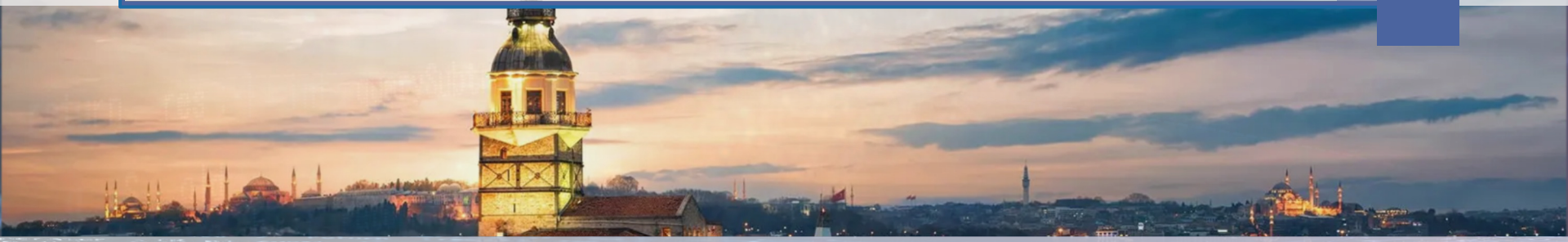
Integration of Physics-Based and Data-Driven Models for Parameter Estimation: with Applications to Image and Speech Signal Processing

Jie Chen* **Xiuheng Wang†** **Ziye Yang***

*School of Artificial Intelligence,

Northwestern Polytechnical University, China.

† Université de Lorraine, CRAN, CNRS, France.



IEEE MLSP 2025 Tutorial Istanbul, Turkey August 31st, 2025

Contributors

Prof. Jie Chen

Dr. Xiuheng Wang

Ziye Yang

Dr. Min Zhao

Dr. Longbin Yan

Dr. Shuaikai Shi

Dr. Wenxing Yang

Linruize Tang

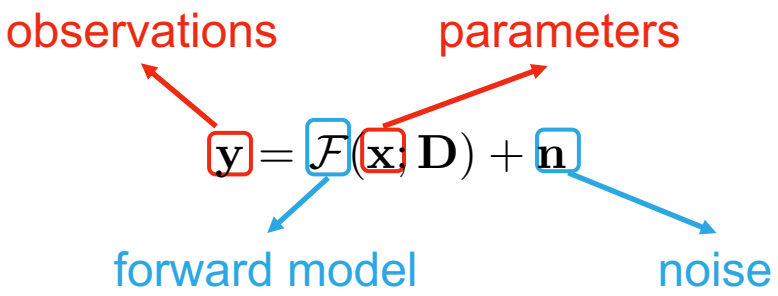
Kai Xie

Haonan Hu

Chengbo Chang

Parameter Estimation

Scientists and engineers frequently wish to relate physical parameters characterizing a model to collected observations[1]:



Three fundamental problems:

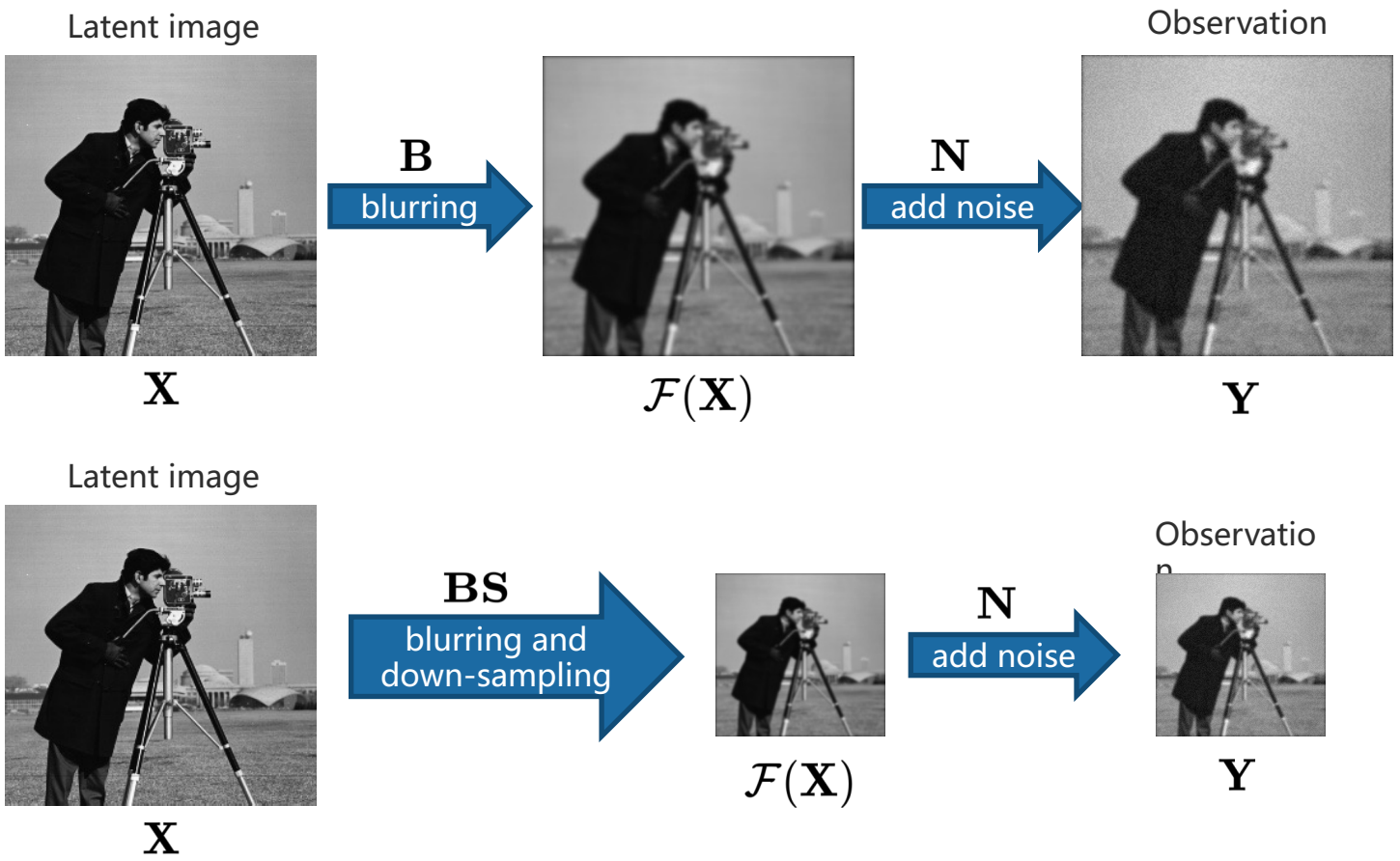
- Forward problem: to find $\mathbf{y}$ given $\mathbf{x}$ (and other information or data $\mathbf{D}$);
- Parameter estimation (inverse) problem: to find $\mathbf{x}$ given $\mathbf{y}$ (and $\mathbf{D}$);
- System identification: to determine $\mathcal{F}(\cdot)$ given $\mathbf{x}$ and $\mathbf{y}$ (and $\mathbf{D}$).

[1] R. Aster, B. Borchers, and C. Thurber, "Parameter Estimation and Inverse Problems", Elsevier, 2011.

Parameter Estimation

Example 1 - image processing :

Recover the latent clean image from its degraded observation.



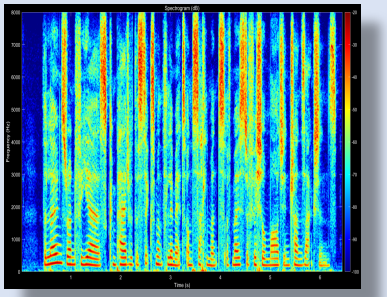
Parameter Estimation

Example 2 – Speech signal processing:

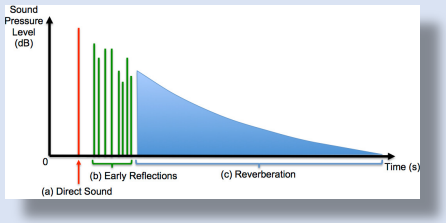
Recover the clean speech from its corrupted observation.

speech dereverberation

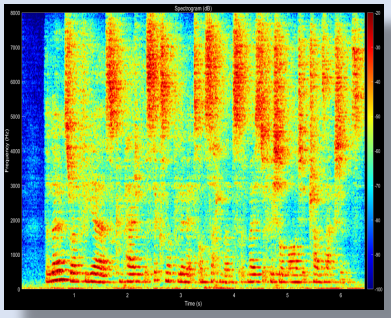
Clean speech x



$\mathcal{F}(x) = x * h$

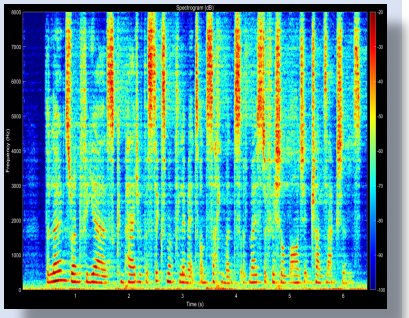


Room impulse response (RIR) h



Reverberation speech y

Processing



Dereverberated speech $\hat{x}$

Objective: to improve speech clarity and intelligibility

Physics-based Models

Grounded in expert knowledge, solved via numerical methods, e.g.,

- Inverse filtering / deconvolution;
- Beamforming;
- Least square methods;
- Matrix/Tensor factorization.

Pros

Incorporate models based on physical mechanisms

Clear interpretation

Cons

Without fully exploring the inherent data priors

Limited performance

Data-driven Models

Powered by historical data, guided by machine learning, e.g.,

- Kernel machine;
- Manifold learning;
- Deep neural networks;

Pros

Rapidly developed deep neural networks in particular

Powerful capability

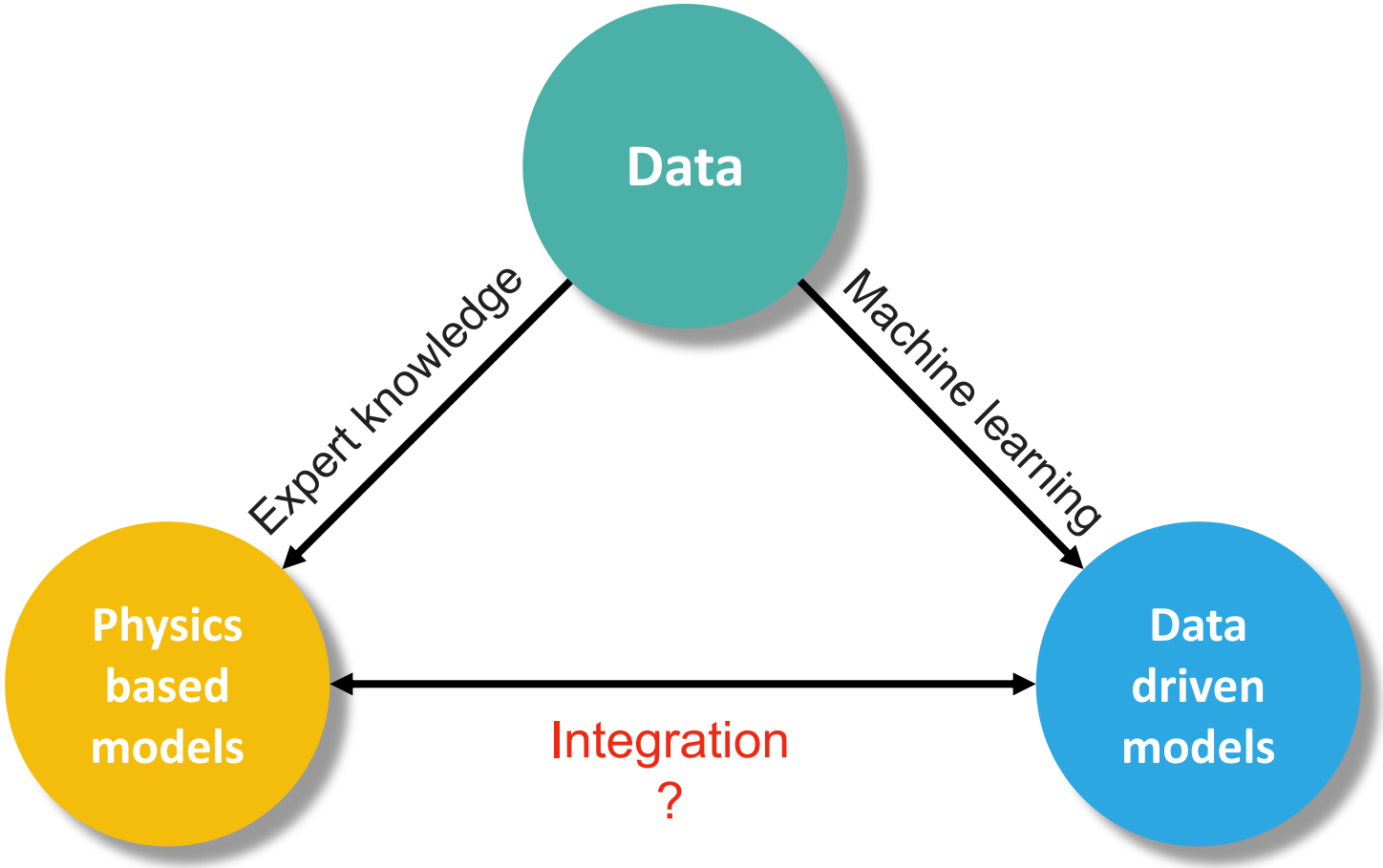
Cons

Directly applying can be black boxes

Less interpretable or general

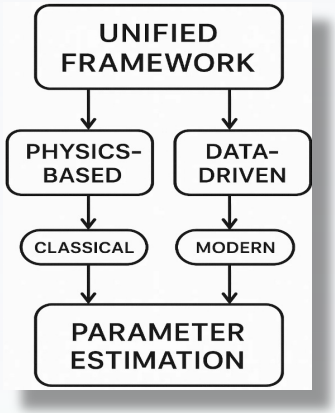
Motivations

Is it possible to leverage the advantages of both physics-based and data-driven models?



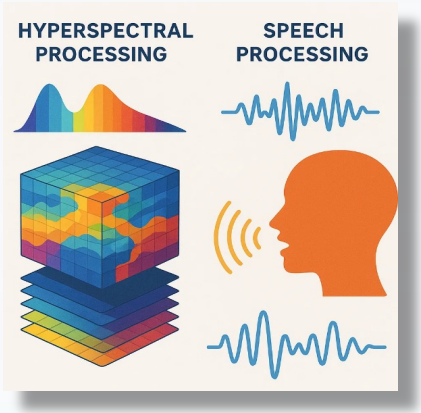
Takeaways from this Tutorial

1. Unified Framework



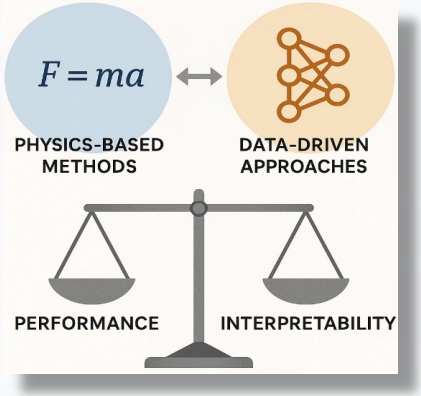
- A unified framework with **hybrid strategy** that accommodates both classical and modern parameter estimation methods

2. Cross-domain Applicability



- The hybrid methodology can be extended to **a wide range** of signal processing applications, such as image processing and speech processing

3. Bridging Interpretability & Performance



- Physics-based methods provide **interpretability**, while data-driven approaches offer **flexibility**.

- **This tutorial is based upon (but not limited to) our papers:**

[1] J. Chen, M. Zhao, X. Wang, C. Richard and S. Rahardja, "Integration of physics-based and data-driven models for hyperspectral image unmixing: A summary of current methods". In: IEEE Signal Processing Magazine 40.2 (2023), pp. 61–74.
[2] Z. Yang, W. Yang, K. Xie and J. Chen, "Integrating Data Priors with Weighted Prediction Error for Speech Dereverberation", IEEE/ACM Transactions on Audio, Speech, Language and Processing, vol. 32, pp. 3908-3923, 2024.

Outline

Outline

General Problem Formulation

A general parameter estimation problem:

$$\text{find } \mathbf{x} \quad s.t. \quad \mathbf{y} = \mathcal{F}(\mathbf{x}) + \mathbf{n}\,.$$

Maximum A Posteriori (MAP) formulation:

$$\begin{aligned}\hat{\mathbf{x}} &= \arg\max_{\mathbf{x}} \{\log p(\mathbf{y}|\mathbf{x}) \log p(\mathbf{x})\} \\ &= \arg\min_{\mathbf{x}} \left\{ \frac{1}{2\sigma^2} \|\mathbf{y} - \mathcal{F}(\mathbf{x})\|^2 + \alpha \mathcal{R}(\mathbf{x}) \right\}.\end{aligned}$$

i.i.d. Gaussian

Regularized optimization:

$$\begin{aligned}\hat{\mathbf{x}} &= \arg\min_{\mathbf{x}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\} \\ &= \arg\min_{\mathbf{x}} \left\{ \|\mathbf{y} - \mathcal{F}(\mathbf{x})\|^2 + \lambda \mathcal{R}(\mathbf{x}) \right\} \qquad \text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}).\end{aligned}$$

This problem can have different problem formulations depending on whether $\mathcal{F}(\cdot)$ is (partially) known or not[1].

[1] J. Chen, M. Zhao, X. Wang, C. Richard and S. Rahardja, “Integration of physics-based and data-driven models for hyperspectral image unmixing: A summary of current methods”. In: IEEE Signal Processing Magazine 40.2 (2023), pp. 61–74.

General Problem Formulation

Problem formulation 1: Assuming $\mathcal{F}(\cdot)$ is known, the formulation is

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\}$$
$$\text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}).$$

Examples:

- Least square methods[1]:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 + \lambda \mathcal{R}(\mathbf{x}) \right\}.$$

- Regularized nonnegative matrix factorization[2]

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \left\{ \|\mathbf{Y} - \mathbf{A}\mathbf{X}\|_{\mathrm{F}}^2 + \lambda \mathcal{R}(\mathbf{A}, \mathbf{X}) \right\}.$$

[1] D. Heinz and C. Chang, “Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery”. In: IEEE Trans. Geosci. Remote Sens. 39.3 (2001), 529–545.
[2] J. Peng, W. Sun, H. Li, W. Li, X. Meng, C. Ge, and Q. Du, “Low-rank and sparse representation for hyperspectral image processing: A review”. In: IEEE Geosci. Remote Sens. Mag. 10.1 (2021), 10–43.

General Problem Formulation

Problem formulation 2: Assuming $\mathcal{F}(\cdot)$ is (partially) unknown, the formulation is

$$\hat{\mathbf{x}} = \underset{\mathbf{x},\mathcal{F}}{\arg\min} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\} \quad \text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}).$$

Examples:

- Kernel-based learning approaches[1]

$$\{\hat{\mathbf{x}}, \hat{\mathcal{F}}\} = \arg \min_{\mathbf{x},\mathcal{F}\in\mathcal{H}_{\text{RKHS}}} \left\{\|\mathbf{y} - \mathcal{F}(\mathbf{x})\|^2 + \lambda \mathcal{R}(\mathcal{F}, \mathbf{x})\right\}.$$

- Manifold learning methods[2]

$$\mathbf{y} = \mathcal{C}(\mathbf{A}\mathbf{x}) + \mathbf{n}\,.$$

[1] J. Chen, C. Richard, and P. Honeine, “Nonlinear estimation of material abundances in hyperspectral images with l1-norm spatial regularization,” IEEE Trans. Geosci. Remote Sens., vol. 52, no. 5, pp. 2654–2665, 2013.
[2] R. Heylen, M. Parente, and P. Gader, “A review of nonlinear hyperspectral unmixing methods,” IEEE J. Sel. Top. Appl. Earth Observat. Remote Sens., vol. 7, no. 6, pp. 1844–1868, 2014.

