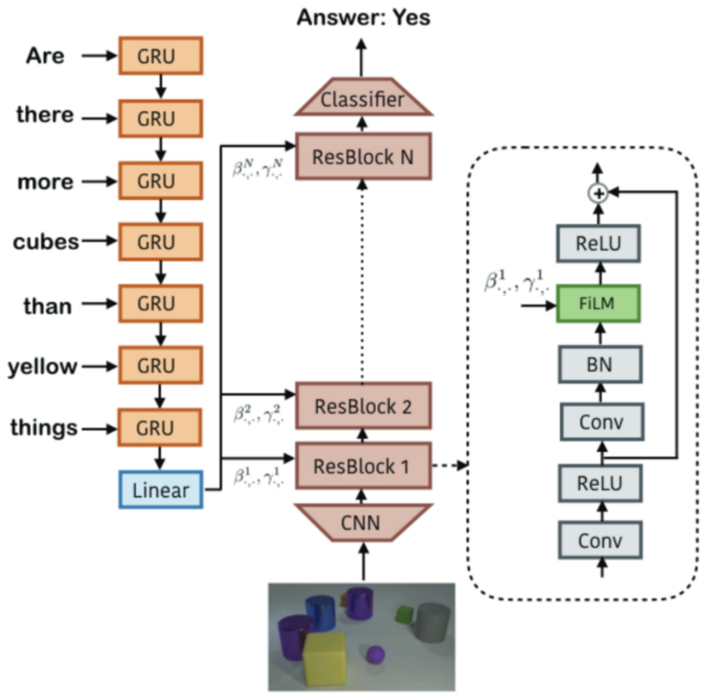
- Gated-Sum

$$f_1 = \sigma(f_a) \cdot f_v,$$
$$f_2 = \sigma(f_v) \cdot f_a,$$
$$f_{av} = f_1 + f_2$$

