ICASSP 2018 @ Calgary, Canada

17 April, 2018

AASP-L1.4

Deep Clustering with Gated Convolutional Networks

◎Li Li[1,2], Hirokazu Kameoka[1]

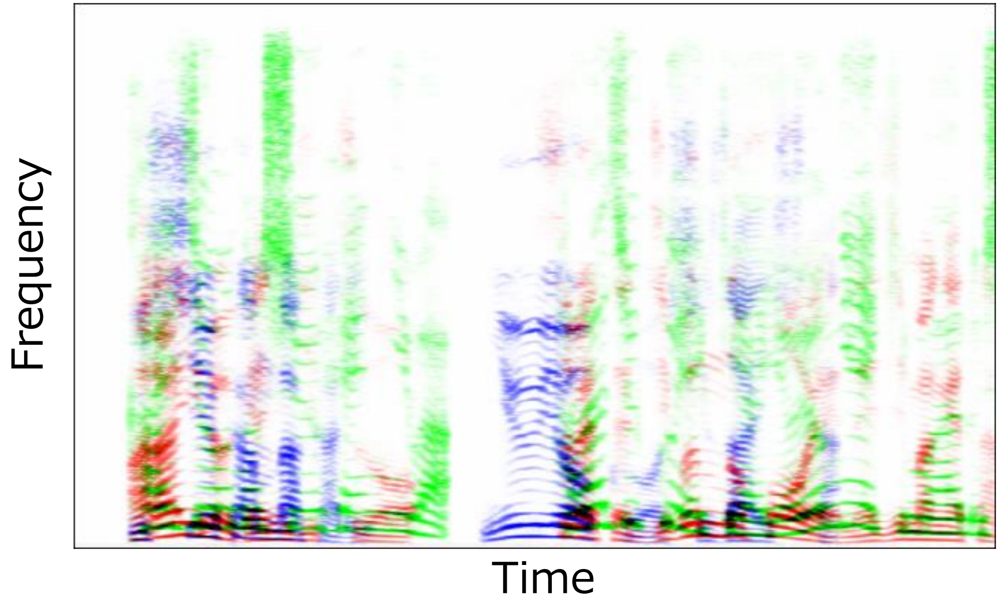
[1] NTT Communication Science Laboratories, NTT Corporation, Japan

[2] University of Tsukuba, Japan

Research background

- **Multi-speaker separation:**
  — Separating all sources from observed multi-speaker mixture signals
  — Very important for e.g., ASR, hearing aids, ⋯
  — Assumption: spectrograms of speech signals are sparse