General Problem Formulation

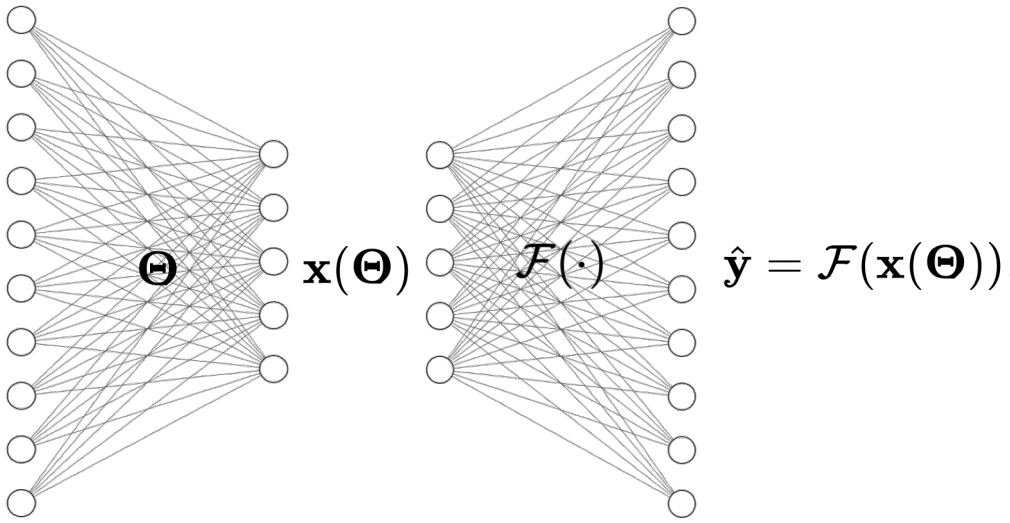
Problem formulation 3: Assuming $\mathcal{F}(\cdot)$ is (partially) unknown, the optimization formulation jointly

- Learn $\mathcal{F}(\cdot)$ from data;
- Considers re-parameterizing $\mathbf{x}$.

$$\{\hat{\mathbf{x}}, \boldsymbol{\Theta}^*\} = \arg\min_{\mathbf{x},\boldsymbol{\Theta},\mathcal{F}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\} \quad \text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}(\boldsymbol{\Theta})).$$

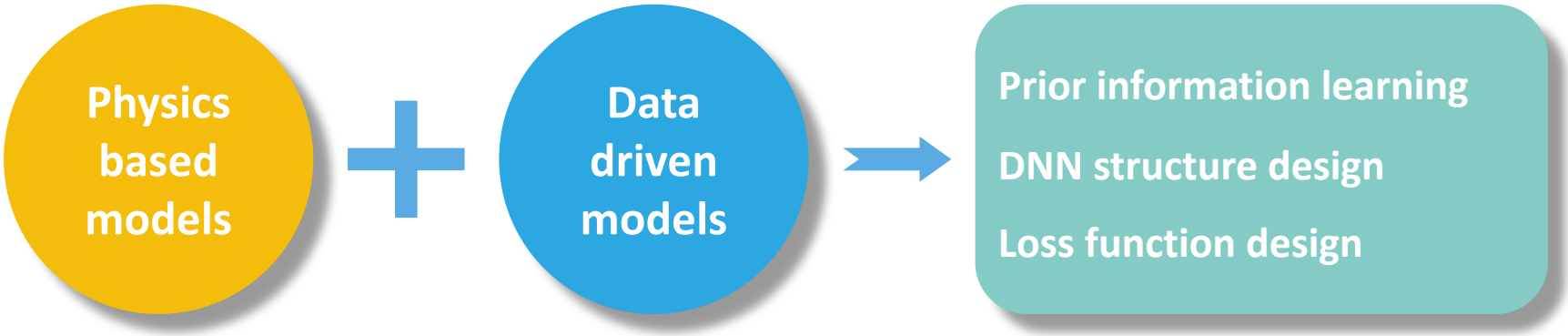
design of $\mathcal{R}$?

Examples:



Hybrid Approaches for Parameter Estimation

Recent physics-based data-driven methods show their power in improving the following three aspects[1][2]:



[1] J. Chen, M. Zhao, X. Wang, C. Richard and S. Rahardja, “Integration of physics-based and data-driven models for hyperspectral image unmixing: A summary of current methods”. In: IEEE Signal Processing Magazine 40.2 (2023), pp. 61–74.
[2] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, “Physics-informed machine learning,” Nature Reviews Physics, vol. 3, no. 6, pp. 422–440, 2021.

Prior Information Learning - I

Problem formulation 1:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\}$$
$$\text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}).$$

Deep prior regularization[1]: plug the output of a DNN into the objective function by defining

$$\mathcal{R}(\mathbf{x}) = d(\mathbf{x}, \tilde{\mathbf{x}})^2 ,$$
$$\text{s.t.} \quad \tilde{\mathbf{x}} = \text{DNN}(\mathbf{y}) .$$

Examples:

- The Frobenius norm[1]: $\mathcal{R}(\mathbf{x}) = \|\mathbf{x} - \tilde{\mathbf{x}}\|^2$;
- The 2D Total Variation (TV) norm[2]: $\mathcal{R}(\mathbf{x}) = \|\mathbf{x} - \tilde{\mathbf{x}}\|_{TV} + \|\mathbf{x}\|_{TV}$.

[1] X. Wang, J. Chen, Q. Wei, and C. Richard, "Hyperspectral image superresolution via deep prior regularization with parameter estimation," IEEE Trans. Circuits Syst. Video Technol., 2021.
[2] M. Vella, B. Zhang, W. Chen, and J. F. C. Mota, "Enhanced hyperspectral image super-resolution via rgb fusion and tv-tv minimization," in Proc. IEEE Int. Conf. Image Process. (ICIP), 2021.

Prior Information Learning - II

Problem formulation 1:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\}$$
$$\text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}).$$

Plug-and-play framework[1]: Plug the denoiser into optimization, without need to explicitly define $\mathcal{R}(\mathbf{x})$.

$$\mathbf{x}^{(k+1)} = \arg\min_{\mathbf{x}} \mathcal{L}\left(\mathcal{F}(\mathbf{x}), \mathbf{y}\right) + \frac{\rho}{2} \left\| \mathbf{x} - \mathbf{z}^{(k)} + \mathbf{v}^{(k)} \right\|_2^2$$
$$\mathbf{z}^{(k+1)} = \frac{\rho}{2} \left\| \mathbf{x} - \mathbf{z}^{(k)} + \mathbf{v}^{(k)} \right\|_2^2 + \lambda \mathcal{R}(\mathbf{x}) = \text{Denoiser}\left(\mathbf{x}^{(k+1)} + \mathbf{v}^{(k)}\right)$$
$$\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \mathbf{x}^{(k+1)} - \mathbf{z}^{(k+1)}$$

Examples:

- Plug-and-play ADMM[1];
- Regularization by Denoising (RED)[2] .

[1] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, “Plug-and-play priors for model-based reconstruction,” in Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP), Austin, TX, USA, pp. 945–948, Dec. 2013.
[2] Y. Romano, M. Elad, and P. Milanfar, “The little engine that could: Regularization by denoising (RED),”SIAM J. Imaging Sci., vol. 10, no. 4, pp. 1804–1844, 2017.

Prior Information Learning - III

Problem formulation 1:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\}$$
$$\text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}).$$

Deep unrolling framework **unfolds iterative optimization** into a trainable **end-to-end deep network**.

The sparse coding problem:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|^2 + \lambda \|\mathbf{x}\|_1 \right\}.$$

Iterative Optimization

$$\mathbf{x}^{(k+1)} = \eta_{\lambda\alpha}\left(\mathbf{x}^{(k)} - \alpha \mathbf{A}^\top (\mathbf{A}\mathbf{x}^{(k)} - \mathbf{y})\right).$$

Unrolled Deep Network

$$\mathbf{x}^{(k+1)} = \eta_{\theta^{(k)}}\left(\mathbf{W}_1 \mathbf{y} + \mathbf{W}_2 \mathbf{x}^{(k)}\right).$$

Examples:

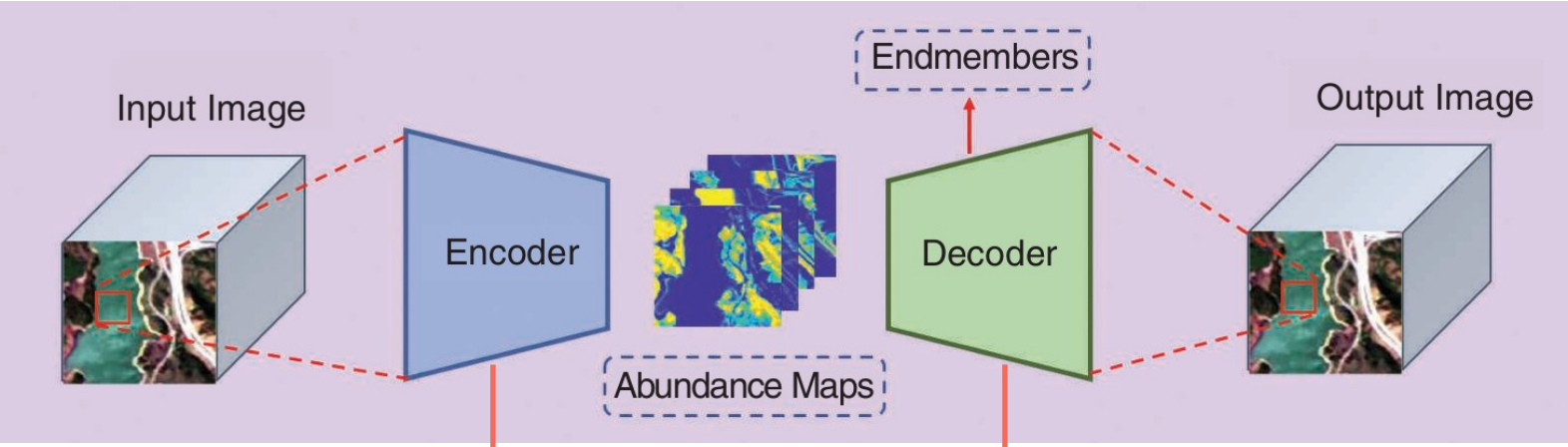
- Learned ISTA[1]
- ADMM-net[2]

[1] K. Gregor and Y. LeCun, “Learning fast approximations of sparse coding,” in Proc. 27th Int. Conf. Mach. Learn. (ICML), 2010, pp. 399–406.
[2] Y. Yang, J. Sun, H. Li, and Z. Xu, “Deep ADMM-Net for compressive sensing MRI,” in Adv. Neural Inf. Process. Syst. (NeurIPS), 2016, pp. 10–18.

DNN structure design

Problem formulation 3:

$$\{\hat{\mathbf{x}}, \boldsymbol{\Theta}^*\} = \arg\min_{\mathbf{x},\boldsymbol{\Theta},\mathcal{F}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\}$$
$$\text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}(\boldsymbol{\Theta})).$$



Encoder Design $\mathcal{Q} = f_{\Theta_{\text{enc}}}(y)$

- **FCN / CNN / Multistream encoders**
- **Kernel-based encoders** (Riemannian metric, spectral correlation)
- **Variational encoders (VAE)** for probabilistic modeling
- **Multibranch / multitask encoders** to capture spectral-spatial priors

$\hat{y} = f_{\Theta_{\text{dec}}}(\mathcal{Q})$ Decoder Design

- **Linear decoder** $\hat{y} = W\Theta$
- **Postnonlinear decoder** $\hat{y} = \Psi(W\Theta)$
- **Additive nonlinear decoder** $\hat{y} = W\Theta + \Psi(W\Theta)$
- **Bilinear decoder** $\hat{y} = W\Theta + Db$
- **Endmember-generating decoder**

$$\hat{y} = f_{\Theta_{\text{end}}}(z) = \Psi(z)$$

Loss function design

Problem formulation 3:

$$\{\hat{\mathbf{x}}, \boldsymbol{\Theta}^*\} = \arg\min_{\mathbf{x},\boldsymbol{\Theta},\mathcal{F}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\}$$
$$\text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}(\boldsymbol{\Theta})).$$

Physically inspired cost functions have been proposed recently to enhance performance.

Examples:

1. General geometric distances:

- Mean-squared error (MSE)[1] $\mathcal{L}_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N} \|y_i - \hat{y}_i\|^2$ ⟶ Sensitive to the scale of spectra
- Spectral angle distance (SAD) $\mathcal{L}_{\text{SAD}} = \frac{1}{N}\sum_{i=1}^{N} \arccos\left(\frac{\langle y_i, \hat{y}_i \rangle}{\|y_i\| \, \|\hat{y}_i\|}\right)$

  Spectral information divergence (SID) $\mathcal{L}_{\text{SID}} = \frac{1}{N}\sum_{i=1}^{N} p_i \log\left(\frac{p_i}{\hat{p}_i}\right)$

  } Scale invariant

[1] S. Ozkan, B. Kaya, and G. B. Akar, “EndNet: Sparse autoencoder network for endmember extraction and hyperspectral unmixing,” IEEE Trans. Geosci. Remote Sens., vol. 57, no. 1, pp. 482–496, Jan. 2019.

Loss function design

Problem formulation 3:

$$\{\hat{\mathbf{x}}, \boldsymbol{\Theta}^*\} = \arg\min_{\mathbf{x},\boldsymbol{\Theta},\mathcal{F}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\} \quad \text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}(\boldsymbol{\Theta})).$$

Physically inspired cost functions have been proposed recently to enhance performance.

Examples:

2. Deep learning metrics:

- Generative adversarial networks (GANs)[1] $\mathcal{L}_{\mathcal{G}} = \mathcal{L}_{\text{AE}} + \mathcal{L}_{\text{adv}}$

Discriminator: distinguish between the real and generated samples

Generator: Generate samples that are similar to real data

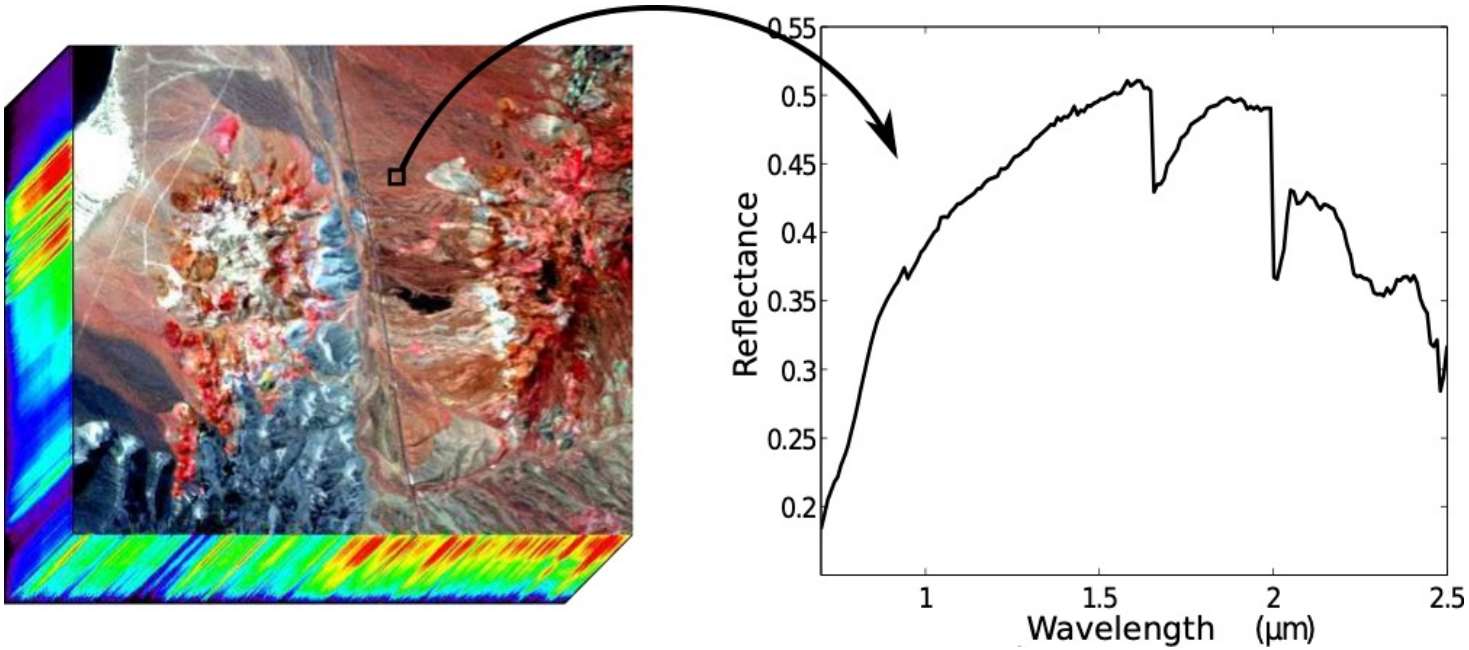
- The perceptual loss[2] $\mathcal{L}_{\text{Perceptual}} = \frac{1}{N}\sum_{i=1}^{N} \|\mathcal{P}(y_i) - \mathcal{P}(\hat{y}_i)\|^2$

a feature extractor

[1] Q. Jin, Y. Ma, F. Fan, J. Huang, X. Mei, and J. Ma, “Adversarial autoencoder network for hyperspectral unmixing,” IEEE Trans. Neural Netw. Learn. Syst., early access, 2022.
[2] L. Gao, Z. Han, D. Hong, B. Zhang, and J. Chanussot, “CyCU-Net: Cycle consistency unmixing network by learning cascaded autoencoders,” IEEE Trans. Geosci. Remote Sens., vol. 60, 2022.

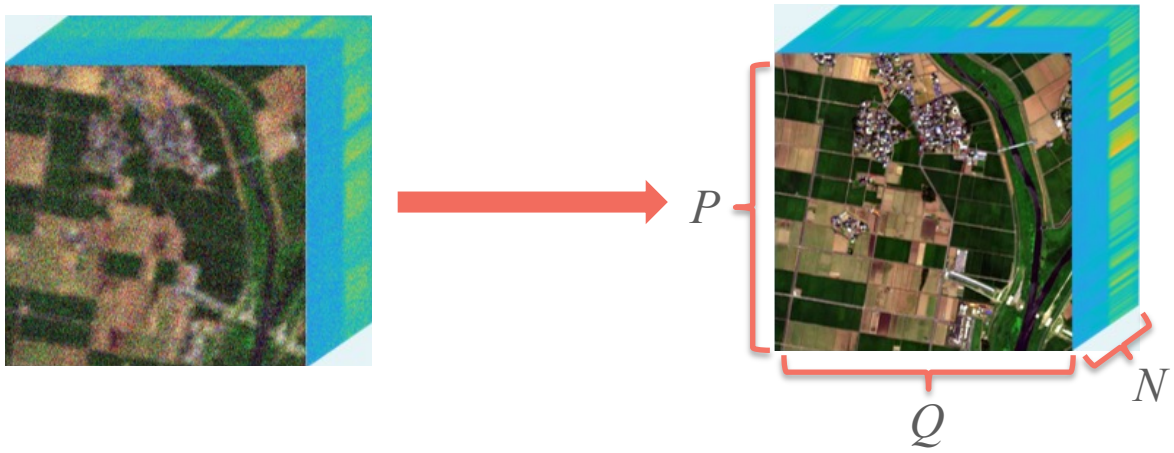
Outline

Background in Hyperpsectral Imaging



- High spectral resolution (hundreds of spectral bands);
- Core part of many applications, e.g., remote sensing, medical imaging, etc.

Background in Hyperpsectral Imaging



An illustration of hyperspectral deconvolution

The abundant spectral information makes hyperspectral image processing more complex than ordinary 2D images:

- Spatial prior;
- Spectral prior.

Hybrid Strategies for Hyperspectral Deconvolution

Problem formulation 1:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\} \quad \text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}).$$

Linear degradation model:

$$\mathbf{Y}_i = \mathcal{H}_i * \mathbf{X}_i + \mathbf{N}_i\,, \quad \forall i.$$

This model can be written as[1]:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}\,,$$

The optimization problem:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \lambda \mathcal{R}(\mathbf{x}) \right\}.$$

The prior information of $\mathbf{x}$ is encoded in $\mathcal{R}(\mathbf{x})$.

[1] X. Wang, J. Chen, and C. Richard, “Tuning-free plug-and-play hyperspectral image deconvolution with deep priors,” IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1–13, 2023.

Hybrid Strategies for Hyperspectral Deconvolution

Constrained optimization:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \lambda\mathcal{R}(\mathbf{z}), \quad \text{s.t.} \quad \mathbf{z} = \mathbf{x}.$$

Augmented Lagrangian :

$$\mathcal{L}_\rho(\mathbf{x}, \mathbf{z}, \mathbf{v}) = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \lambda\Phi(\mathbf{z}) + \mathbf{v}^T(\mathbf{x} - \mathbf{z}) + \frac{\rho}{2}\|\mathbf{x} - \mathbf{z}\|^2.$$

Variable splitting based on ADMM:

$$\begin{aligned}
\mathbf{x}_{k+1} &= \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \frac{\rho_k}{2}\|\mathbf{x} - \tilde{\mathbf{x}}_k\|^2 \\
\mathbf{z}_{k+1} &= \arg\min_{\mathbf{z}} \lambda\Phi(\mathbf{z}) + \frac{\rho_k}{2}\|\tilde{\mathbf{z}}_k - \mathbf{z}\|^2 \\
\mathbf{u}_{k+1} &= \mathbf{u}_k + \mathbf{x}_{k+1} - \mathbf{z}_{k+1}
\end{aligned}$$

where

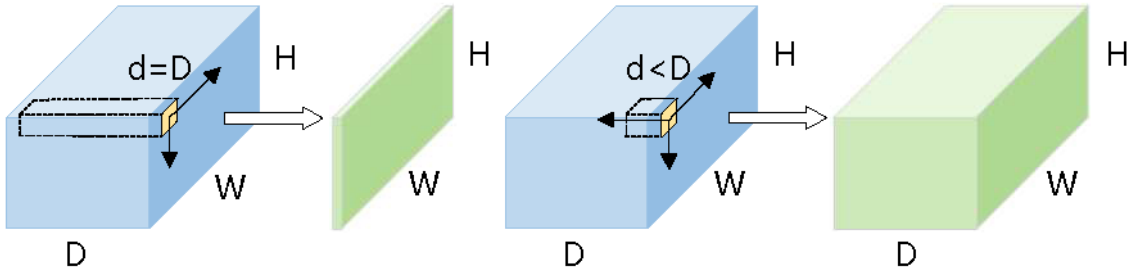
$$\begin{aligned}
\tilde{\mathbf{x}}_k &= \mathbf{z}_k - \mathbf{u}_k \\
\tilde{\mathbf{z}}_k &= \mathbf{x}_{k+1} + \mathbf{u}_k
\end{aligned}$$

Deep denoiser

[1] X. Wang, J. Chen, and C. Richard, “Tuning-free plug-and-play hyperspectral image deconvolution with deep priors,” IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1–13, 2023.