- Gated-Concat

$$f_{av} = [f_1; f_2]$$



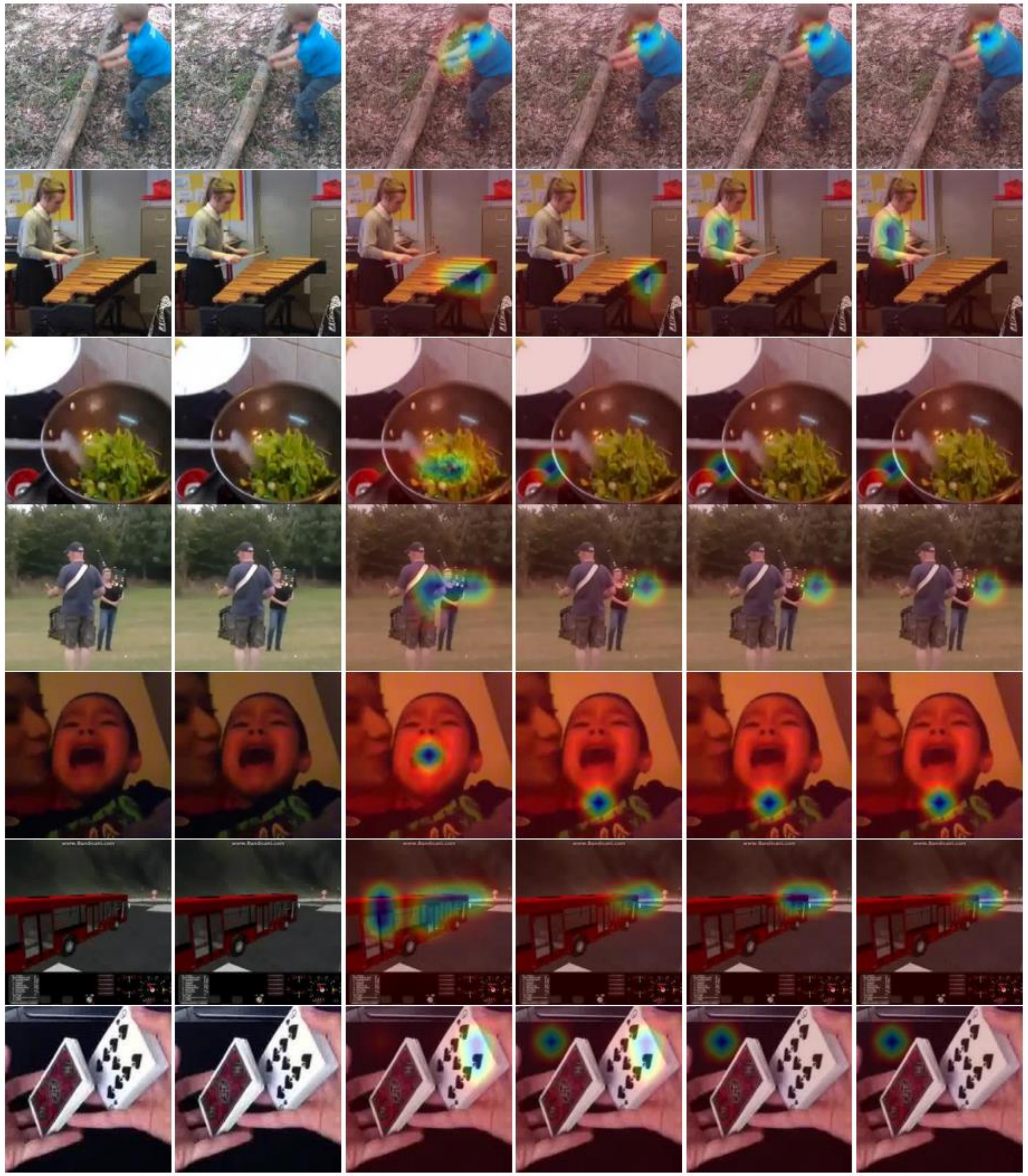


Different Fusions under Attacks

Method	✓AV	✗A	✗V	✗AV	Avg.
Sum	88.46	35.58	45.19	3.85	43.27
Concat	88.46	**51.92**	**45.19**	**15.38**	**50.24**
FiLM [57]	83.65	28.85	39.42	3.85	38.95
Gated-Sum [39]	**89.42**	33.65	44.23	4.81	43.03
Gated-Concat [39]	**89.42**	45.19	43.27	13.46	47.84

- AV models with different fusions achieve competitive performance on attack-free inputs.
- But, all of the models with different fusions are vulnerable to attacks

Visualize Sound Sources under Attacks



Input Frame | Attacked Frame | w/o Attack | Audio Attack | Visual Attack | AV Attack

Audio-Visual Defense

- To encourage unimodal intra-class compactness of AV models, proposing to minimize audio-visual similarity

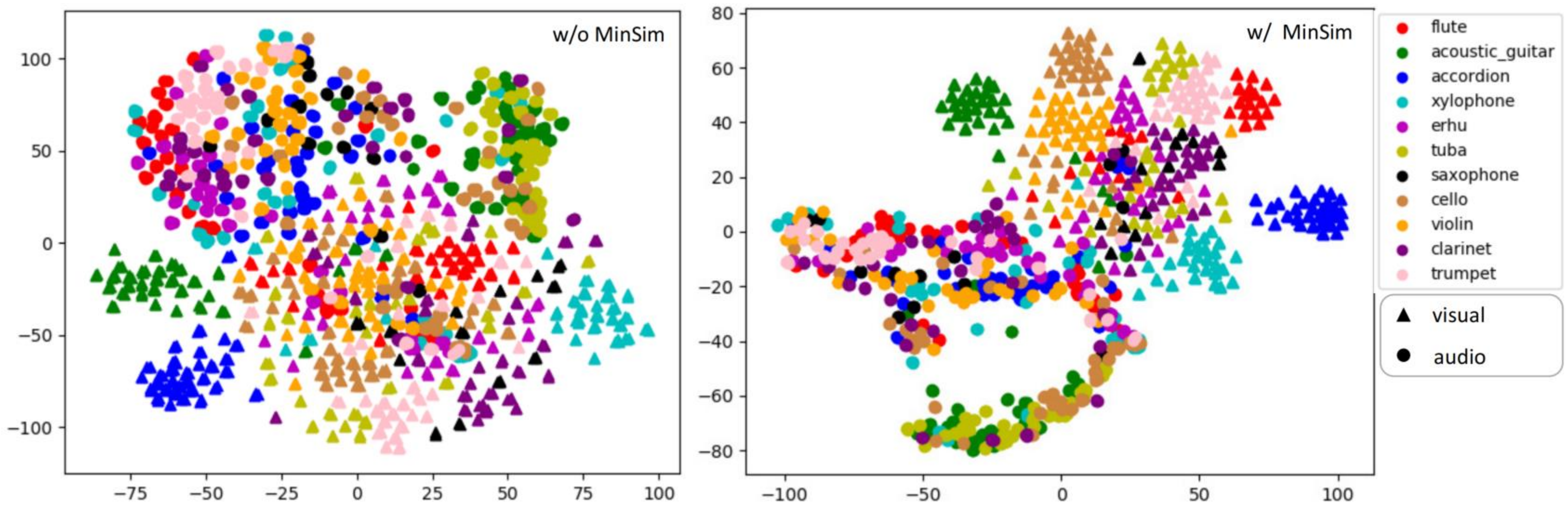
$$\mathcal{L}_{Sim} = \frac{f_a \cdot f_v}{max(||f_a||_2 \cdot ||f_v||_2, \eta)}$$