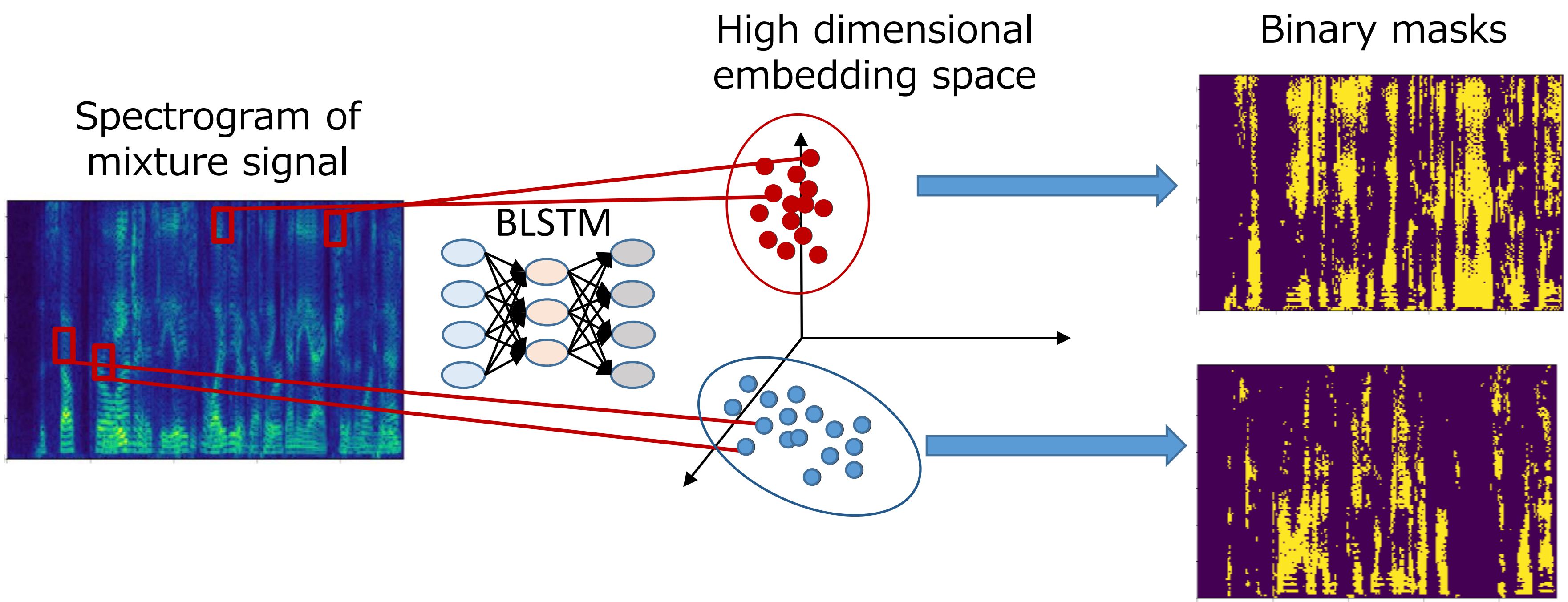
Binary mask estimation



- **Deep Neural Network (DNN)-based methods:**
  [Wang2014, Xu2014, Hershey2016, Kolbak2017]
  — Remarkable separation performance
  — Significantly improved single channel separation performance

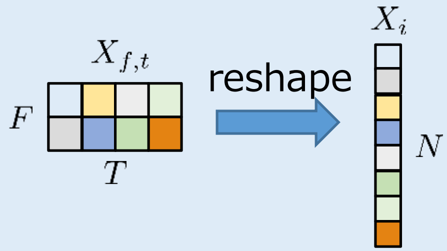
Deep Clustering [Hershey+2016]

- **Deep Clustering (DC)**
  - Greatly improved **speaker-independent** multi-speaker separation
  - Theoretically able to handle arbitrary number of sources
  - **Label permutation invariance:**
    speaker labels DO NOT need to be consistent over different utterances



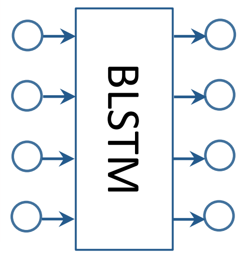
Learn DC network [Hershey+2016]

TF representation of observed signal:
$\mathbf{X} = \{X_i\}_{i=1,\dots,N}$

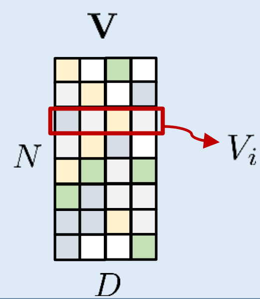


$(N = F \times T)$

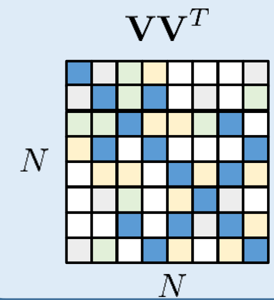
Nonlinear function $g_\theta(\bullet)$



D-dim embedding vector $V_i = g_\theta(X_i)$, where $||V_i|| = 1$



Estimated affinity matrix $\mathbf{V}\mathbf{V}^T$, where $\mathbf{V} = \{V_i\}_{i=1,\dots,N}$



- Objective function :

$$\mathcal{J}(\mathbf{V}) = ||\underbrace{\mathbf{V}\mathbf{V}^T}_{N \times N} - \underbrace{\mathbf{Y}\mathbf{Y}^T}_{N \times N}||_F^2$$