Hybrid Strategies for Hyperspectral Deconvolution

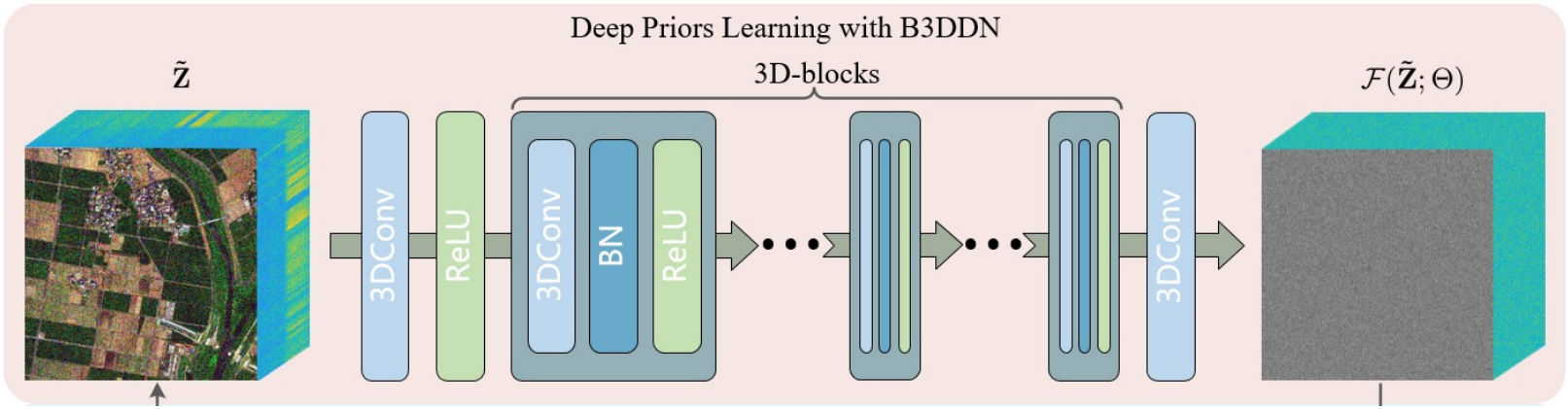
Deep denoiser:



Difference between 2D and 3D convolution

Two advantages:

- Operate locally in Spectral domain
- Involve fewer kernel parameters



Architecture of the proposed B3DDN for hyperspectral image denoising

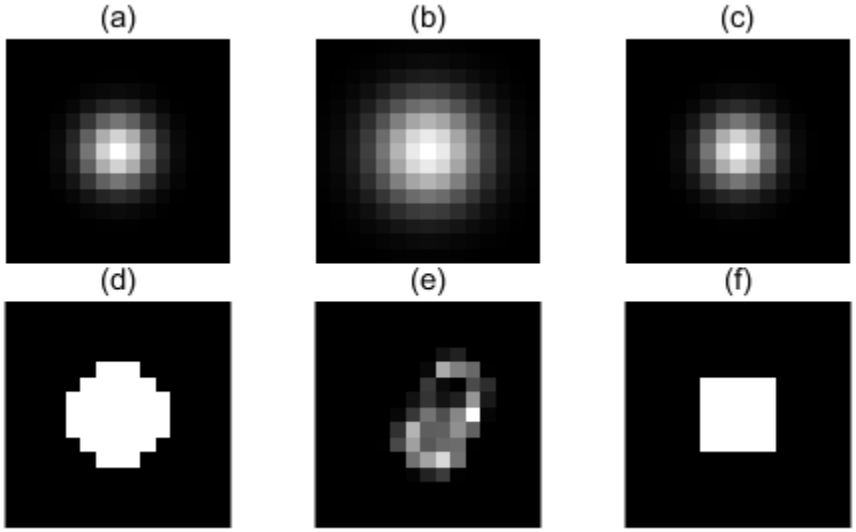
Loss function:

$$\ell(\Theta) = \|\mathcal{F}(\tilde{\mathbf{z}}_m;\Theta) - (\tilde{\mathbf{z}}_m - \mathbf{z}_m)\|_1$$

[1] X. Wang, J. Chen, and C. Richard, “Tuning-free plug-and-play hyperspectral image deconvolution with deep priors,” IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1–13, 2023.

Hybrid Strategies for Hyperspectral Deconvolution

Experiments:



Blurring kernels used in the experiments

Baselines:

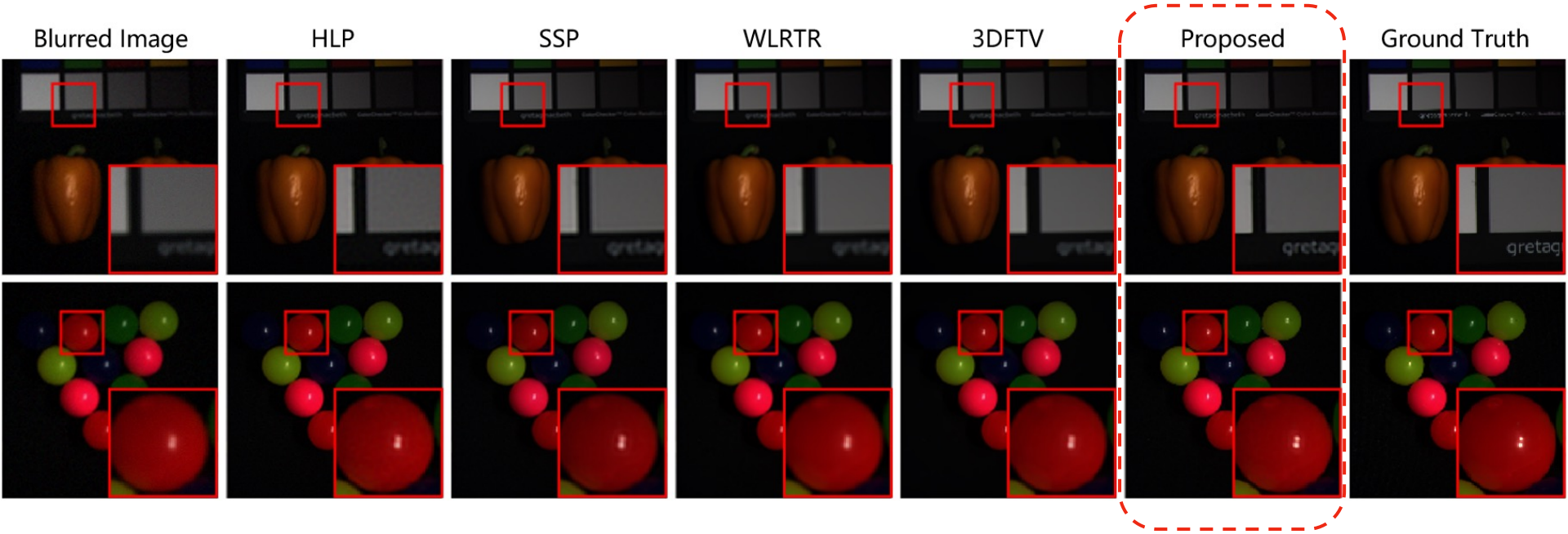
- HLP: considers the spatial smoothness;
- SSP: exploites both the spatial and spectral smoothness;
- WLRTR: captures non-local similarity and spectral correlation;
- 3DFTV: exploits smoothness of images in all dimensions.

Hybrid Strategies for Hyperspectral Deconvolution

Scenarios	Metrics	HLP	SSP	WLRTR	3DFTV	Ours
(a)	RMSE	4.420 ± 1.787	4.848 ± 1.825	4.735 ± 2.076	4.332 ± 1.863	**3.132** ± **1.320**
	PSNR	36.166 ± **3.334**	35.373 ± 3.385	35.872 ± 3.759	36.450 ± 3.793	**39.252** ± 3.465
	SSIM	0.9167 ± 0.0379	0.9305 ± 0.0393	0.9380 ± 0.0466	0.9401 ± 0.0439	**0.9493** ± **0.0367**
	ERGAS	18.15 ± 8.25	19.51 ± 8.12	18.96 ± 8.33	17.34 ± 7.69	**13.01** ± **6.23**
(b)	RMSE	5.707 ± 2.452	5.955 ± 2.398	6.439 ± 2.812	5.667 ± 2.539	**4.581** ± **1.993**
	PSNR	34.034 ± 3.567	33.541 ± **3.492**	33.084 ± 3.740	34.116 ± 3.872	**36.305** ± 3.612
	SSIM	0.8911 ± 0.0483	0.9031 ± 0.0494	0.9025 ± 0.0616	0.9136 ± 0.550	**0.9234** ± **0.0422**
	ERGAS	22.92 ± 10.11	23.71 ± 9.86	25.46 ± 10.84	22.40 ± 9.86	**18.54** ± **8.39**
(c)	RMSE	7.669 ± 1.390	5.270 ± 1.622	5.099 ± 1.972	5.016 ± 1.727	**4.225** ± **1.324**
	PSNR	30.599 ± **1.550**	34.309 ± 2.607	34.827 ± 3.201	34.741 ± 2.975	**36.211** ± 2.485
	SSIM	0.6406 ± 0.0337	0.8565 ± 0.0539	**0.8956** ± **0.0387**	0.8851 ± 0.0390	0.8708 ± 0.0594
	ERGAS	33.49 ± 16.27	22.28 ± 10.14	20.80 ± **9.07**	20.47 ± 8.66	**18.64** ± 9.28
(d)	RMSE	4.189 ± 1.636	4.584 ± 1.680	4.328 ± 1.903	4.167 ± 1.803	**2.305** ± **0.938**
	PSNR	36.548 ± 3.181	35.862 ± 3.331	36.686 ± 3.736	36.805 ± 3.803	**41.653** ± **3.074**
	SSIM	0.9165 ± 0.0348	0.9354 ± 0.0374	0.9450 ± 0.0436	0.9403 ± 0.0430	**0.9542** ± **0.0340**
	ERGAS	17.36 ± 7.98	18.49 ± 7.67	17.45 ± 7.82	16.69 ± 7.46	**9.86** ± **5.17**
(e)	RMSE	3.759 ± 1.166	3.954 ± **1.333**	4.335 ± 1.780	3.587 ± 1.443	**3.041** ± 2.783
	PSNR	37.149 ± **2.492**	37.160 ± 3.108	36.497 ± 3.490	37.991 ± 3.543	**40.722** ± 5.730
	SSIM	0.9118 ± **0.0239**	**0.9472** ± 0.0311	0.9428 ± 0.0436	0.9510 ± 0.0397	0.8907 ± 0.1642
	ERGAS	15.94 ± 7.39	16.01 ± **6.56**	17.46 ± 7.49	14.37 ± 6.16	**15.56** ± 19.66
(f)	RMSE	3.971 ± 1.453	4.356 ± 1.563	4.109 ± 1.765	3.957 ± 1.666	**2.280** ± **1.231**
	PSNR	36.910 ± **2.985**	36.322 ± 3.302	37.130 ± 3.698	37.225 ± 3.743	**41.932** ± 3.687
	SSIM	0.9195 ± 0.0270	0.9397 ± 0.334	0.9480 ± 0.0450	0.9468 ± 0410	**0.9475** ± **0.0618**
	ERGAS	16.58 ± 7.61	17.60 ± 7.26	16.64 ± 7.46	15.89 ± 7.04	**9.79** ± **5.89**

Quantitative comparison

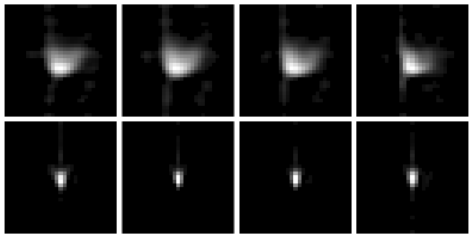
Hybrid Strategies for Hyperspectral Deconvolution



Visual comparison on synthetic data

Hybrid Strategies for Hyperspectral Deconvolution

Real data collection:



Visual comparison on real data[1]

[1] Avaible at: https://github.com/xiuheng-wang/Tuning_free_PnP_HSI_deconvolution

Hybrid Strategies for Hyperspectral Deconvolution

Constrained optimization:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \lambda\mathcal{R}(\mathbf{z}), \quad \text{s.t.} \quad \mathbf{z} = \mathbf{x}.$$

Augmented Lagrangian：

$$\mathcal{L}_\rho(\mathbf{x}, \mathbf{z}, \mathbf{v}) = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \lambda\Phi(\mathbf{z}) + \frac{\rho}{2}\|\mathbf{x} - \mathbf{z}\|^2.$$

Variable splitting based on HQS[1]:

$$\mathbf{z}_{k+1} = \arg\min_{\mathbf{z}} \lambda\Phi(\mathbf{z}) + \frac{\rho}{2}\|\mathbf{x}_k - \mathbf{z}\|^2$$

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \frac{\rho}{2}\|\mathbf{x} - \mathbf{z}_{k+1}\|^2$$

$$\mathbf{z}_{k+1} = \text{denoiser}(\mathbf{x}_k, \Theta_k)$$

$$\mathbf{x}_{k+1} = (\mathbf{H}^T\mathbf{H} + \rho\mathbf{I})^{-1}(\mathbf{H}^T\mathbf{y} + \rho\mathbf{z}_{k+1})$$

$$\mathbf{x}_{k+1} = \text{block}(\mathbf{x}_k, \Theta_k)$$

[1] A. Gkillas, D. Ampeliotis, and K. Berberidis, “A highly interpretable deep equilibrium network for hyperspectral image deconvolution,” in ICASSP, 2023, pp. 1–5.

Hybrid Strategies for Hyperspectral Deconvolution

Deep unrolling framework[1]:

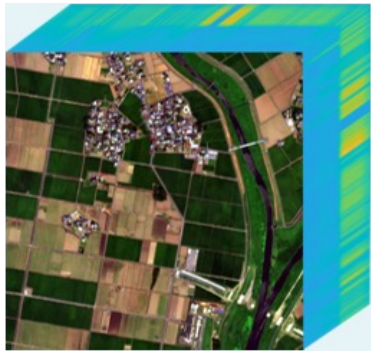
Input image

$$\mathbf{x}_1 = \text{block}(\mathbf{x}_0, \Theta_0)$$

$$\mathbf{x}_{k+1} = \text{block}(\mathbf{x}_k, \Theta_k)$$

$$\mathbf{x}_K = \text{block}(\mathbf{x}_{K-1}, \Theta_{K-1})$$

Output image



[1] A. Gkillas, D. Ampeliotis, and K. Berberidis, “A highly interpretable deep equilibrium network for hyperspectral image deconvolution,” in ICASSP, 2023, pp. 1–5.

Hybrid Strategies for Hyperspectral Deconvolution

Experimental result[2]:

		Plug-and-play[1]	deep unrolling[2]
(a)	PSNR	41.17	**41.78**
	SSIM	0.9602	**0.9793**
	SAM	6.27	**5.94**
(b)	PSNR	35.96	**38.51**
	SSIM	0.9199	**0.9647**
	SAM	7.77	**6.68**
(c)	PSNR	37.42	**39.65**
	SSIM	0.9395	**0.9678**
	SAM	6.46	**6.13**
(d)	PSNR	39.88	**41.77**
	SSIM	0.9477	**0.9801**
	SAM	6.80	**6.01**
(e)	PSNR	42.71	**45.21**
	SSIM	0.9669	**0.9884**
	SAM	6.29	**5.55**

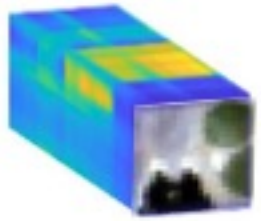
[1] X. Wang, J. Chen, C. Richard, and D. Brie, “Learning spectral-spatial prior via 3ddncnn for hyperspectral image deconvolution,” in ICASSP, 2020, pp. 2403–2407.
[2] A. Gkillas, D. Ampeliotis, and K. Berberidis, “A highly interpretable deep equilibrium network for hyperspectral image deconvolution,” in ICASSP, 2023, pp. 1–5.

Hybrid Strategies for Hyperspectral Deconvolution

Plug-and-play *v.s.* Deep unrolling:

	Plug-and-play	Deep unrolling
Flexibility	Very flexible, reusable denoiser	Less flexible, needs retraining
Task Specificity	Generic, not task-optimized	Task-specific, highly adapted
Data	No paired data needed	Requires paired training data
Inference Speed	Slow, many iterations	Fast, fixed number of layers
Performance	Not always optimal	Often state-of-the-art

Hybrid Strategies for Hyperspectral Fusion



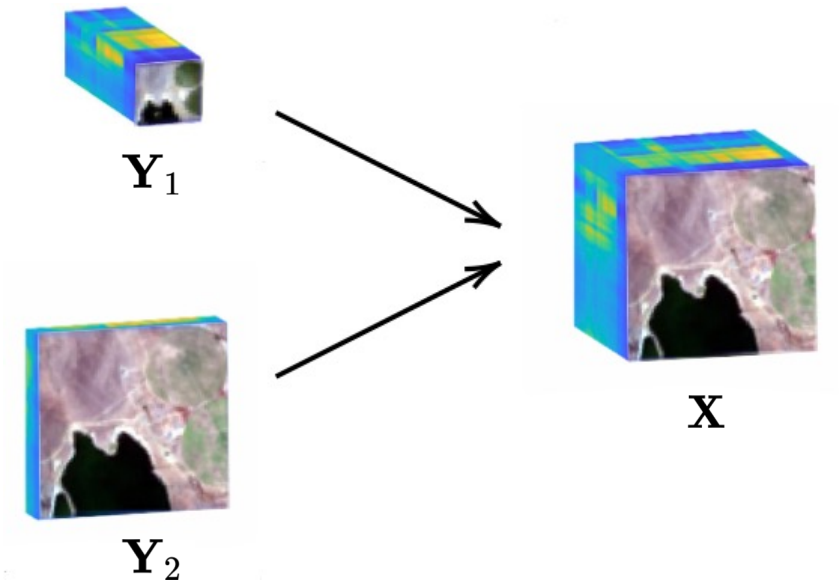
Hyperspectral Image (HI):

- Hundreds of spectral bands
- Low spatial resolution

Multispectral Image (MI):

- Very few spectral bands
- High spatial resolution

Hybrid Strategies for Hyperspectral Fusion



Linear degradation model:

$$\mathbf{Y}_1 = \mathbf{XBS} + \mathbf{N}_1\,,$$
$$\mathbf{Y}_2 = \mathbf{RX} + \mathbf{N}_2\,.$$

The optimization problem:

$$\widehat{\mathbf{X}} = \arg\min_{\mathbf{X}} \left\{ \|\mathbf{Y}_1 - \mathbf{XBS}\|_{\mathrm{F}}^2 + \|\mathbf{Y}_2 - \mathbf{RX}\|_{\mathrm{F}}^2 + \lambda \mathcal{R}(\mathbf{X}) \right\}.$$

Hybrid Strategies for Hyperspectral Fusion

Deep Prior Regularization[1] :

$$\begin{aligned}\widehat{\mathbf{X}} &= \arg\min_{\mathbf{X}} \left\{ \|\mathbf{Y}_1 - \mathbf{XBS}\|_{\mathrm{F}}^2 + \|\mathbf{Y}_2 - \mathbf{RX}\|_{\mathrm{F}}^2 + \lambda \mathcal{R}(\mathbf{X}) \right\} \\ &= \arg\min_{\mathbf{X}} \left\{ \|\mathbf{Y}_1 - \mathbf{XBS}\|_{\mathrm{F}}^2 + \|\mathbf{Y}_2 - \mathbf{RX}\|_{\mathrm{F}}^2 + \lambda \|\mathbf{X} - \widetilde{\mathbf{X}}\|_{\mathrm{F}}^2 \right\}\end{aligned}$$

This formulation allows a straight-forward solver[2] with the optimum defined by

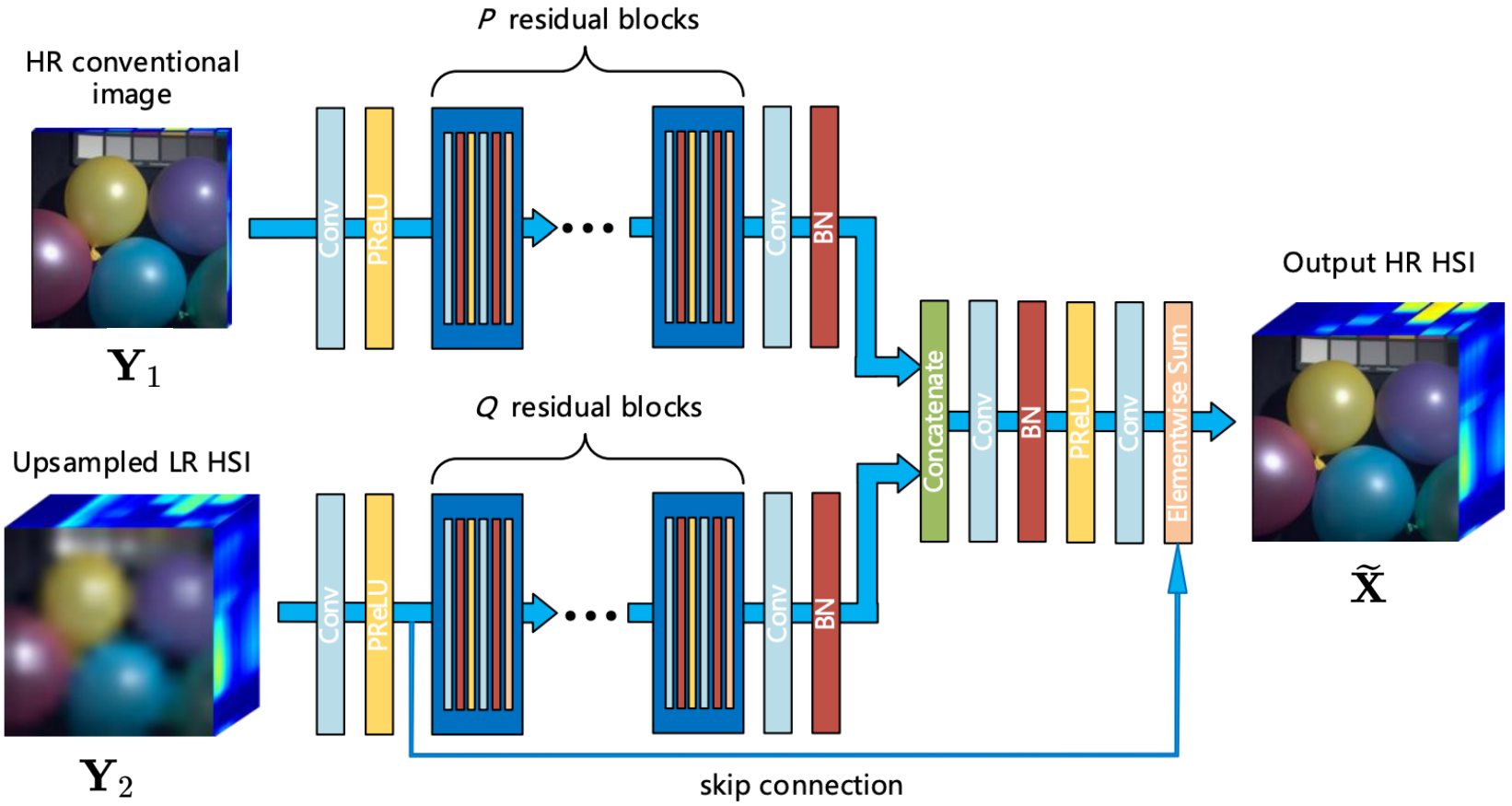
$$\mathbf{C}_1\mathbf{X}^* + \mathbf{X}^*\mathbf{C}_2 = \mathbf{C}_3\,,$$

where

$$\begin{aligned}\mathbf{C_1} &= \mathbf{R}^T\mathbf{R} + \lambda\mathbf{I}_B \\ \mathbf{C_2} &= (\mathbf{BS})(\mathbf{BS})^T \\ \mathbf{C_3} &= \mathbf{R}^T\mathbf{Z} + \mathbf{Y}(\mathbf{BS})^T + \lambda\widetilde{\mathbf{X}}\end{aligned}$$

[1] X. Wang, J. Chen, Q. Wei, and C. Richard, “Hyperspectral image superresolution via deep prior regularization with parameter estimation,” IEEE Trans. Circuits Syst. Video Technol., 2021.
[2] Q. Wei, N. Dobigeon, and J.-Y. Tourneret, “Fast fusion of multi-band images based on solving a sylvester equation,” IEEE Trans. Image Process., vol. 24, no. 11, pp. 4109–4121, 2015.