- Full modal is optimized by a joint objective function

$$\mathcal{L} = \mathcal{L}_{CE} + \mathcal{L}_{Sim}$$

  - With the second term, the model will tend to learn separated audio and visual embeddings
  - The first term will still urge the features to be discriminative, which will implicitly encourage the both separated unimodal embeddings to be more compact and separable

Audio-Visual Defense



With the constraint, the model learns more compact and separable unimodal embeddings

Audio-Visual Defense

Audio and Visual feature denoising

- Using external memory bank to restore cleaner features

$$\min_{\alpha_a} ||f_a^{adv} - M_a\alpha_a||_2^2 + \lambda_a||\alpha_a||_1$$
$$\min_{\alpha_v} ||f_v^{adv} - M_v\alpha_v||_2^2 + \lambda_v||\alpha_v||_1$$

Defense Results

- Relative improvement (RI) metric

$$\text{Avg} = \frac{1}{3}(\text{✗A} + \text{✗V} + \text{✗AV})$$

$$\text{RI} = (\text{✓AV}_m + \text{Avg}_m) - (\text{✓AV}_n + \text{Avg}_n)$$

- Avoid a shortcut when audio-visual defense

Defense (MUSIC)	✓AV	✗A	✗V	✗AV	Avg	RI
None	88.46	51.92	45.19	15.38	37.50	0.00
Unimodal A	59.62	0.00	59.62	0.00	19.87	-46.47
Unimodal V	81.73	**81.73**	11.54	11.54	34.94	-9.29
PCL [51]	83.65	**81.73**	37.50	36.54	51.91	9.60
MaxSim	89.42	52.88	45.19	31.73	43.27	6.73
MinSim	**91.35**	70.19	46.15	36.54	50.96	16.35
ExFMem	89.42	53.85	50.00	20.19	41.34	4.80
MinSim+ExFMem	90.38	73.08	**53.85**	**42.31**	**56.41**	**20.83**
Defense (Kinetics)	**✓AV**	**✗A**	**✗V**	**✗AV**	**Avg.**	**RI**
None	72.42	36.40	26.35	8.09	23.61	0.00
Unimodal A	35.99	1.87	35.99	1.87	13.24	-46.80
Unimodal V	66.08	**66.08**	18.72	18.72	34.50	4.55
PCL [51]	64.50	63.43	29.28	**28.67**	**40.46**	8.93
MaxSim	71.39	34.95	29.57	21.46	28.66	4.02
MinSim	70.88	52.42	28.12	21.62	34.05	8.99
ExFMem	**72.71**	41.56	29.93	10.44	27.31	3.99
MinSim+ExFMem	71.33	55.96	**30.57**	24.90	37.14	**12.44**

- **Advantage:**
  - The structure of article is novel and completive, begin with confirm problem exists by a lot of means, and then propose the method to solve it.
  - It provide a visualize experiment to show the reason for attack.
- **Disadvantage:**
  - The audio use waveforms and the architecture of the network is too simple, and I think it maybe cannot exact a good feature.
- **Inspiration:**
  - The ways to attack modal, fusion and defense.
- **Feature work:**
  - How to deal with situation with losing one of the modality?
  - Whether it will influent in speaker identification task?

```
self.features = \
    nn.Sequential(
    # block 1
    nn.Conv1d(1, 64, kernel_size=3, stride=2, padding=1),
    nn.BatchNorm1d(64),
    nn.ReLU(),
    nn.Conv1d(64, 64, kernel_size=3, stride=2, padding=1),
    nn.BatchNorm1d(64),
    nn.ReLU(),
    nn.MaxPool1d(kernel_size=2, stride=2),
    # block 2
    nn.Conv1d(64, 128, kernel_size=3, stride=2, padding=1),
    nn.BatchNorm1d(128),
    nn.ReLU(),
    nn.Conv1d(128, 128, kernel_size=3, stride=2, padding=1),
    nn.BatchNorm1d(128),
    nn.ReLU(),
    nn.MaxPool1d(kernel_size=2, stride=2),
    # block 3
    nn.Conv1d(128, 256, kernel_size=3, stride=2, padding=1),
    nn.BatchNorm1d(256),
    nn.ReLU(),
    nn.Conv1d(256, 256, kernel_size=3, stride=2, padding=1),
    nn.BatchNorm1d(256),
    nn.ReLU(),
    nn.MaxPool1d(kernel_size=2, stride=2),
    # block 4
    nn.Conv1d(256, 512, kernel_size=3, stride=2, padding=1),
    nn.BatchNorm1d(512),
    nn.ReLU(),
    nn.Conv1d(512, 512, kernel_size=3, stride=2, padding=1),
    nn.BatchNorm1d(512),
    nn.ReLU(),
    nn.MaxPool1d(kernel_size=2, stride=2),
)
```

Thank you!