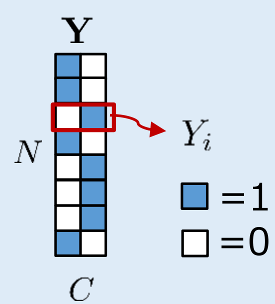
$$= \underbrace{||\mathbf{V}^T\mathbf{V}||_F^2}_{D \times D} - 2\underbrace{||\mathbf{V}^T\mathbf{Y}||_F^2}_{D \times C} + \underbrace{||\mathbf{Y}^T\mathbf{Y}||_F^2}_{C \times C}$$

Minimize squared Frobenius norm

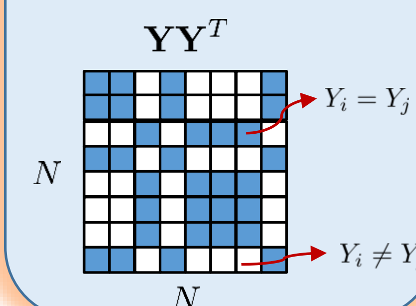
e.g., consider to embed a 8kHz mixture signal with C=2, F=129, T=100 (about 1.6s) to an embedding space with D=40

$$N \times N = 12900 \times 12900 = 166410000$$
$$D \times D = 40 \times 40 = 1600$$
$$D \times C = 40 \times 2 = 80$$
$$C \times C = 2 \times 2 = 4$$

Source label $\mathbf{Y} = \{Y_i\}_{i=1,\dots,N}$



Binary affinity matrix $\mathbf{Y}\mathbf{Y}^T$

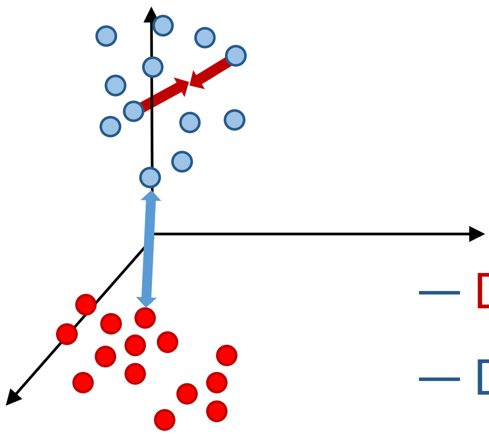


Objective function

- **Objective function for DC [Hershey+2016] :**

$$\text{minimize} \quad \mathcal{J}(\mathbf{V}) = ||\mathbf{V}\mathbf{V}^T - \mathbf{Y}\mathbf{Y}^T||_F^2$$

$$= \underbrace{\sum_{i,j:y_i=y_j} (||V_i - V_j||^2 - 1)} + \underbrace{\sum_{i,j} < V_i, V_j >^2}$$



— Dominated by the same source ➡ become parallel

— Dominated by different sources ➡ become orthogonal

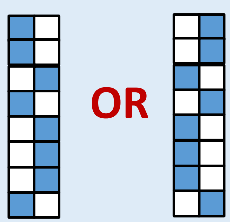
— Using $\mathbf{Y}\mathbf{Y}^T$ instead of directly using $\mathbf{Y}$ for training