Hybrid Strategies for Hyperspectral Fusion



Architecture of the proposed TSFN for hyperspectral image fusion.

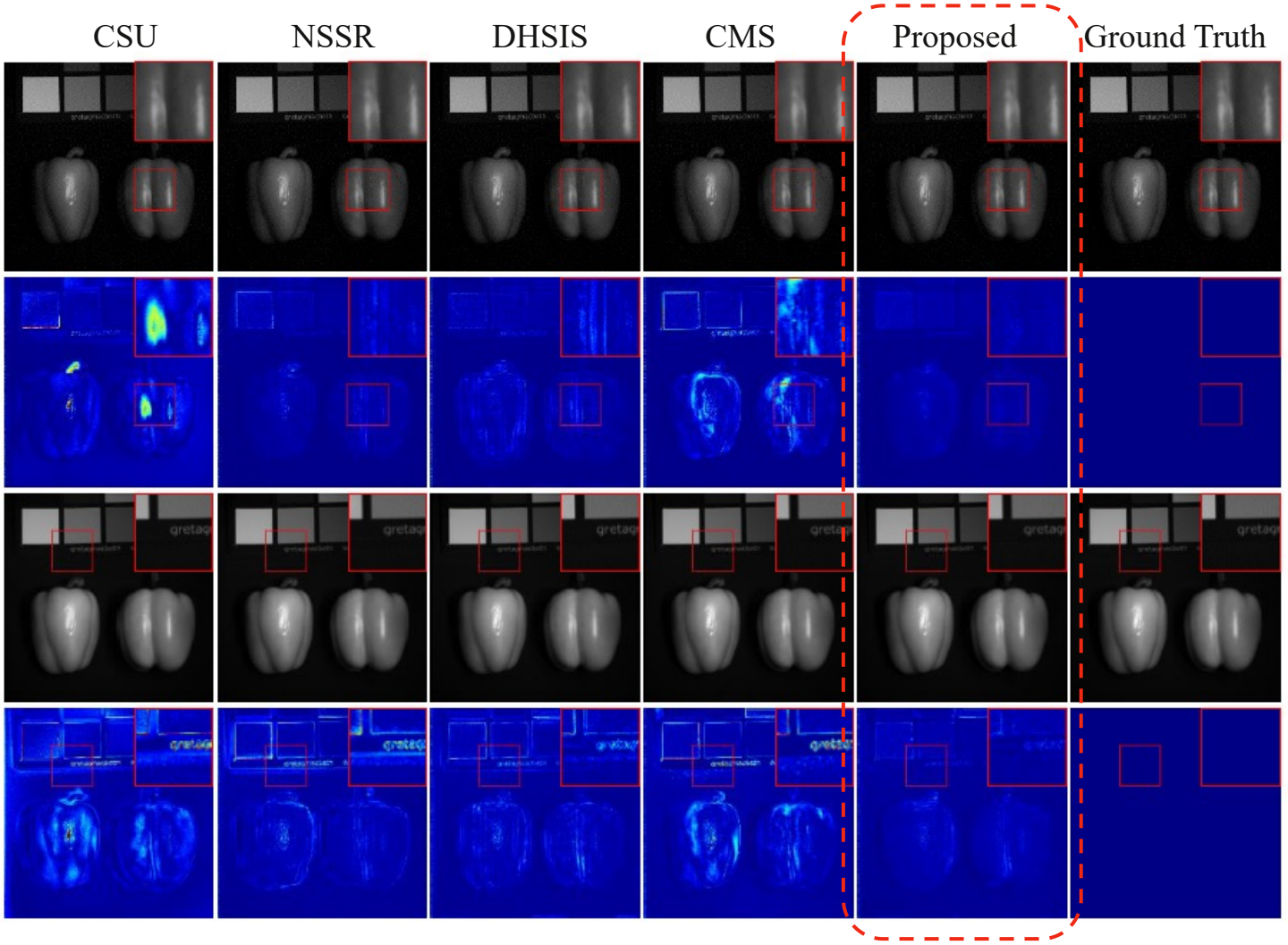
[1] X. Wang, J. Chen, Q. Wei, and C. Richard, “Hyperspectral image superresolution via deep prior regularization with parameter estimation,” IEEE Trans. Circuits Syst. Video Technol., 2021.

Hybrid Strategies for Hyperspectral Fusion

	Metric	CSU	NSSR	DHSIS	CMS	LTTR	UAL	Proposed
$s = 8$	RMSE	2.57	1.47	1.36	1.65	1.49	1.42	**1.12**
	PSNR	41.76	46.66	47.01	45.29	46.55	46.47	**48.63**
	ERGAS	1.196	0.665	0.621	0.734	0.672	0.640	**0.511**
	SAM	6.27	3.72	3.74	3.89	3.84	3.60	**3.23**
$s = 16$	RMSE	2.82	1.77	1.79	1.99	1.84	1.57	**1.39**
	PSNR	41.01	45.31	44.74	43.94	44.87	45.80	**47.02**
	ERGAS	0.643	0.398	0.391	0.445	0.415	0.345	**0.310**
	SAM	6.47	4.32	4.57	4.37	4.63	3.84	**3.76**
$s = 32$	RMSE	3.02	2.24	2.45	2.35	2.28	1.85	**1.80**
	PSNR	40.44	43.49	42.34	42.66	43.33	44.66	**45.07**
	ERGAS	0.336	0.244	0.257	0.257	0.248	0.196	**0.190**
	SAM	6.83	5.22	5.87	5.04	5.46	**4.33**	4.59

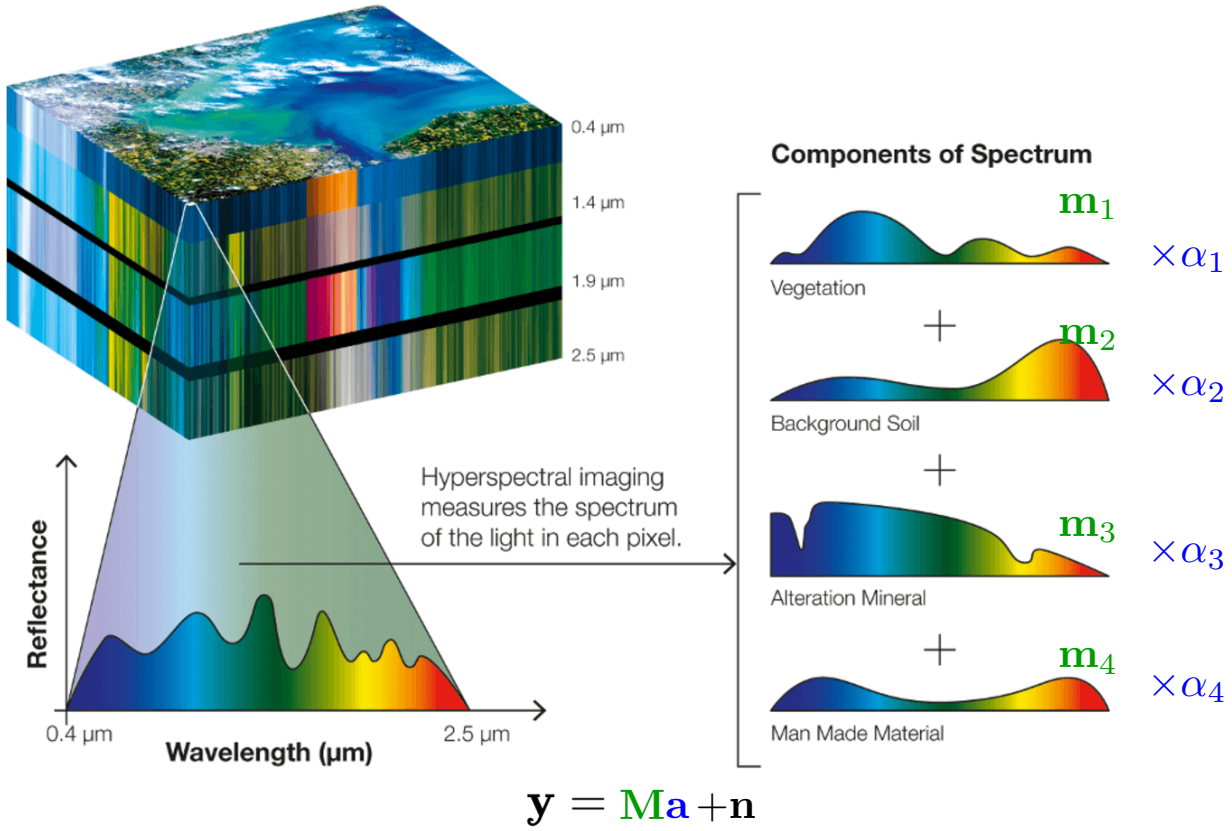
Quantitative comparison

Hybrid Strategies for Hyperspectral Fusion



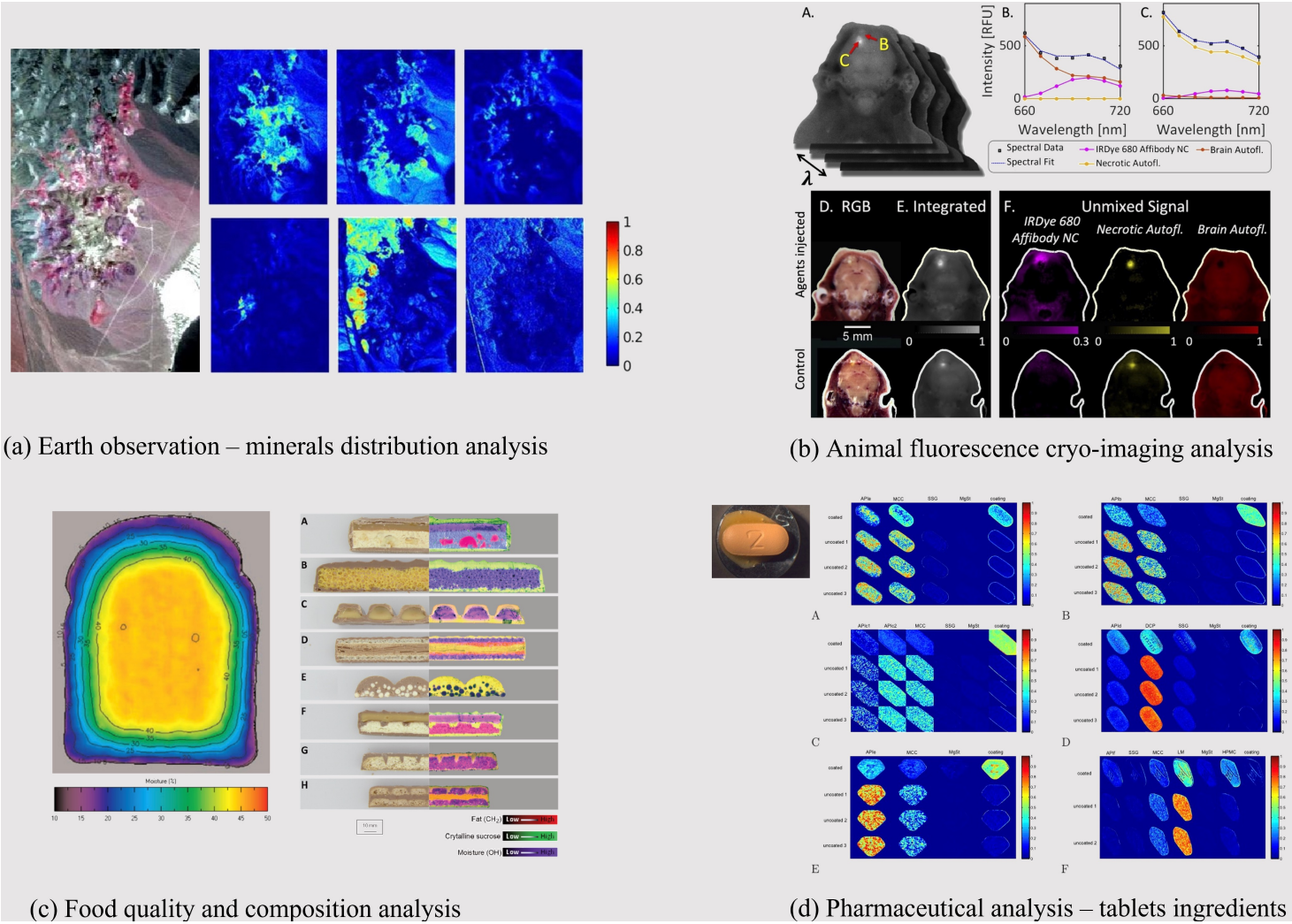
Visual comparison

Hybrid Strategies for Hyperspectral Unmixing



Hyperspectral unmixing aims at separating a mixed pixel into a set of endmembers and their corresponding abundances.

Hybrid Strategies for Hyperspectral Unmixing



(a) Earth observation – minerals distribution analysis

(b) Animal fluorescence cryo-imaging analysis

(c) Food quality and composition analysis

(d) Pharmaceutical analysis – tablets ingredients

Applications of hyperspectral unmixing

Hybrid Strategies for Hyperspectral Unmixing

An observed spectrum can be described by

$$\mathbf{y} = \mathcal{F}^{\star}(\mathbf{M}, \mathbf{a}) + \mathbf{n}$$

The constraints ANC and ASC are often considered:

$$\Omega_M = \{\mathbf{M} : \mathbf{M} \geq \mathbf{0}\}$$
$$\Omega_a = \{\mathbf{a} : \mathbf{a} \geq \mathbf{0};\ a_1 + a_2 + \cdots + a_R = 1\}$$

Examples of physical mixture models[1]

- Linear mixture model: $\mathbf{y} = \mathbf{M}\mathbf{a} + \mathbf{n}$
- Bilinear model: $\mathbf{y} = \mathbf{M}\mathbf{a} + \sum_{i=1}^{R}\sum_{j=1}^{R} \gamma_{i,j}\mathbf{m}_i \odot \mathbf{m}_j + \mathbf{n}$
- Additive nonlinear models: $\mathbf{y} = \mathbf{M}\mathbf{a} + \mathcal{F}_{\text{add}}(\mathbf{M}, \mathbf{a}) + \mathbf{n}$
- Post-nonlinear models: $\mathbf{y} = \mathcal{F}_{\text{post}}(\mathbf{M}, \mathbf{a}) + \mathbf{n}$

[1] J. Chen, M. Zhao, X. Wang, C. Richard and S. Rahardja, “Integration of physics-based and data-driven models for hyperspectral image unmixing: A summary of current methods”. In: IEEE Signal Processing Magazine 40.2 (2023), pp. 61–74.

Hybrid Strategies for Hyperspectral Unmixing

Two examples of linear unmixing:

- The Fully Constrained Least Square (FCLS) problem[1]:

$$\hat{\boldsymbol{a}} = \underset{\boldsymbol{a}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{M}\boldsymbol{a}\|^2$$

$$\text{s.t.} \quad \boldsymbol{a} \in \Omega_\alpha.$$

- The regularized NMF unmixing[2]:

$$\left\{\hat{\boldsymbol{M}}, \{\hat{\boldsymbol{a}}_i\}_{i=1}^N\right\} = \underset{\boldsymbol{M}, \{\boldsymbol{a}_i\}_{i=1}^N}{\operatorname{argmin}} \|\boldsymbol{Y} - \boldsymbol{M}\boldsymbol{A}\|_F^2 + \mathcal{R}\left(\boldsymbol{M}, \{\boldsymbol{a}_i\}_{i=1}^N\right)$$

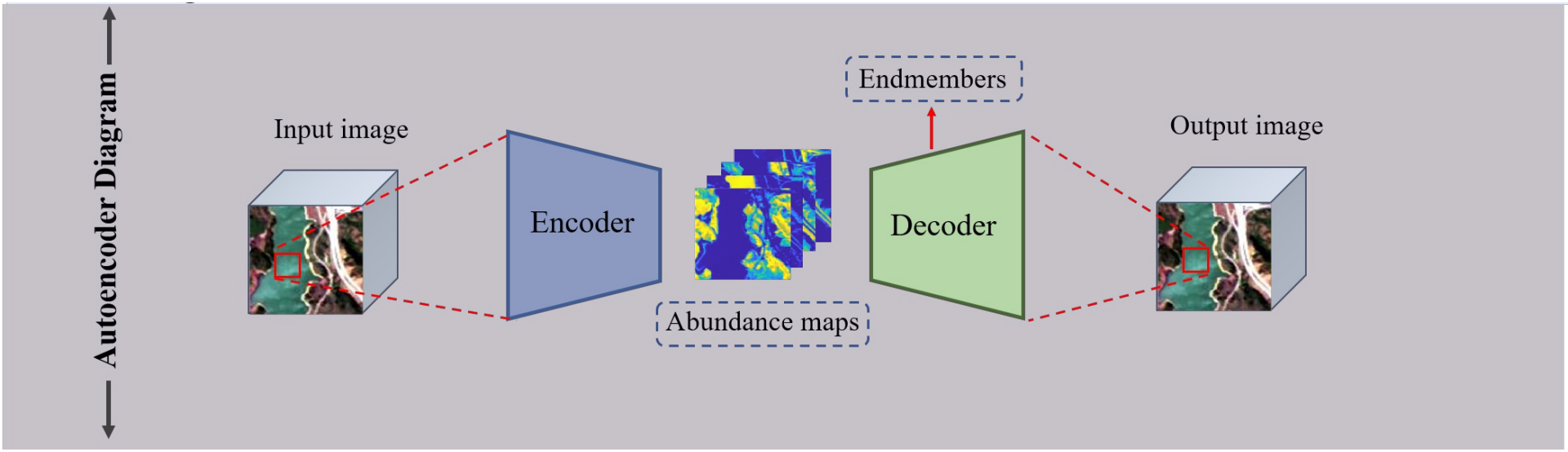
$$\text{s.t.} \quad \boldsymbol{M} \in \Omega_M, \text{ and } \boldsymbol{a}_i \in \Omega_a,$$

[1] D. Heinz and C. Chang, “Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery,” IEEE Trans. Geosci. Remote Sens., vol. 39, no. 3, pp. 529–545, 2001.
[2] J. Peng, W. Sun, H. Li, W. Li, X. Meng, C. Ge, and Q. Du, “Low-rank and sparse representation for hyperspectral image processing: A review,” IEEE Geosci. Remote Sens. Mag., vol. 10, no. 1, pp. 10–43, 2021.

Hybrid Strategies for Hyperspectral Unmixing

$$\{\hat{\mathbf{x}}, \boldsymbol{\Theta}^*\} = \arg\min_{\mathbf{x},\boldsymbol{\Theta},\mathcal{F}} \{\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) + \mathcal{R}(\mathbf{x})\}$$
$$\text{with} \quad \hat{\mathbf{y}} = \mathcal{F}(\mathbf{x}(\boldsymbol{\Theta})).$$

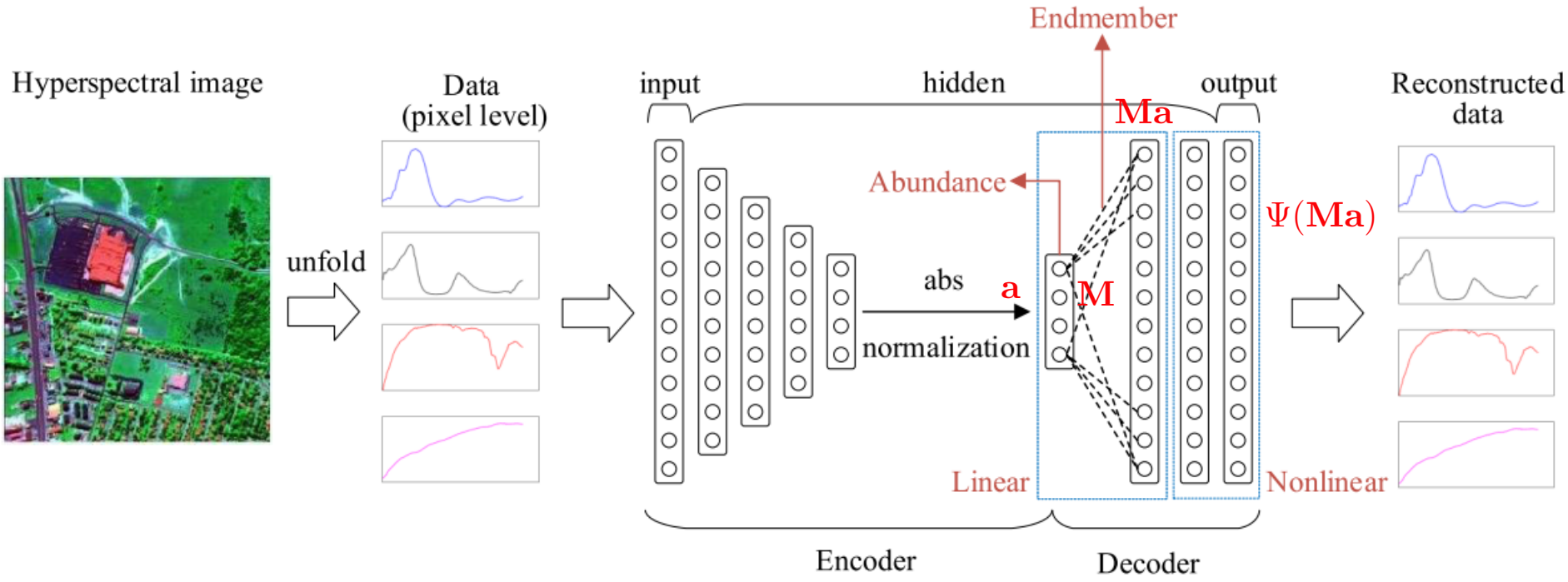
Deep autoencoder based unmixing DNNs preserve physical interpretation by modeling the spectral mixture mechanism[1].



- **Encoder** seeks for a low dimensional representation (i.e., abundances);
- **Decoder** contains endmember information and reconstructs the input.

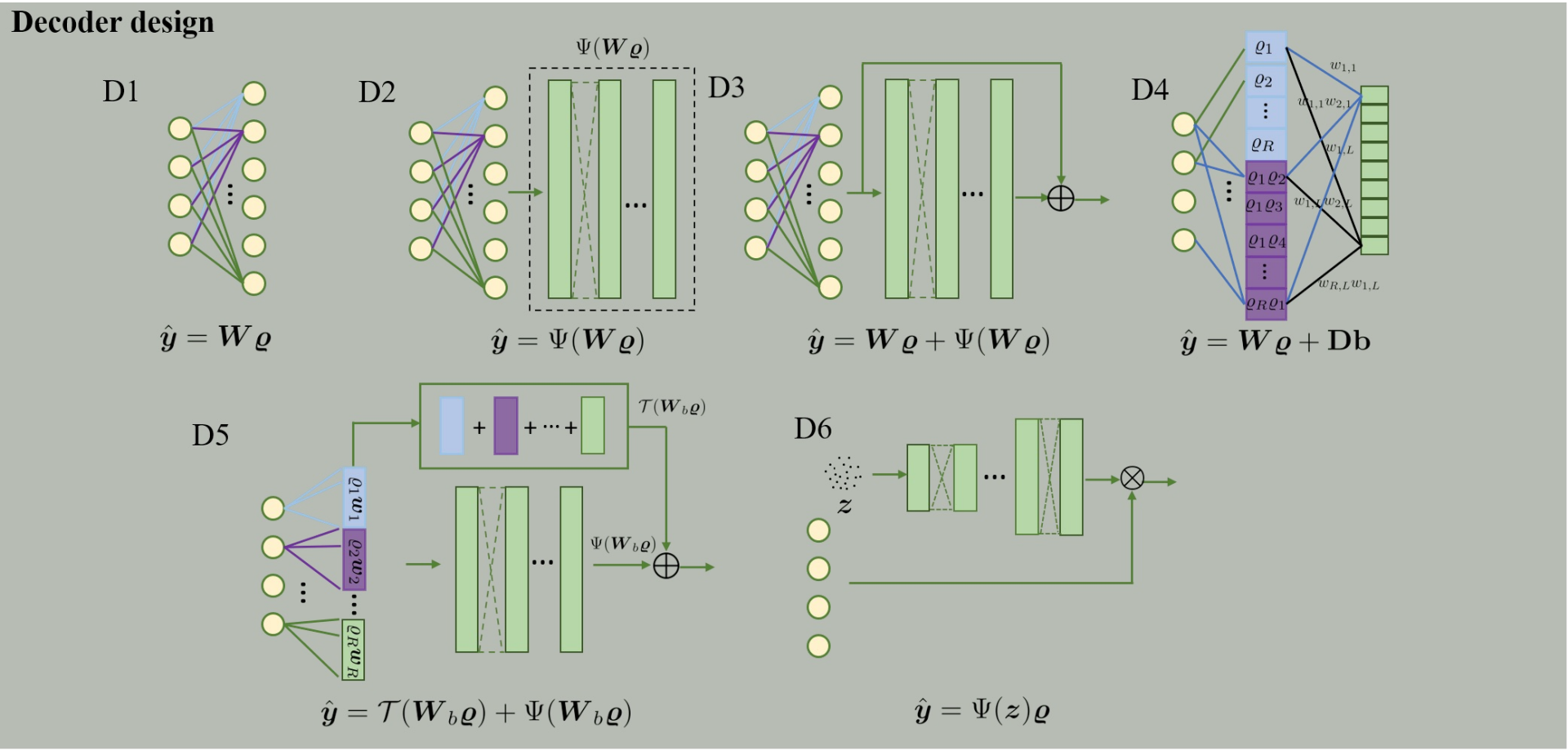
Hybrid Strategies for Hyperspectral Unmixing

An example[1]:



[1] M. Wang, M. Zhao, J. Chen, and S. Rahardja, “Nonlinear unmixing of hyperspectral data via deep autoencoder networks,” IEEE Geosci. Remote Sens. Lett., vol. 16, no. 9, pp. 1467–1471, 2019.

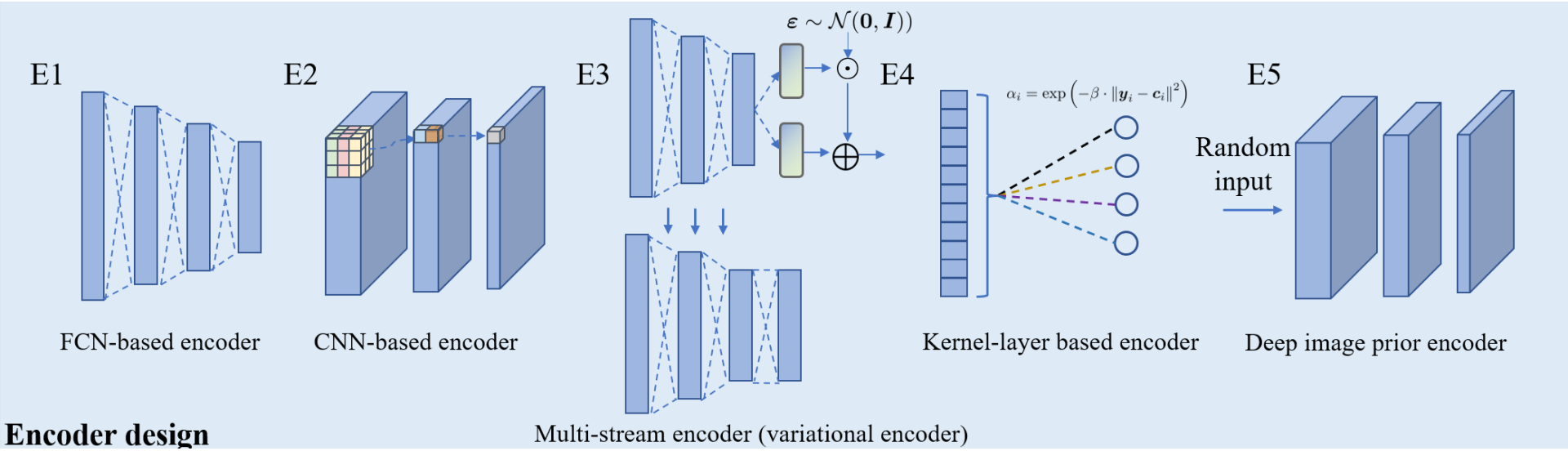
Hybrid Strategies for Hyperspectral Unmixing



Design of decoders to learn models from data

Hybrid Strategies for Hyperspectral Unmixing

Separately show the design of encoders and decoders, to illustrate their extra flexibility and combination potential.



Design of encoders to learn priors from data

Hybrid Strategies for Hyperspectral Unmixing

Examples:

- **E2 + D5:** The 3D-CNN based autoencoder network for additive nonlinearity unmixing[1];
- **E1 + D1:** A denoising autoencoder with sparsity for spectral unmixing[2];
- **E4 + D3/D4:** The kernelization and cross product layer based autoencoder for bilinear and post nonlinearity unmixing[3];
- **E3 + D6:** The probabilistic generative model based autoencoder for spectral unmixing with spectral variability[4].

[1] M. Zhao, M. Wang, J. Chen, and S. Rahardja, “Hyperspectral Unmixing for Additive Nonlinear Models With a 3-D-CNN Autoencoder Network”. In: IEEE Trans. Geosci. Remote Sens. 60 (2022), 1–15.
[2] Y. Qu and H. Qi, “uDAS: An untied denoising autoencoder with sparsity for spectral unmixing”. In: IEEE Trans. Geosci. Remote Sens. 57.3 (2018), pp. 1698–1712.
[3] K. T. Shahid and I. D. Schizas, “Unsupervised Hyperspectral Unmixing via Nonlinear Autoencoders”. In: IEEE Trans. Geosci. Remote Sens. 60 (2021), 1–13.
[4] S. Shi, M. Zhao, L. Zhang, Y. Altmann, and J. Chen, “Probabilistic Generative Model for Hyperspectral Unmixing Accounting for Endmember Variability”. In: IEEE Trans. Geosci. Remote Sens. 60 (2022), 1–15.

Outline

Applications in speech processing

- **Background**

Colin Cherry (1953)

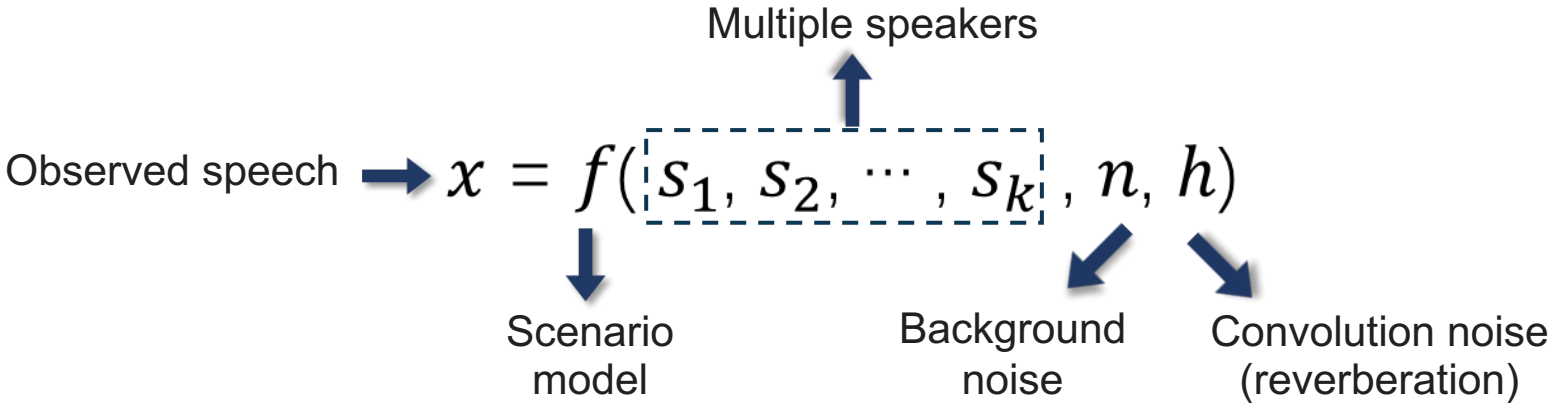
Cocktail party problem

- How can machines, like humans, robustly recognize a target speaker's speech in the presence of multiple interferences?

- Background noise
- Reverberation
- Multiple speaker

Applications in speech processing

- **Problem definition -- In complex acoustic scenarios with multiple people interacting**



- When K=1, only background noise exists, the problem degenerates into **speech denoising**.

$$x = f(s, y) \approx s + y$$

- When K=1, only convolutional noise exists, the problem degenerates into **speech dereverberation**. √

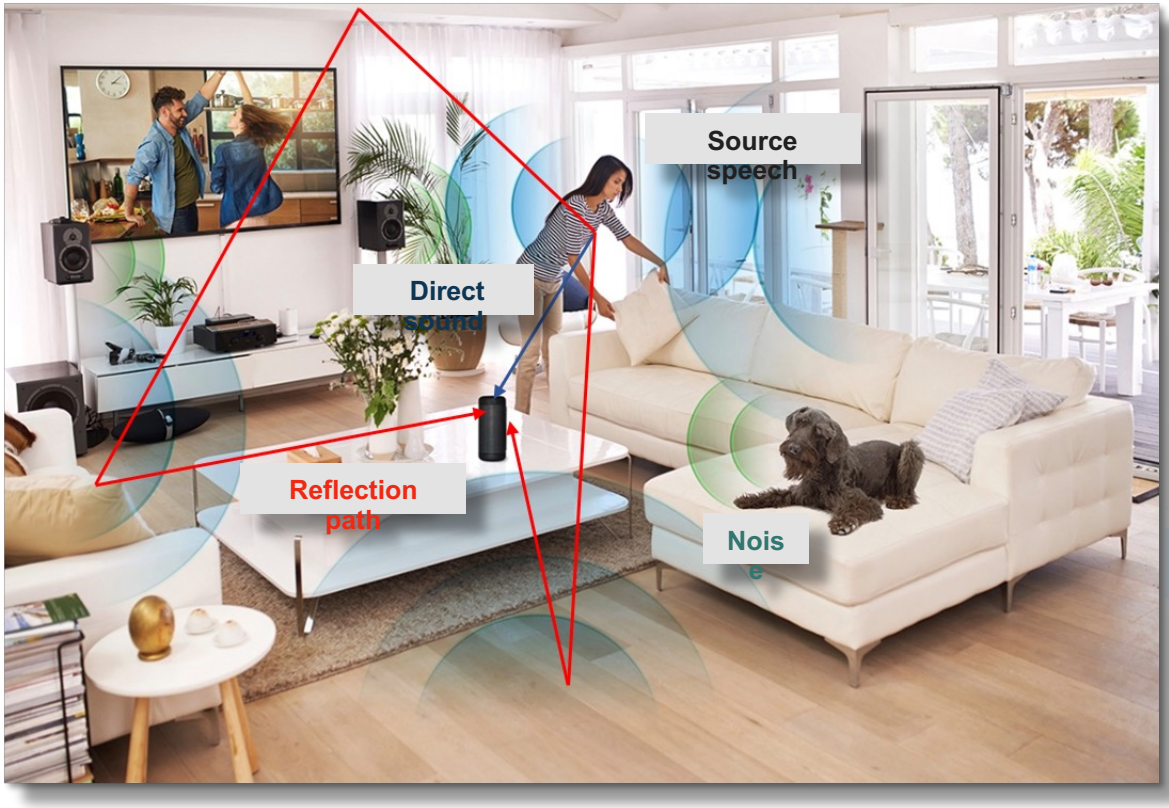
$$x = f(s, h) \approx s * h$$

- When K>1, with no noise interference, the problem degenerates into **speech separation**. √

$$x = f(s_1, s_2, \cdots, s_k) \approx s_1 + s_2 + \cdots + s_k$$

Application in speech dereverberation

■ Reverberation causes



Direct Sound

- The **shortest path** from the sound source to the microphone
- Preserves speech clarity and localization information

Early Reflections

- First few reflections from walls, floor, and ceiling
- **Arrival delay < 50 ms**

Late reverberation

- Residual energy accumulated from multiple reflections
- **Arrival delay > 50 ms**: forms a “reverberant tail”.

[1] P. A. Naylor and N. D. Gaubitch, Speech Dereverberation. Noise Control Engineering Journal, 2011, vol. 59.

Application in speech dereverberation

Early reflections

- **Improves speech quality and intelligibility, contributing to a sense of envelopment.**

Late Reverberation

- **Causes spectral overlap of speech signals, reducing intelligibility.**

The goal of speech dereverberation

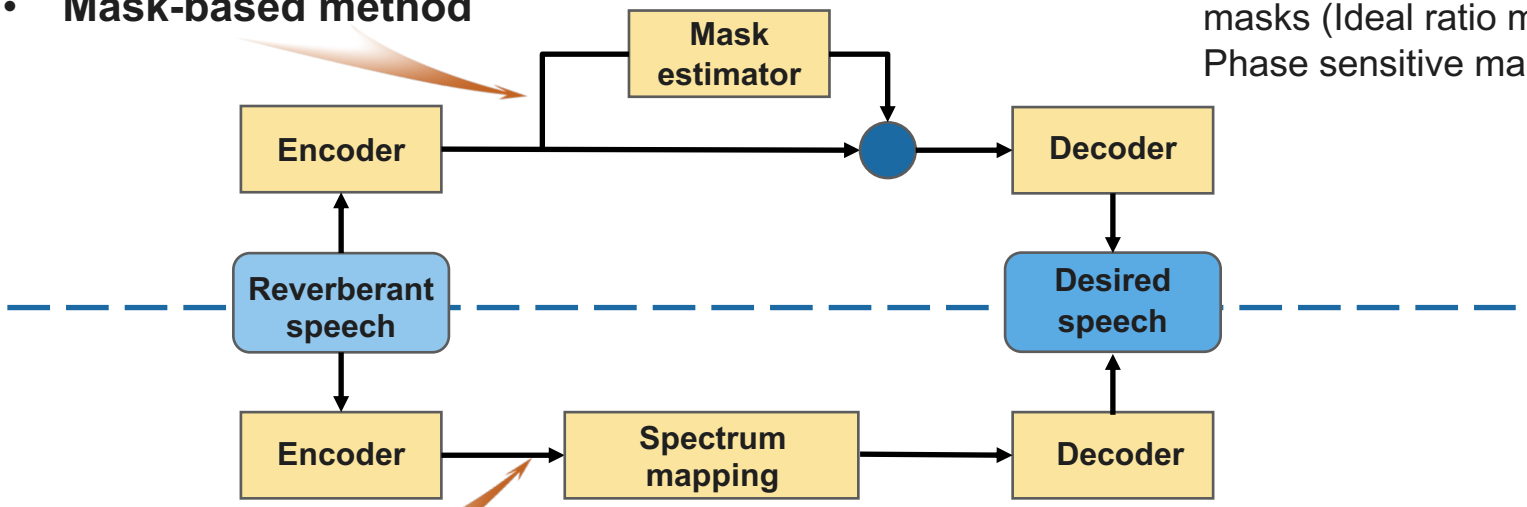
- Remove **late reverberation components**
- preserving the direct sound and early reverberation components.

Application in speech dereverberation

■ Speech dereverberation with data-driven methods

➢ Supervised learning problem

- **Mask-based method**



- **Mapping-based method**

✓ Predict time–frequency masks (Ideal ratio mask, Phase sensitive mask...)

✓ directly map reverberant → clean speech (DNN, BLSTM, Transformer...)