➡ **Label permutation invariance**

Source label $\mathbf{Y} = \{Y_i\}_{i=1,\dots,N}$

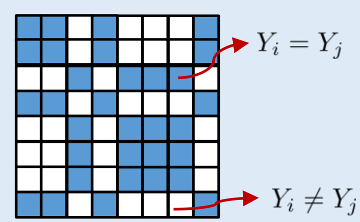
■ =1
□ =0

$\mathbf{Y}$ **OR** $\mathbf{Y}$



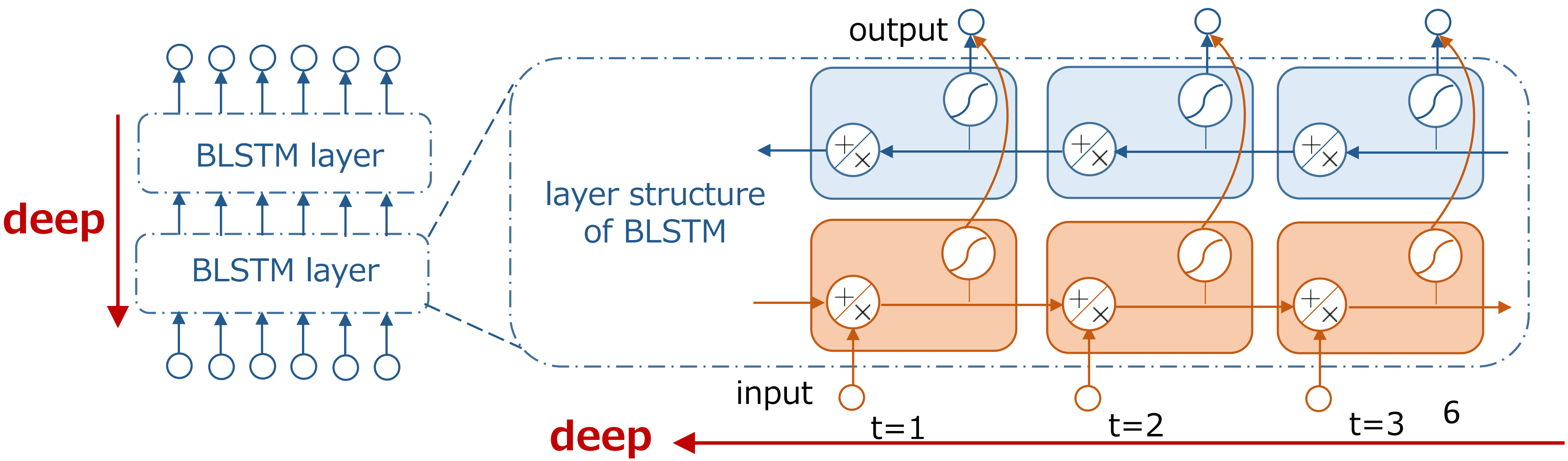
Binary affinity matrix $\mathbf{Y}\mathbf{Y}^T$

$\mathbf{Y}\mathbf{Y}^T$



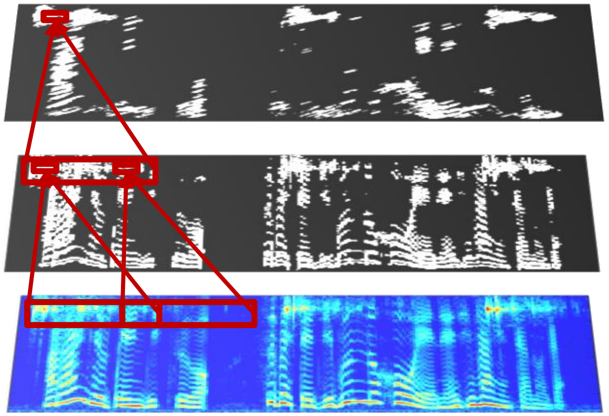
BLSTM for DC embedding

- Deep clustering uses bidirectional long short-term memory (BLSTM) to model the embedding process [Hershey+2016].
- Bidirectional long short-term memory (BLSTM):
  - A kind of recurrent neural networks (RNNs)
  - Natural choice for modeling time series data
  - Training becomes challenging when network becomes deeper
  - Difficult to employ parallel implementations



Gated convolutional networks

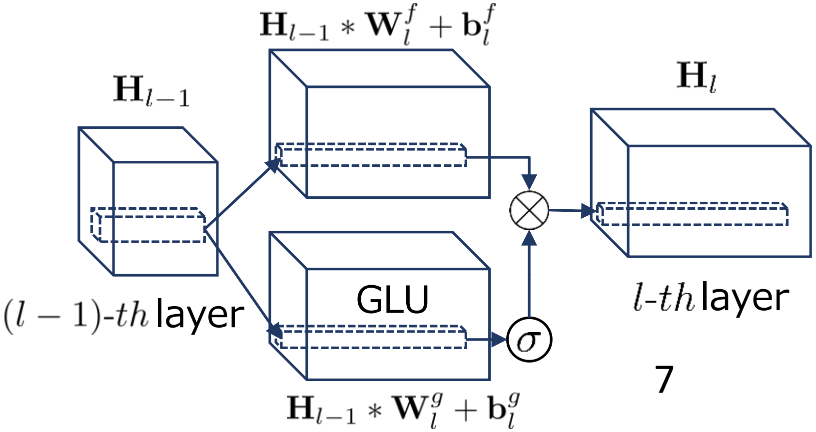
- **Convolutional neural networks** (CNNs):
  - Practically much easier to train
  - Less prone to overfitting
  - Well suited to parallel implementations