➢ Challenges

✓ Generalization across rooms, speakers, and devices
✓ Limited real-world paired training data
✓ Reverberation often co-occurs with noise → overlap with speech denoising

Application in speech dereverberation

■ **Speech dereverberation with physics-based method**

➢ Addresses the dereverberation problem based on the speech convolution model

• **Single-channel**	• **Multi-channel**
✓ Acoustic channel equalization-based approach	✓ Beamforming approach
✓ Deconvolution approach (RIR is known)	✓ Multi-channel linear prediction approach

Application in speech dereverberation

- **Speech Dereverberation with Deconvolution Regularized by Denoising**
  - Signal model is given by $x(t) = h(t) * s(t) + y(t) \longrightarrow \mathbf{x} = \mathbf{H}\mathbf{s} + \mathbf{y}$
  - The fidelity term of dereverberation process can be formulated by an inverse problem

$$\min_{\mathbf{s}} \ \|\mathbf{x} - \mathbf{H}$$

$$H = \begin{bmatrix} h_0 & h_{N-1} & \cdots & h_2 & h_1 \\ h_1 & h_0 & \cdots & \cdots & h_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ h_{N-2} & h_{N-3} & \cdots & h_0 & h_{N-1} \\ h_{N-1} & h_{N-2} & \cdots & h_1 & h_0 \end{bmatrix}$$

Even with ***H*** available, single-chan
noise remains an *ill-posed problem*

  - Incorporate the regularization term to stabilize the solution process

$$\min_{\mathbf{s}} \ \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 + \mathcal{R}(\mathbf{s})$$ Regularization term

RED: $\mathcal{R}(\mathbf{s}) = \frac{1}{2}\langle \mathbf{s}, \mathbf{s} - f(\mathbf{s}) \rangle$

$$\min_{\mathbf{s}} \ \|\mathbf{x} - \mathbf{H}\mathbf{s}\|^2 + \frac{\beta}{2}\mathbf{z}^\top(\mathbf{z} - f(\mathbf{z}))$$
$$\text{s.t.} \ \ \mathbf{z} = \mathbf{s}.$$

Auxiliary variable

[1] H. Hu, Z. Yang, J. Chen, and L. Zhang, “Speech dereverberation with deconvolution regularized by denoising,” in Asia Pacific Signal, Inf. Process. Assoc. Annual Summit Conf. (APSIPA ASC). IEEE, 2024, pp. 1–6.

Application in speech dereverberation

- **Speech Dereverberation with Deconvolution Regularized by Denoising**
  - Signal model is given by $x(t) = h(t) * s(t) + y(t) \longrightarrow \mathbf{x} = \mathbf{H}\mathbf{s} + \mathbf{y}$
  - The fidelity term of dereverberation process can be formulated by an inverse problem

$$\min_{\mathbf{s}} \ \|\mathbf{x} - \mathbf{H}\mathbf{s}\|^2$$

Even with ***H*** available, single-channel observation under additive noise remains an *ill-posed problem*.

  - Incorporate the regularization term to stabilize the solution process

$$\min_{\mathbf{s}} \ \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 + \mathcal{R}(\mathbf{s})$$ Regularization term

RED: $\mathcal{R}(\mathbf{s}) = \frac{1}{2}\langle \mathbf{s}, \mathbf{s} - f(\mathbf{s}) \rangle$

$$\min_{\mathbf{s}} \ \|\mathbf{x} - \mathbf{H}\mathbf{s}\|^2 + \frac{\beta}{2}\mathbf{z}^\top(\mathbf{z} - f(\mathbf{z}))$$
$$\text{s.t.} \ \ \mathbf{z} = \mathbf{s}.$$

Auxiliary variable

[1] H. Hu, Z. Yang, J. Chen, and L. Zhang, “Speech dereverberation with deconvolution regularized by denoising,” in Asia Pacific Signal, Inf. Process. Assoc. Annual Summit Conf. (APSIPA ASC). IEEE, 2024, pp. 1–6.

Application in speech dereverberation

- **Tackle the problem via the Half-Quadratic Splitting (HQS) algorithm**

$$\mathcal{L}(\mathbf{s},\mathbf{z}) = \|\mathbf{x}-\mathbf{H}\mathbf{s}\|^2 + \frac{\rho}{2}\|\mathbf{s}-\mathbf{z}\|^2 + \frac{\beta}{2}\,\mathbf{z}^\top\big(\mathbf{z}-f(\mathbf{z})\big)$$

- With respect to **s**:

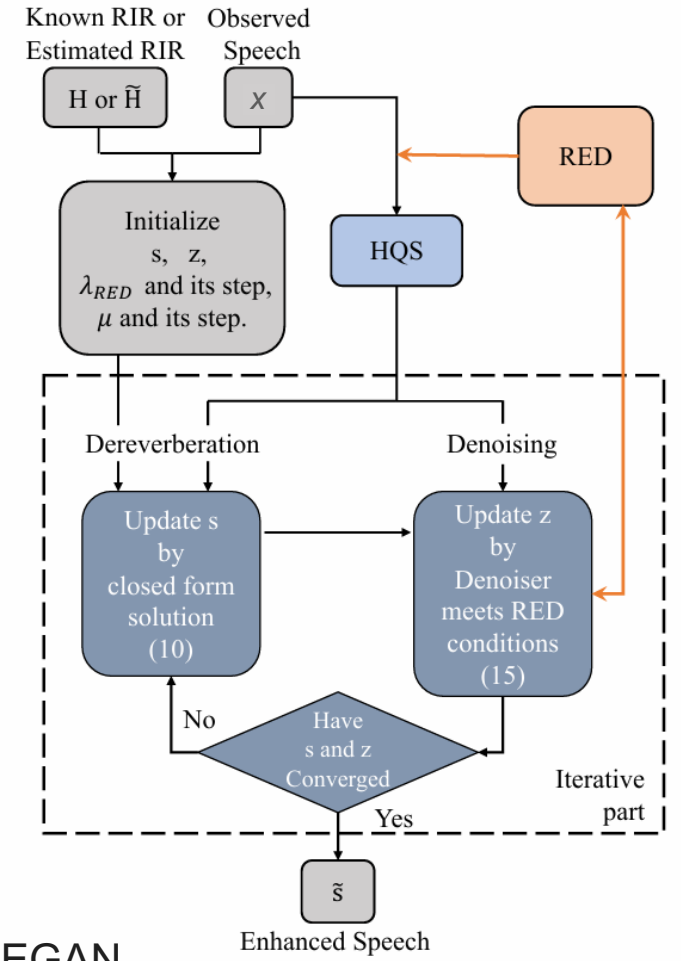
$$\mathbf{s}^{(\ell+1)} = \arg\min_{\mathbf{s}} \ \|\mathbf{x}-\mathbf{H}\mathbf{s}\|^2 + \frac{\rho}{2}\|\mathbf{s}-\mathbf{z}^{(\ell)}\|^2$$

$$\mathbf{s}^{(\ell+1)} = \left(\mathbf{H}^\top\mathbf{H} + \frac{\rho}{2}\mathbf{I}\right)^{-1}\left(\mathbf{H}^\top\mathbf{x} + \frac{\rho}{2}\mathbf{z}^{(n)}\right)$$

- With respect to **z**:

$$\mathbf{z}^{(n+1)} = \arg\min_{\mathbf{z}} \ \frac{\rho}{2}\|\mathbf{s}^{(n+1)}-\mathbf{z}\|^2 + \frac{\beta}{2}\,\mathbf{z}^\top\big(\mathbf{z}-f(\mathbf{z})\big)$$

$$\mathbf{z}^{(\ell+1,i)} = \frac{\rho}{\rho+\beta}\,\mathbf{s}^{(\ell+1,i)} + \frac{\beta}{\rho+\beta}\,f\left(\mathbf{z}^{(\ell+1,i)}\right)$$



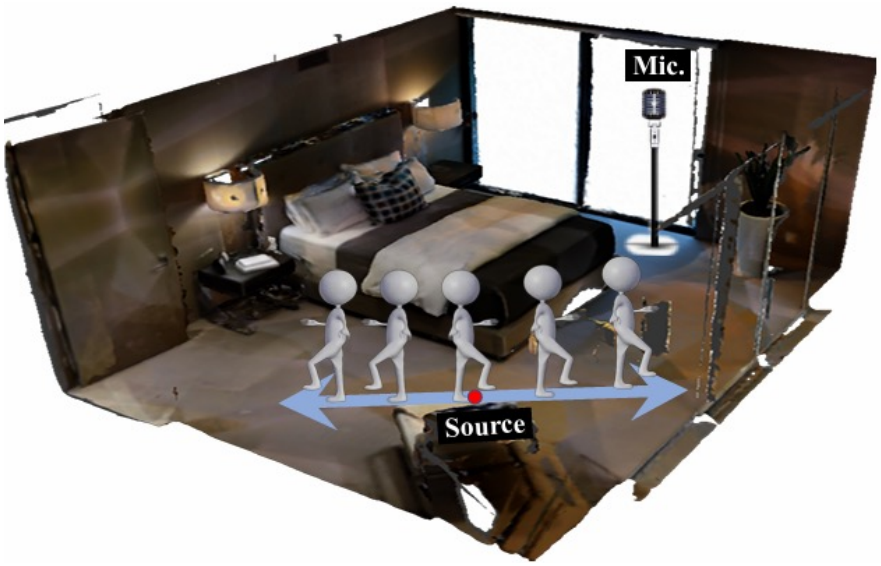
Pre-trained denoiser: SEGAN

[1] H. Hu, Z. Yang, J. Chen, and L. Zhang, “Speech dereverberation with deconvolution regularized by denoising,” in Asia Pacific Signal, Inf. Process. Assoc. Annual Summit Conf. (APSIPA ASC). IEEE, 2024, pp. 1–6.

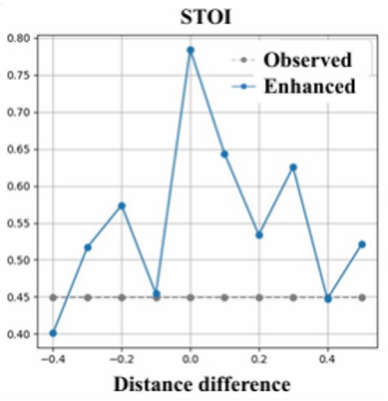
Application in speech dereverberation

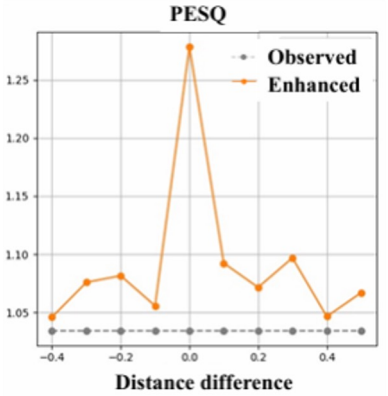
Experiments

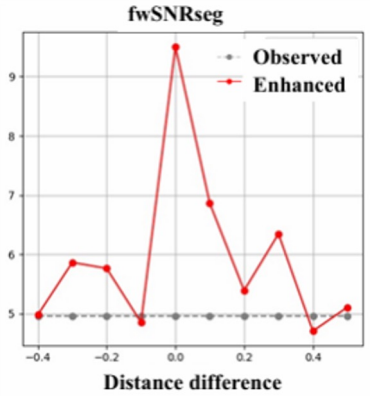
Simulated scenario of **RIR positional mismatch** based on SoundSpaces combined with Matterport3D



Room size	Randomly selected from the Matterport3D datase
Clean speech corpus	Voice Bank corpus
RIR generation	SoundSpaces (ray tracing)
Source–microphone position	Random placement
T60 values	Approximately 190, 430, 890 ms (set around target values)
Noise levels	0, 10, 20 dB (White Gaussain Noise)
RIR mismatch introduction	**Fix the microphone and move the source to simulate positional shift**







Takeaways

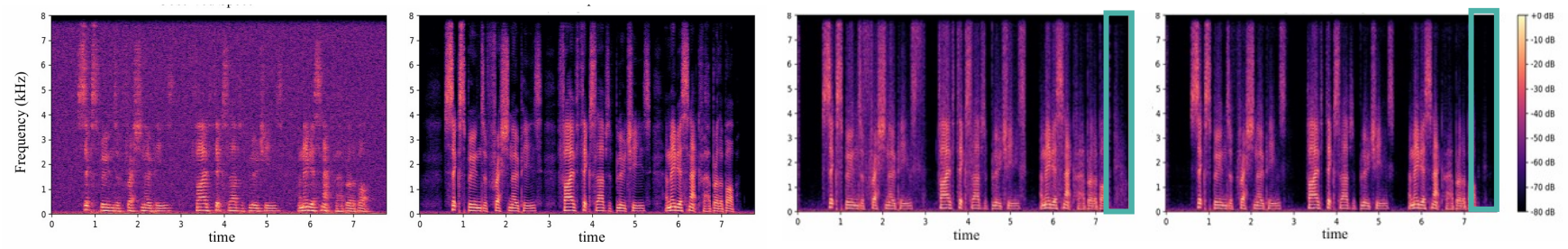
Robust against and positional RIR mismatches, effectively stabilizing the ill-posed system

Application in speech dereverberation

Experiments

SNR (dB)	Error(α) (%)	T_{60} → Methods↓	190 ms STOI ↑	190 ms PESQ ↑	190 ms F-SNR (dB)↑	190 ms Iter. ↓	430 ms STOI ↑	430 ms PESQ ↑	430 ms F-SNR (dB)↑	430 ms Iter. ↓	890 ms STOI ↑	890 ms PESQ ↑	890 ms F-SNR (dB)↑	890 ms Iter. ↓
0 dB	0%	Observed	0.631	1.035	3.772	-	0.548	1.028	3.482	-	0.554	1.048	3.480	-
		JE	0.752	1.074	5.965	81	0.634	1.042	3.400	250	0.728	1.099	6.052	222
		Proposed-s	0.763	1.099	6.318	84	0.648	1.039	3.996	238	0.755	1.156	7.368	270
		Proposed-d	**0.821**	**1.291**	**9.737**	**35**	**0.769**	**1.190**	**7.349**	**38**	**0.784**	**1.332**	**8.782**	**40**
10 dB	0%	Observed	0.718	1.078	5.721	-	0.607	1.034	4.911	-	0.611	1.100	4.496	-
		JE	0.862	1.625	10.636	**36**	0.801	1.154	7.250	105	0.827	1.698	10.796	124
		Proposed-s	0.868	1.683	11.065	**36**	0.811	1.208	8.061	112	0.828	1.721	11.869	103
		Proposed-d	**0.881**	**1.868**	**11.896**	**36**	**0.836**	**1.368**	**10.116**	**32**	**0.848**	**1.761**	**11.994**	**36**
20 dB	0%	Observed	0.772	1.313	7.263	-	0.659	1.095	6.205	-	0.635	1.209	5.149	-
		JE-0	0.897	2.104	11.392	**9**	0.879	1.902	12.321	40	0.873	2.109	13.483	**5**
		Proposed-s	**0.901**	2.206	11.430	10	**0.888**	2.049	12.932	**36**	0.881	2.482	14.962	24
		Proposed-d	0.900	**2.266**	**11.464**	10	**0.888**	**2.118**	**13.009**	**36**	**0.883**	**2.644**	**15.489**	8
	15%	JE	0.891	2.033	**11.050**	**7**	0.882	1.774	11.050	44	0.862	1.887	12.010	43
		Proposed-d	**0.896**	**2.188**	11.004	9	**0.888**	**2.068**	**11.759**	**36**	**0.876**	**2.229**	**12.455**	**8**

- JE[1]: Joint enhancement frame work via the PnP strategy to simultaneously execute dereverberation and denoising



Observed Clean JE Proposed

[1] A. Raikar, S. Basu, L. Pandey, and R. Hegde, "Multi channel joint dereverberation and denoising using deep priors," IEEE India Council International Conference (INDICON), pp. 1–6, 2018.

Application in speech dereverberation

■ Weighted Prediction Error (WPE) Algorithm -- Multi-channel method

- **Multi-channel linear prediction**

$$\mathbf{b}_1(n-D,k)$$

$$\begin{bmatrix} B_1(n-D,k) & \cdots & B_1(n-D-L,k) \\ \vdots & \ddots & \vdots \\ B_Q(n-D,k) & \cdots & B_Q(n-D-L,k) \end{bmatrix}$$

  - n: Time frame indices
  - k: Frequency bin indices
  - L: Filter order
  - D: Time delay
  - Q: Number of channels

- Observed signal:

$$\overline{\mathbf{b}}(n-D,k) = \mathrm{col}\{\underbrace{\mathbf{b}_1(n-D,k),\ldots,\mathbf{b}_Q(n-D,k)}_{L\times Q}\}$$

- Multi-channel autoregressive model for channel modeling

$$B_{\mathrm{ref}}(n,k) = \underline{\hat{S}(n,k)} + \underline{\overline{\mathbf{w}}^{\mathrm{H}}(k)\,\overline{\mathbf{b}}(n-D,k)}$$

Desired speech **Late reverberation part**

- Use Gaussian distribution as clean speech prior model, and Maximum likelihood method to construct cost function:

$$\mathcal{J}\left(\{\overline{\mathbf{w}}(k)\}_{k=1}^{K},\{\sigma(k)\}_{k=1}^{K}\right) = \sum_{k=1}^{K}\sum_{n=1}^{N}\frac{\left|B_{\mathrm{ref}}(n,k)-\overline{\mathbf{w}}^{\mathrm{H}}(k)\,\overline{\mathbf{b}}(n-D,k)\right|^2}{\sigma(n,k)}+\log\pi\sigma(n,k)$$

Alternately estimate

➢ Challenge: Lack data priors ⟶ reduces performance under complex conditions

[1] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, “Speech dereverberation based on variance-normalized delayed linear prediction,” IEEE/ACM Trans. Audio, Speech, Lang. Process., vol. 18, no. 7, pp. 1717–1731, Sep. 2010.

Application in speech dereverberation

■ Current hybrid strategies

- Concatenation
  - Data-driven front-end + physics-based back-end
  - Physics-based front-end + data-driven back-end

- Neural networks estimating intermediate variables in the WPE method
  e.g. utilize deep neural network (DNN) to estimate spectrum

$$\mathcal{J}_{\mathrm{WPE}}\left(\{\overline{\mathbf{w}}(k)\}_{k=1}^{K}, \{\boldsymbol{\sigma}(k)\}_{k=1}^{K}\right) = \sum_{k=1}^{K}\sum_{n=1}^{N}\frac{|\hat{S}(n,k)|^2}{\sigma(n,k)} + \log \pi\sigma(n,k)$$

How to integrate the advantages of the two different approaches by formulating the optimization problem from the very beginning?

[1] K. Kinoshita, M. Delcroix, H. Kwon, T. Mori, and T. Nakatani, “Neural network-based spectrum estimation for online WPE dereverberation,” in Proc. Interspeech, 2017, pp. 384–388.

Application in speech dereverberation

■ Integrating Data Priors to the WPE method

→ Conduct speech dereverberation under complex conditions

Complex conditions: Diffuse noise, **Dereverberation + denoise**

- The desired speech signal：$R(n,k)=X_{\mathrm{ref}}(n,k)-\overline{\mathbf{w}}^{\mathrm{H}}(k)\overline{\mathbf{x}}(n-D,k)-V(n,k)$
- Insert speech prior via **regularization by denoising (RED)**
- The cost function: $\mathcal{J}_{\mathrm{WPE\_Reg}}\left(\{\overline{\mathbf{w}}(k)\}_{k=1}^{K},\{\boldsymbol{\sigma}(k)\}_{k=1}^{K}\right)=\mathcal{J}_{\mathrm{WPE}}+\beta\mathcal{J}_{\mathrm{Reg}}\left(\mathbf{R}\right)$
- The regularization term: $\mathcal{J}_{\mathrm{Reg}}\left(\mathbf{R}\right)=\frac{1}{2}\langle\mathbf{R},\mathbf{R}-\Omega(\mathbf{R})\rangle$ ← Deep denoiser output
- The **full problem formulation**:

$$\min_{\overline{\mathbf{w}}(k),\boldsymbol{\sigma}(k),\mathbf{R},\mathbf{V}}\sum_{k=1}^{K}\sum_{n=1}^{N}\frac{|\hat{S}(n,k)|^2}{\sigma(n,k)}+\log\pi\sigma(n,k)+\frac{\beta}{2}\langle\mathbf{R},\mathbf{R}-\Omega(\mathbf{R})\rangle$$

$$\text{s.t.}\quad R(n,k)=X_{\mathrm{ref}}(n,k)-\overline{\mathbf{w}}^{\mathrm{H}}(k)\overline{\mathbf{x}}(n-D,k)-V(n,k)\ \text{ for }\ n=1,\ldots,N\ \text{and}\ k=1,\ldots,K,$$

[1] Z. Yang, W. Yang, K. Xie and J. Chen, “Integrating Data Priors with Weighted Prediction Error for Speech Dereverberation”, IEEE/ACM Transactions on Audio, Speech, Language and Processing, vol. 32, pp. 3908-3923, 2024.
[2] Z. Yang, W. Yang, K. Xie and J. Chen, “Speech Dereverberation Using Weighted Prediction Error with Prior Learnt from Data”, in Proc. EUSIPCO, pp. 356-360, 2023

Application in speech dereverberation

Integrating Data Priors to the WPE method

→ Solving method

- The corresponding (scaled) **augmented Lagrangian function**

$$\mathscr{L}\left(\{\overline{\mathbf{w}}(k)\}_{k=1}^{K},\{\boldsymbol{\sigma}(k)\}_{k=1}^{K},\mathbf{R},\mathbf{V},\mathbf{P}\right)=\mathcal{J}_{\mathrm{WPE}}+\frac{\beta}{2}\langle\mathbf{R},\mathbf{R}-\Omega(\mathbf{R})\rangle+\frac{\rho}{2}\sum_{k=1}^{K}\sum_{n=1}^{N}\Big(\big|[X_{\mathrm{ref}}(n,k)-\overline{\mathbf{w}}^{\mathrm{H}}(k)\times\overline{\mathbf{x}}(n-D,k)]-V(n,k)-R(n,k)+P(n,k)\big|^{2}-|P(n,k)|^{2}\Big)$$