- **Gated convolutional networks** [Dauphin 2016]:
  - Formulation:

$$\mathbf{H}_l = (\mathbf{H}_{l-1} * \mathbf{W}_l^f + \mathbf{b}_l^f) \otimes \sigma(\mathbf{H}_{l-1} * \mathbf{W}_l^g + \mathbf{b}_l^g)$$

  - Data-driven gate mechanism: Gated Linear Units (GLUs)
  - Excellent potential for capturing long-term dependencies of time series data
  - Suitable for modeling spectrograms since spectrograms have region dependency



Objectives of this work

This work proposes adopting CNN-based architectures for modeling the embedding process of deep clustering.

Proposed method:
Gated Convolutional Deep Clustering (GCDC)

- We aim to answer ···

Q1: what kind of CNN-based architecture is appropriate for DC ?

1. 1D convolution or 2D convolution
2. Dilated CNN
3. Strided CNN
4. Skip architecture

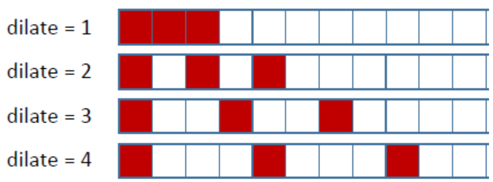
Q2: is it possible to train the model using small amount of dataset ?

5 network architectures

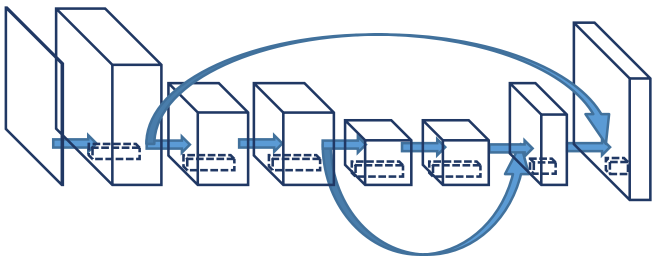
- 1D convolution or 2D convolution:

	Input	Output	filter
1D convolution	Size: 1xT / Channel: F	Size: 1xT / Channel: FxD	$(1, k_T)$
2D convolution	Size: FxT / Channel: 1	Size: FxT / Channel: D	(k_F, k_T)

- Dilated CNN
  - Dilating zeros to handle wider receptive fields



- Strided CNN (Bottleneck)
- Skip architecture
  - Combining output with lower layer outputs



- **Investigated network architectures:**

#1	2D, B, w/o skip	2D convolution / strided CNN
#2	2D, B, w/ skip	2D convolution / strided CNN / skip architecture
#3	2D, DC	2D convolution / dilated CNN
#4	1D	1D convolution
#5	1D, DC	1D convolution / dilated CNN

Speaker-Independent Multi-speaker Separation Experiments

Experimental conditions

- Data: Wall Street Journal (WSJ0)

Full: Training/Validation/Test data	30h/ 10h/ 5h
Sub: Training/Validation/Test data	5.5h/ 0.5h/ 5h
Input SNRs	[0, 10] dB
Sampling rate	8 kHz

- Experimental settings

Window length / shift	254 / 127 sample points
Dimension of embedding vector	20 / 40
Optimizer	Adam
Minibatch size	8 or 16
Learning rate	0.0005

- Evaluation: signal-to-distortion ratio improvement (SDRi) [dB]

Separation performance

- 2-speaker separation:

model		Training dataset	
		Sub (5.5 h)	Full (30 h)
proposed	2D, B, w/o skip	3.90	5.49
	2D, B, w/ skip	3.78	5.23
	2D, DC	5.78	**6.78**
	1D	3.49	5.16
	1D, DC	3.94	**6.36**
conventional (baseline)	BLSTM (our implementation)	1.57	2.46
	BLSTM, 600 nodes, 2L [1]	-	**5.7**

- 3-speaker separation:

model		Full (30 h)
proposed	2D, DC	**3.14**
	1D, DC	2.48
conventional (baseline)	BLSTM, 600 nodes, 2L [1]	**2.2**

[1] J. R. Hershey et al., ICASSP, pp. 31-35, 2016.