  - β: Trade-off parameter
  - ρ: Penalty parameter
  - *Scaled dual variable*

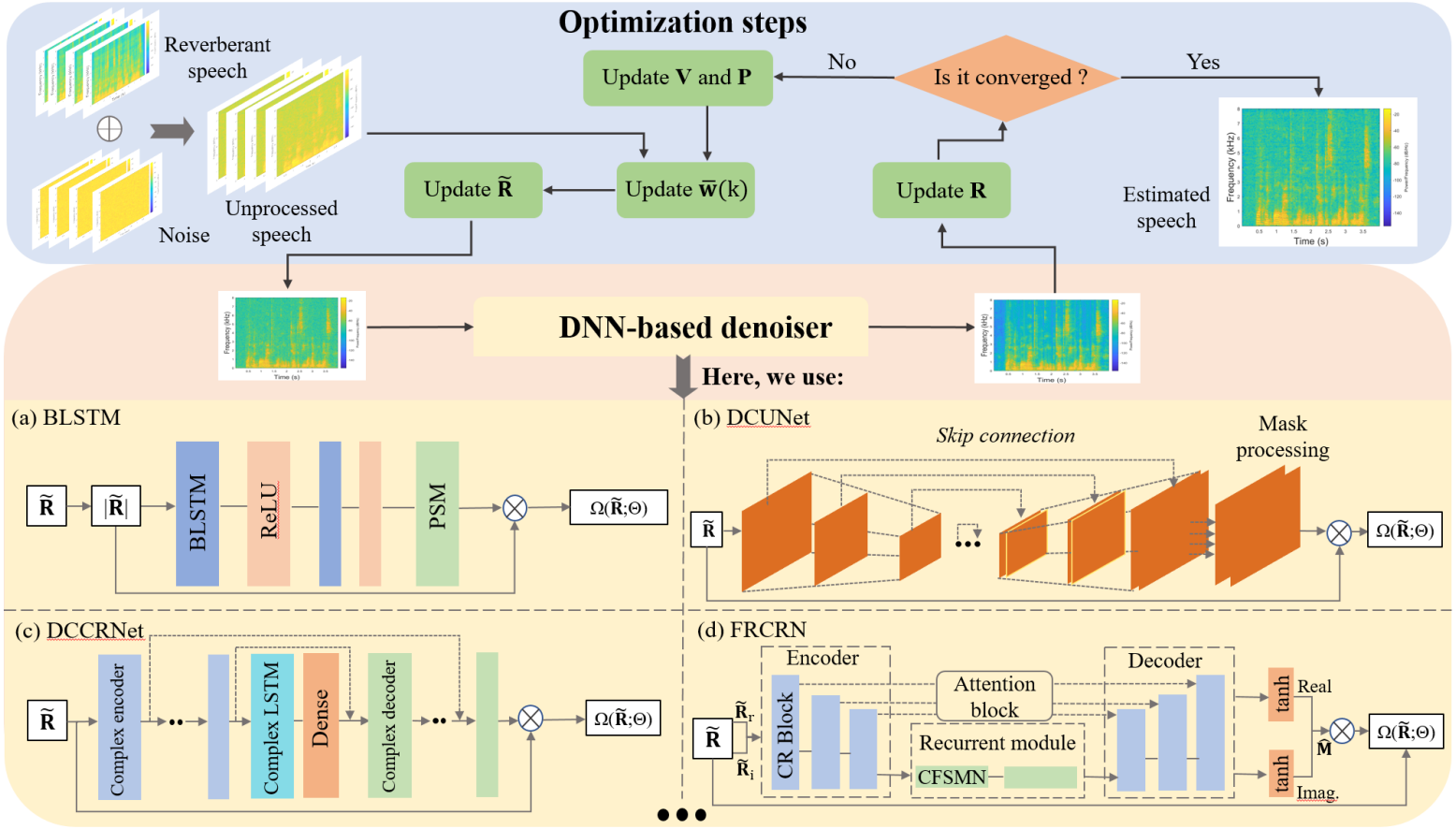
- Using variable splitting method to **solve the problem**
  - With respect to $\overline{\mathbf{w}}(k)$: $\overline{\mathbf{w}}^{(\ell+1)}(k)=\left[R_{\overline{\mathbf{x}}}^{(\ell+1)}(k)\right]^{-1}\mathbf{p}_{\overline{\mathbf{x}}}^{(\ell+1)}(k)$
  - With respect to $\boldsymbol{\sigma}(k)$: $\sigma(n,k)=\max\left\{|\hat{S}(n,k)|^{2},\epsilon\right\}$
  - With respect to $\mathbf{R}$: $\mathbf{R}^{(\ell+1,i+1)}=\mu\tilde{\mathbf{R}}^{(\ell+1)}+(1-\mu)\Omega\left(\mathbf{R}^{(\ell+1,i)}\right)$ ← *The fixed-point iteration*
  - With respect to $\mathbf{V}$: $\mathbf{V}^{(\ell+1)}=\hat{\mathbf{S}}^{(\ell+1)}-\mathbf{R}^{(\ell+1)}+\mathbf{P}^{(\ell)}$
  - With respect to $\mathbf{P}$: $\mathbf{P}^{(\ell+1)}=\mathbf{P}^{(\ell)}+\hat{\mathbf{S}}^{(\ell+1)}-\mathbf{V}^{(\ell+1)}-\mathbf{R}^{(\ell+1)}$

[1] Z. Yang, W. Yang, K. Xie and J. Chen, “Integrating Data Priors with Weighted Prediction Error for Speech Dereverberation”, IEEE/ACM Transactions on Audio, Speech, Language and Processing, vol. 32, pp. 3908-3923, 2024.

[2] Z. Yang, W. Yang, K. Xie and J. Chen, “Speech Dereverberation Using Weighted Prediction Error with Prior Learnt from Data”, in Proc. EUSIPCO, pp. 356-360, 2023

Application in speech dereverberation

The overall framework



➢ **Takeaway:** The Optimization framework remains the same

The denoiser module is flexible

[1] Z. Yang, W. Yang, K. Xie and J. Chen, “Integrating Data Priors with Weighted Prediction Error for Speech Dereverberation”, IEEE/ACM Transactions on Audio, Speech, Language and Processing, vol. 32, pp. 3908-3923, 2024.

Application in speech dereverberation

■ Experimental setting

Room type	Room A	Room B
Room length (m)	8-13	15-20
Room width (m)	8-13	15-20
Room height (m)	2.8-3.8	3-4
T60 (ms)	400-800	800-1200
Minimum distance from source to wall (m)	0.5	0.8
Minimum distance from source to microphone (m)	0.8	1.3
Microphone array	Linear array with 4 channels	
	Inner distance = 40 mm	
	Microphone type = Omnidirectional	

Room A

- **Medium room with moderate reverberation**

Room B

- **Large room with high reverberation**

Corpus:WSJ0

- **Evaluation metrics**

Perceptual Evaluation of Speech Quality (PESQ) ↑

Short-Time Objective Intelligibility (STOI) ↑

Cepstral Distance(CD) ↓

Frequency-weighted segmental Signal-to-Noise Ratio (F-SNR) ↑

Application in speech dereverberation

- **Experimental results in Room A**

Additive noise type	Noise level		SNR = 0 dB			SNR = 10 dB			SNR = 20 dB		
	Method	Denoiser	PESQ	CD	F-SNR	PESQ	CD	F-SNR	PESQ	CD	F-SNR
WGN	Unprocessed	-	1.42	8.47	4.72	1.93	7.97	5.72	2.08	6.69	6.31
	WPE	-	1.31	8.63	4.34	1.87	8.35	5.74	2.28	7.27	7.37
	DNN-WPE	-	1.36	8.49	4.68	1.87	7.75	6.34	2.33	6.67	**8.15**
	Conv-TasNet	-	1.35	9.04	4.78	1.92	8.26	5.84	2.02	7.18	5.94
	SepFormer	-	1.88	7.14	5.78	2.00	6.95	6.60	2.10	5.60	7.03
	Denoise + WPE	BLSTM	1.56	8.63	4.64	1.96	8.28	5.65	2.54	7.17	7.07
		DCUNet	1.28	8.52	4.67	1.65	8.07	5.74	2.03	6.77	6.56
		DCCRNet	1.17	8.51	4.69	1.93	8.04	5.70	1.91	6.77	6.46
		FRCRN	1.95	8.55	4.56	2.22	8.24	5.69	2.45	7.00	6.48
	WPE + Denoise	BLSTM	1.49	7.94	4.00	1.83	7.36	5.16	2.24	6.44	5.99
		DCUNet	1.47	8.88	4.92	2.06	8.10	6.18	2.59	7.30	6.50
		DCCRNet	1.10	8.68	4.23	1.86	9.36	5.73	2.30	8.33	6.08
		FRCRN	1.94	6.82	5.87	2.26	6.65	6.16	2.65	6.15	6.52
	PnPWPE	BLSTM	1.70	6.79	6.35	2.29	6.81	**7.40**	2.66	**5.33**	7.84
		DCUNet	1.41	8.44	5.12	2.18	7.71	6.75	2.64	6.24	7.68
		DCCRNet	1.20	8.89	4.79	2.03	8.36	6.47	2.54	6.66	7.65
		FRCRN	**1.97**	**6.34**	**6.38**	**2.44**	**5.69**	7.15	**2.67**	**5.33**	7.84
Cafe	Unprocessed	-	1.31	6.92	3.04	2.04	6.19	4.60	2.08	5.47	6.07
	WPE	-	1.63	7.05	2.93	2.03	6.19	4.69	2.32	4.75	6.73
	DNN-WPE	-	1.66	6.54	4.77	2.29	5.68	**6.67**	2.37	4.55	**7.77**
	Conv-TasNet	-	1.53	6.91	4.54	1.98	6.14	5.56	2.23	5.43	6.50
	SepFormer	-	1.73	6.54	4.87	2.11	5.82	5.78	2.09	5.15	6.93
	Denoise + WPE	BLSTM	1.04	6.87	2.85	1.71	6.25	4.35	2.33	5.19	6.14
		DCUNet	1.35	6.94	2.98	2.02	6.21	4.47	2.14	5.27	6.10
		DCCRNet	1.32	6.92	3.02	1.87	6.21	4.44	2.00	5.39	6.01
		FRCRN	1.70	7.03	3.10	2.06	6.32	4.55	2.33	5.52	6.10
	WPE + Denoise	BLSTM	0.97	8.20	3.29	1.69	7.17	5.04	2.32	5.14	6.83
		DCUNet	1.67	7.82	4.11	2.13	7.31	5.48	2.48	6.24	7.23
		DCCRNet	1.58	8.49	**4.89**	1.67	7.20	5.92	2.31	7.26	6.79
		FRCRN	1.70	6.44	4.82	2.20	5.91	6.00	**2.58**	4.72	7.46
	PnPWPE	BLSTM	1.54	6.79	3.62	2.24	5.83	5.62	2.45	4.54	7.32
		DCUNet	1.58	6.71	4.05	2.33	5.63	5.78	2.55	4.45	7.21
		DCCRNet	1.82	6.85	4.79	2.25	**5.52**	6.15	2.48	4.50	7.25
		FRCRN	**1.88**	**6.34**	4.41	**2.34**	**5.52**	5.73	**2.58**	**4.33**	7.34

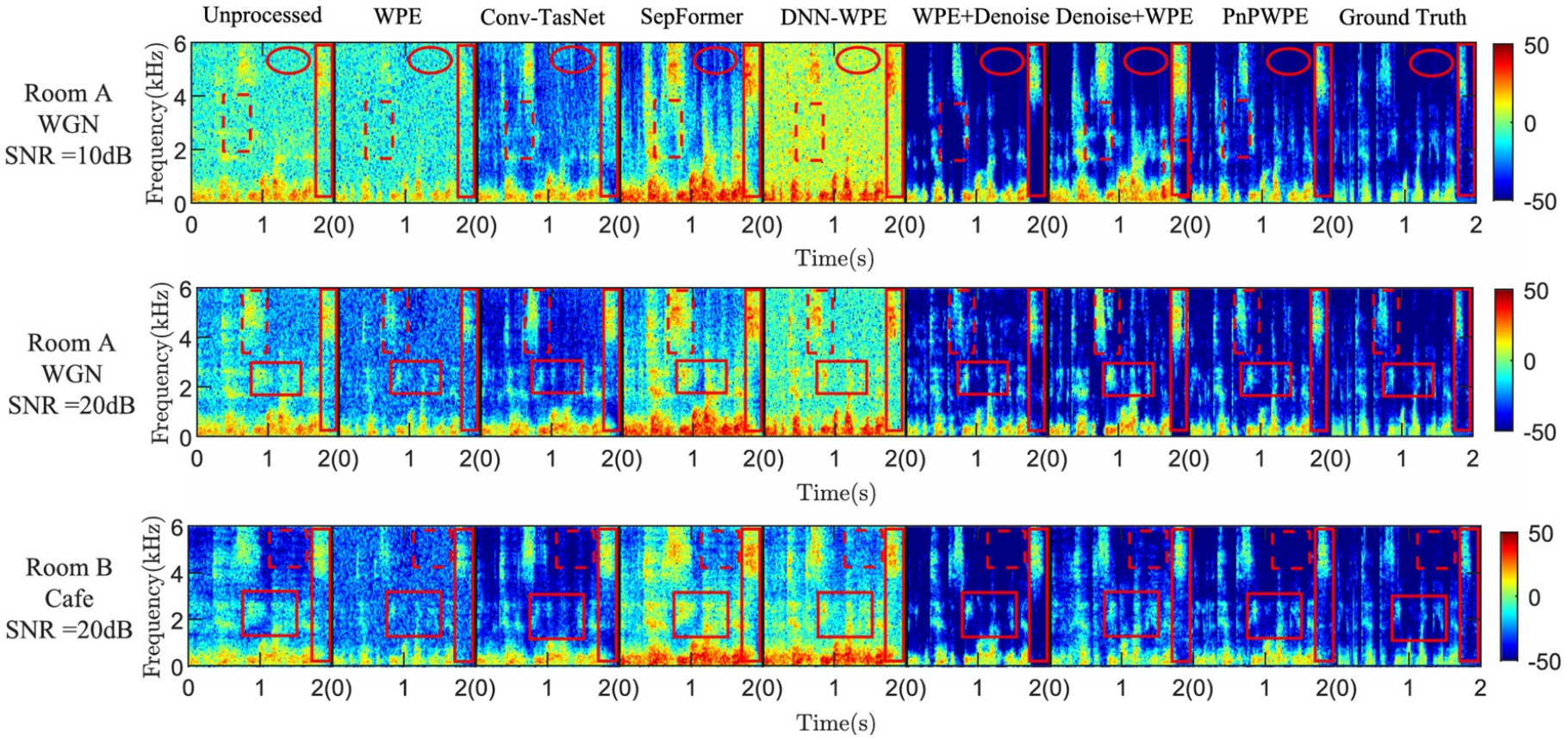
Application in speech dereverberation

- **Experimental results in Room B**

Additive noise type	Noise level		SNR = 0 dB			SNR = 10 dB			SNR = 20 dB		
	Method	Denoiser	PESQ	CD	F-SNR	PESQ	CD	F-SNR	PESQ	CD	F-SNR
WGN	Unprocessed	-	1.21	8.42	4.61	1.48	7.94	5.61	1.71	6.75	6.09
	WPE	-	1.22	8.59	4.32	1.69	8.29	5.73	2.14	7.24	7.37
	DNN-WPE	-	1.75	8.66	4.39	1.74	7.78	5.97	2.14	6.32	**8.53**
	Conv-TasNet	-	0.67	9.11	4.38	1.64	8.26	5.70	1.81	7.29	5.70
	SepFormer	-	1.77	7.19	5.06	1.96	7.07	5.99	1.98	6.18	6.57
	Denoise + WPE	BLSTM	1.47	8.70	4.57	1.77	8.32	5.50	2.21	7.24	6.97
		DCUNet	0.94	8.48	4.57	1.50	8.05	5.52	1.61	6.78	6.25
		DCCRNet	0.75	8.47	4.59	1.23	8.02	5.53	1.44	6.82	6.09
		FRCRN	1.77	8.51	4.47	1.42	8.15	5.45	2.17	7.13	5.79
	WPE + Denoise	BLSTM	1.46	8.00	3.68	1.61	7.47	4.98	1.98	6.51	5.98
		DCUNet	1.16	8.89	4.57	1.81	8.15	6.11	2.24	7.54	6.45
		DCCRNet	0.71	9.68	3.93	1.55	9.41	5.48	2.21	8.39	6.05
		FRCRN	1.78	6.92	5.46	2.01	6.66	5.97	2.39	6.11	6.57
	PnPWPE	BLSTM	1.58	7.02	**5.99**	1.98	**6.38**	7.02	2.28	5.36	7.84
		DCUNet	1.22	8.38	4.95	1.85	7.74	6.51	2.37	6.43	7.76
		DCCRNet	1.19	8.69	4.56	1.82	8.23	6.13	**2.40**	6.73	7.81
		FRCRN	**1.82**	**6.37**	5.96	**2.08**	6.45	**7.11**	2.33	**5.17**	7.58
Cafe	Unprocessed	-	1.33	6.88	3.49	1.57	6.30	4.46	1.67	5.76	5.46
	WPE	-	1.38	7.06	3.26	1.89	6.17	4.50	2.09	4.99	6.01
	DNN-WPE	-	1.40	6.97	4.52	1.95	5.77	5.07	2.14	**4.42**	6.07
	Conv-TasNet	-	1.19	6.78	4.96	1.81	6.32	5.41	1.87	5.76	5.79
	SepFormer	-	1.42	6.73	**5.30**	1.76	5.70	5.44	1.86	5.62	6.43
	Denoise + WPE	BLSTM	0.96	6.85	3.44	1.57	6.42	4.04	1.86	5.53	6.39
		DCUNet	0.94	6.85	3.41	1.68	6.31	4.30	1.70	5.54	5.45
		DCCRNet	0.75	6.87	3.39	1.58	6.29	4.23	1.65	5.70	5.27
		FRCRN	1.47	7.01	3.49	2.03	6.56	4.29	2.14	5.86	5.37
	WPE + Denoise	BLSTM	0.47	8.24	3.20	1.62	7.28	4.80	1.95	5.53	6.39
		DCUNet	1.17	7.80	4.28	1.74	7.18	5.38	2.06	6.76	6.42
		DCCRNet	1.03	8.51	4.89	1.67	7.69	5.30	1.99	7.34	6.08
		FRCRN	1.14	6.52	4.83	1.98	5.99	5.65	2.16	5.23	**6.63**
	PnPWPE	BLSTM	1.37	6.82	3.82	**2.08**	7.28	4.80	2.11	4.88	6.37
		DCUNet	1.42	6.79	3.94	2.00	5.74	5.38	2.14	4.88	6.35
		DCCRNet	**1.50**	6.64	4.50	2.03	**5.60**	**5.69**	2.13	4.85	6.41
		FRCRN	1.41	**6.48**	4.23	**2.08**	5.66	5.62	**2.26**	4.79	6.52

Application in speech dereverberation

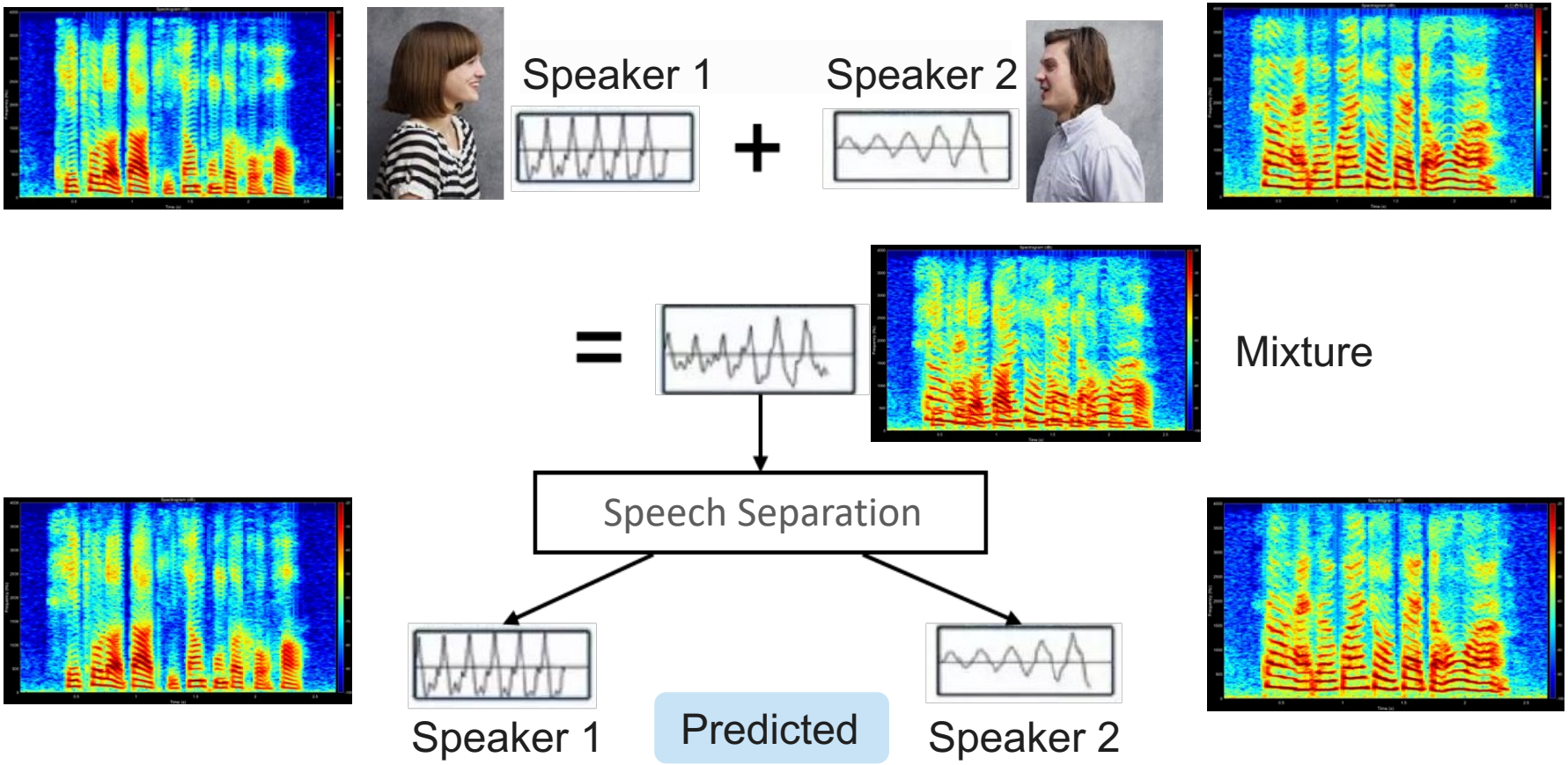
- **Visualization results**



- **Clean**
- **Observed**
- **WPE**
- **Denoiser only**
- **Denoiser+WPE**
- **PnPWPE**