Separation performance

- 2-speaker separation:

model		Training dataset	
		Sub (5.5 h)	**Full (30 h)**
proposed	2D, B, w/o skip	3.90	5.49
	2D, B, w/ skip	3.78	5.23
	2D, DC	5.78	**6.78**
	1D	3.49	5.16
	1D, DC	3.94	**6.36**
conventional (baseline)	BLSTM (our implementation)	1.57	2.46
	BLSTM, 600 nodes, 2L [1]	-	**5.7**

- **Models using dilated CNNs outperformed the baseline.**
- **2D, DC showed the capability to perform well even only limited scale dataset being provided.**

[1] J. R. Hershey et al., ICASSP, pp. 31-35, 2016.

Comparison of various embedding dimensions

- 2-speaker separation (full training dataset, single GPU)

model		**Embedding dimension**	
		D=20	**D=40**
proposed	2D, DC	6.78	**6.71**
	1D, DC	6.36	6.39
conventional	BLSTM, 600 nodes, 2L [1]	5.7	6.0
	BLSTM, 600 nodes, 4L [2] (fine-tuned, very deep)	-	**9.4**

2.7dB

- **Increasing embedding dimension from 20 to 40 did NOT improve 2-speaker separation performance.**
- **About 3 dB lower than the deeper and fine-tuned BLSTM-based model.**

[1] J. R. Hershey et al., ICASSP, pp. 31-35, 2016.
[2] Y. Isik et al., Interspeech, pp. 545-549, 2016.

Deeper architectures

- 2-speaker separation

model			Training data Sub (5.5h)	Training data Full (30h)	Computational cost
Proposed	2D, DC	5L	5.78	6.78	1 GPU / 1 day
		8L	6.77	8.32	2 GPUs / 2 days
		14L	7.26	**9.07**	4 GPUs / 3 days
conventional	BLSTM, 600 nodes, 2L [1]		-	6.0	About 1 week (our implementation)
	BLSTM, 600 nodes, 4L [2] (fine-tuned, very deep)		-	**9.4**	

- **The proposed method achieved a comparable result to [2].**
- **The proposed method can be trained quickly even with deep architectures.**

[1] J. R. Hershey et al., ICASSP, pp. 31-35, 2016.
[2] Y. Isik et al., Interspeech, pp. 545-549, 2016.

Deeper architectures

- 2-speaker separation

model			Training data		Computational cost
			Sub (5.5h)	**Full (30h)**	
Proposed	2D, DC	5L	5.78	6.78	1 GPU / 1 day
		8L	6.77	8.32	2 GPUs / 2 days
		14L	7.26	9.07	4 GPUs / 3 days
conventional	BLSTM, 600 nodes, 2L [1]		-	6.0	About 1 week (our implementation)
	BLSTM, 600 nodes, 4L [2] (fine-tuned, very deep)		-	9.4	

model	Training data	SDRi [dB]	Computational cost
2D, DC, 14L	**1h**	5.56	4 GPUs / **4 hours**

[1] J. R. Hershey et al., ICASSP, pp. 31-35, 2016.
[2] Y. Isik et al., Interspeech, pp. 545-549, 2016.

Conclusions

- **Proposed method:**
  - **Gated Convolutional Deep Clustering (GCDC)**
  - Using gated convolutional networks to model embedding process of deep clustering
- **Appropriate architecture for multi-speaker separation tasks**
  - Gated convolutional networks / 2D convolution / Dilated CNN
  - Highest score: 9.07dB
- **GCDC can be trained quickly and perform well even only limited dataset available**
  - 1h training data / 4GPUs / 4hours / 5.56dB
- **Future work:**
  - Much deeper architectures
  - Fine-tuning the models
  - Application of gated convolutional networks to Deep Attractor Networks (DANet)[Chen+2017]

Thank you for your attention!