Application in speech separation

- **Speech separation is to separate multiple speakers and reduce mutual interference.**

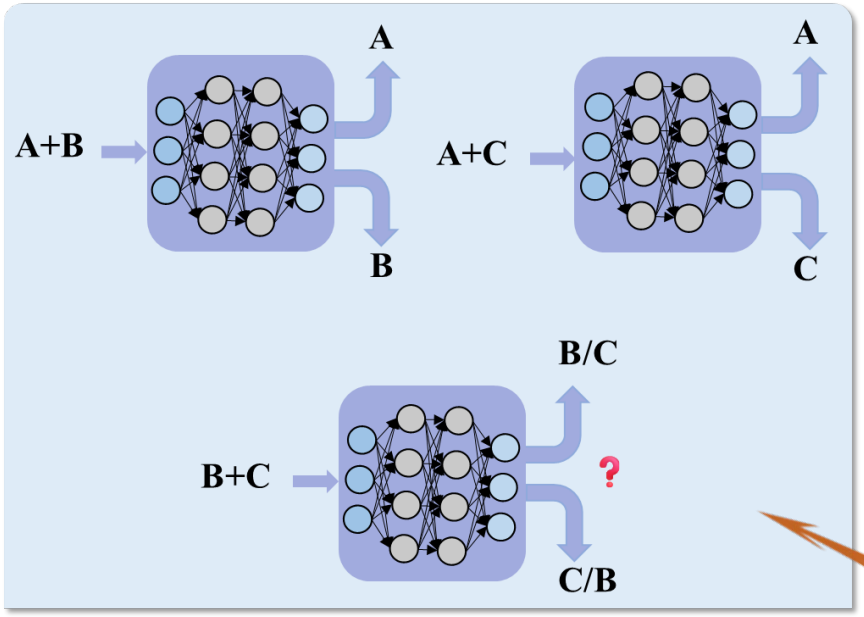


Application in speech separation

Current data-driven methods

➢ Time-Frequency domain

Learn speech characteritics in time-frequency space



The optimization objective is ill-defined, causing gradient reversal and leading to training failure. ✗

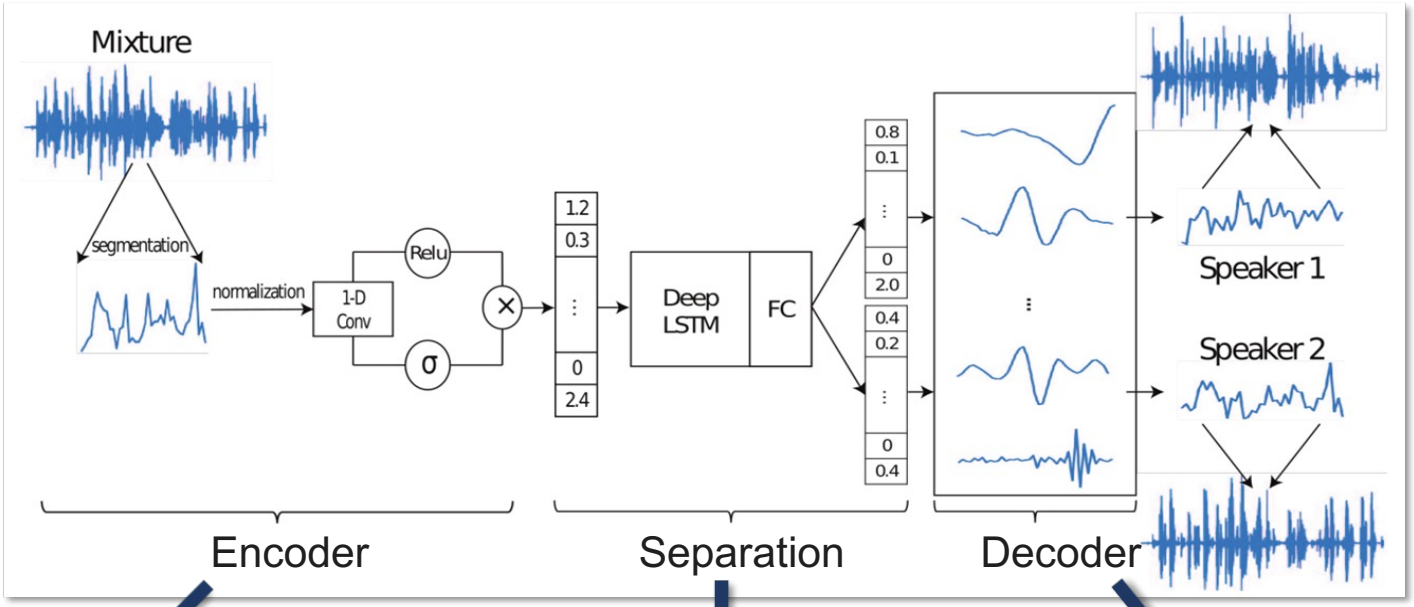
- ✓ Deep clustering
- ✓ Permutation invariant training (**PIT**)

Two perspectives to address the **speaker permutation probelm**

Application in speech separation

- **Current data-driven methods**
  - **Time domain**

    waveform-level modeling



Convert mixture waveform into latent representation

Conv-TasNet -- Temporal convolutional networks

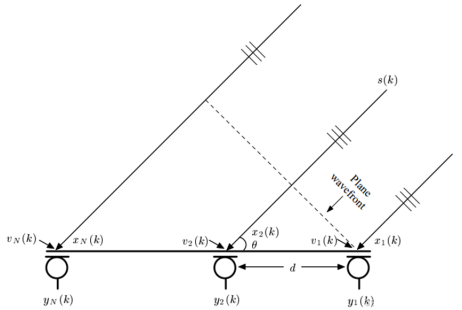
DPRNN -- Dual-path RNN for long sequences

SepFormer -- Transformer-based separation

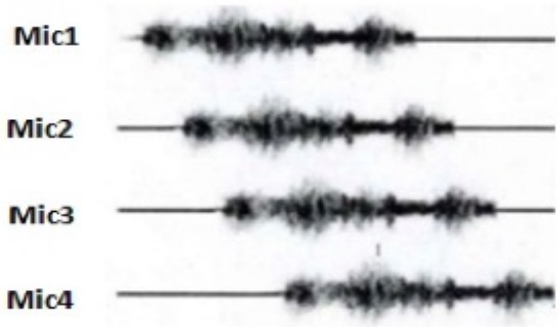
Reconstruct separated waveforms back to time domain

Application in speech separation

- **Current traditional methods**
  - **Spatial filtering -- Beamforming**



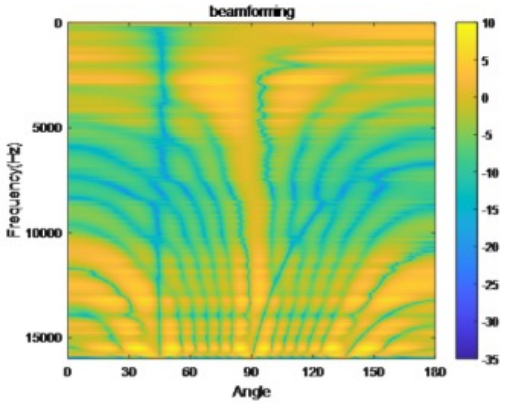
- **Delay and Sum beamforming**



$$\text{Output} = \sum_i \left(Mic_i(t + i\tau) \right)$$

- Accurate estimation of time-of-arrival is challenging in reverberant environments.
- Not robust against diffuse noise

- **Adaptive beamforming**



Minimum Variance Distortionless Response (MVDR) beamforming

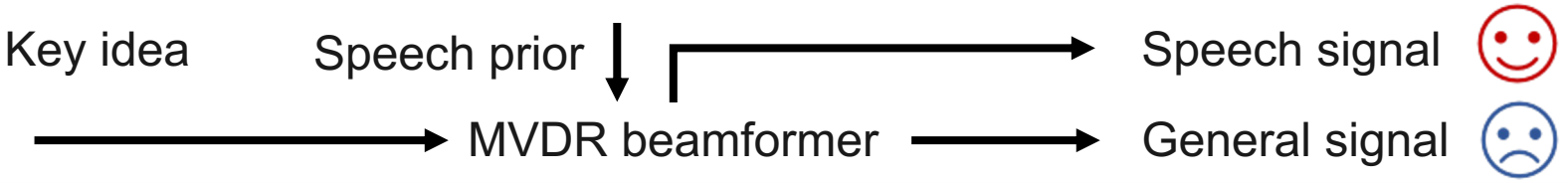
$$\boldsymbol{\omega}^* = \arg\min_{\boldsymbol{\omega}} \boldsymbol{\omega}^H \boldsymbol{R}_x \boldsymbol{\omega}$$

$$s.t. \quad \boldsymbol{\omega}^H \boldsymbol{a}(\theta_d) = 1$$

- The direction of the interference is known.

Application in speech separation

■ Hybrid strategy -- PnPMVDR



➢ The corresponding (scaled) **augmented Lagrangian function**

$$\mathcal{L}(\mathbf{w}(k),\mathbf{R},\mathbf{V},\mathbf{P}) = \frac{1}{N}\sum_{k=1}^{K}\left\|\mathbf{w}^{\mathrm{H}}(k)\mathbf{X}(k)\right\|_2^2 + \underbrace{\frac{\beta}{2}\langle \mathbf{R}, \mathbf{R}-\Omega(\mathbf{R})\rangle}_{\textbf{regularization}}$$
$$+\frac{\rho}{2}\sum_{k=1}^{K}\left(\left\|\mathbf{w}^{\mathrm{H}}(k)\boldsymbol{X}(k)-\boldsymbol{R}(k)-\boldsymbol{V}(k)+\boldsymbol{P}(k)\right\|_2^2\right) - \frac{\rho}{2}\sum_{k=1}^{K}\left(\|\boldsymbol{P}(k)\|_2^2\right) + \sum_{k=1}^{K}\boldsymbol{\lambda}(k)\left(\mathbf{w}^{\mathrm{H}}(k)\boldsymbol{a}(k)-1\right)$$

■ β: **Trade-off parameter** ■ ρ: **Penalty parameter**

➢ Using variable splitting method to **solving the problem**

◆ With respect to $\overline{\mathbf{w}}(k)$: $\mathbf{w}^{(\ell+1)}(k) = \frac{1}{N\rho+2}\boldsymbol{R}_x^{-1}(k)\left\{\rho[\boldsymbol{Z}^{(\ell)}(k)]^{\mathrm{H}} - \boldsymbol{F}^{(\ell)}(k)\right\}$

◆ With respect to $\mathbf{R}$: $\mathbf{R}^{(\ell+1,i+1)} = \mu\tilde{\mathbf{R}}^{(\ell+1)} + (1-\mu)\Omega\left(\mathbf{R}^{(\ell+1,i)}\right)$ ← *The fixed-point iteration with the denoiser prior*

◆ With respect to $\mathbf{P}$: $\boldsymbol{P}^{(\ell+1)} = \boldsymbol{P}^{(\ell)} + \widehat{\boldsymbol{S}}^{(\ell+1)} - \boldsymbol{R}^{(\ell+1)} - \boldsymbol{V}^{(\ell)}$

◆ With respect to $\mathbf{V}$: $\boldsymbol{V}^{(\ell+1)} = \widehat{\boldsymbol{S}}^{(\ell+1)} - \boldsymbol{R}^{(\ell+1)} + \boldsymbol{P}^{(\ell+1)}$

[1] C. Chang, Z. Yang, and J. Chen, “Plug-and-play mvdr beamforming for speech separation,” in Proc. 2024 IEEE Int. Conf. Acoust., Speech, Signal Process. IEEE, 2024, pp. 1346–1350.

Application in speech separation

- Deep denoiser: Frequency Recurrent Convolutional Recurrent Networks[1]
- additive noise: WGN

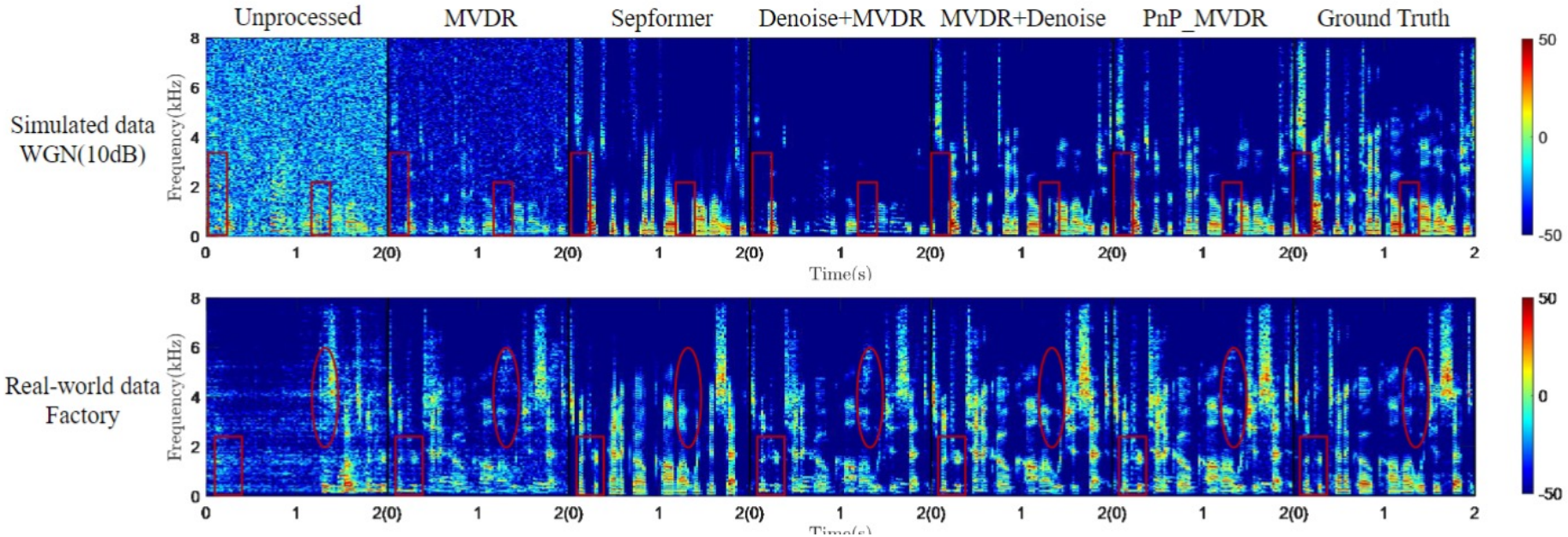
Noise level	Comparison methods	Speaker 1			Speaker 2		
		fwSNRseg (dB) ↑	PESQ ↑	STOI ↑	fwSNRseg (dB) ↑	PESQ ↑	STOI ↑
SNR = 10 dB	Unprocessed speech	5.59	1.19	0.59	6.73	1.84	0.84
	MVDR	9.62	2.17	0.94	9.22	2.33	0.94
	FRCRN+MVDR	7.06	1.95	0.81	7.21	2.13	0.89
	MVDR+FRCRN	12.17	3.23	**0.98**	11.80	3.23	**0.96**
	Proposed	**13.28**	**3.29**	**0.98**	**12.42**	**3.26**	**0.96**
		$(\rho = 95, \mu = 0.40)$			$(\rho = 95, \mu = 0.25)$		
SNR = 0 dB	Unprocessed speech	4.61	0.98	0.50	5.27	1.45	0.75
	MVDR	6.51	1.49	0.78	7.21	1.94	0.89
	FRCRN+MVDR	4.58	1.62	0.66	6.28	2.17	0.87
	MVDR+FRCRN	9.60	2.69	**0.94**	9.89	2.73	**0.95**
	Proposed	**10.92**	**2.77**	**0.94**	**11.23**	**2.81**	**0.95**
		$(\rho = 80, \mu = 0.45)$			$(\rho = 5, \mu = 0.45)$		
SNR = -10 dB	Unprocessed speech	4.41	0.96	0.41	4.65	1.00	0.56
	MVDR	4.88	0.98	0.53	5.28	1.39	0.75
	FRCRN+MVDR	2.97	1.09	0.38	3.90	1.91	0.77
	MVDR+FRCRN	4.34	1.24	**0.61**	7.92	2.27	**0.89**
	Proposed	**6.25**	**1.30**	**0.61**	**9.12**	**2.37**	**0.89**
		$(\rho = 95, \mu = 0.50)$			$(\rho = 50, \mu = 0.35)$		

Takeaway: By plugging speech priors into the MVDR framework, we can significantly enhance both speech quality and intelligibility under noisy environments

[1] S. Zhao, B. Ma,K.N.Watcharasupat, and W.-S. Gan, "FRCRN: Boosting feature representation using frequency recurrence for monaural speech enhancement," in Proc. 2022 IEEE Int. Conf. Acoust., Speech, Signal Process., May 2022, pp. 9281–9285.

Application in speech separation

- **PnPMVDR**



- **Clean**
- **Observed**
- **MVDR + Denoise**
- **PnPMVDR**

[1] C. Chang, Z. Yang, and J. Chen, “Plug-and-play mvdr beamforming for speech separation,” in Proc. 2024 IEEE Int. Conf. Acoust., Speech, Signal Process. IEEE, 2024, pp. 1346–1350.

Outline

Introduction

Part I: Methodology
- General Problem Formulation
- Hybrid Approaches for Parameter Estimation
- State-of-the-Art Hybrid Techniques

Part II: Applications to Image Processing
- Background in Hyperspectral Imaging
- Hybrid Strategies for Hyperspectral Deconvolution
- Hybrid Strategies for Hyperspectral Fusion
- Hybrid Strategies for Hyperspectral Unmixing

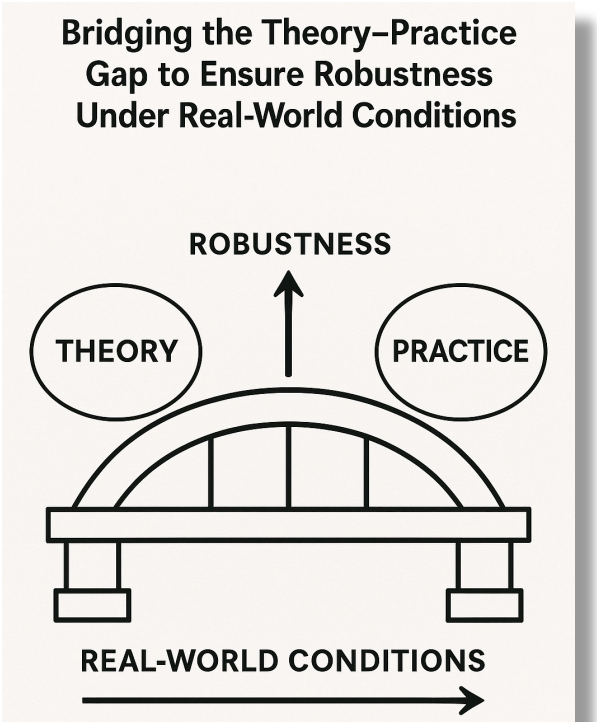
Part III: Applications to Speech Signal Processing
- Backgrond in Speech Processing
- Hybrid Strategies for Speech Dereverberation
- Hybrid Strategies for Speech Separation

Future Directions

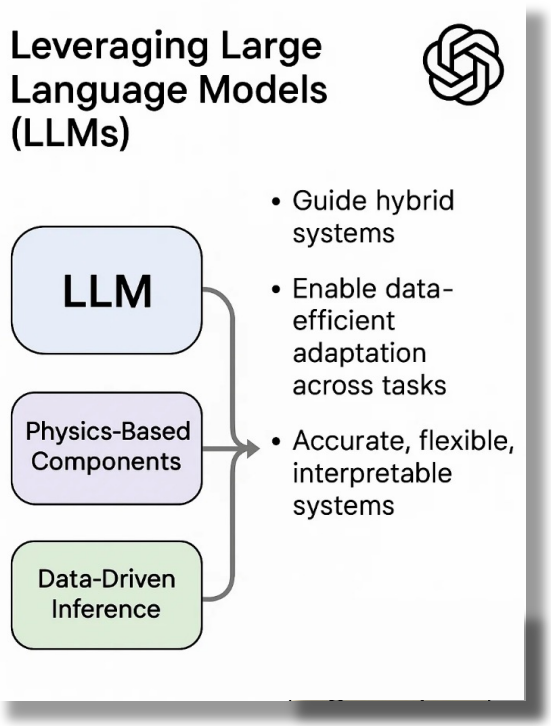
Future Directions

1. Extend hybrid methodologies to new modalities

2. Bridging the theory–practice gap



3. Leveraging large language models



Thanks!

Jie Chen* Xiuheng Wang† Ziye Yang*

*School of Artificial Intelligence,

Northwestern Polytechnical University, China.

† Université de Lorraine, CRAN, CNRS, France.