

Deep Learning Approaches for Speech Enhancement Toward Robust ASR

Xingyu Shen

**A Thesis
in
The Department
of
Electrical and Computer Engineering**

**Presented in Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy (Electrical and Computer Engineering) at
Concordia University
Montréal, Québec, Canada**

April 2026

© Xingyu Shen, 2026

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Mr. Xingyu Shen**

Entitled: **Deep Learning Approaches for Speech Enhancement Toward Robust ASR**

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Electrical and Computer Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

Dr. Sabine Bergler Chair

Dr. Sreeraman Rajan External Examiner

Dr. Arash Mohammadi Arms-Length Examiner

Dr. William Lynch Examiner

Dr. Maria Amer Examiner

Dr. Wei-Ping Zhu Supervisor

Approved by

Dr. Jun Cai, Graduate Program Director
Department of Electrical and Computer Engineering

2026

Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

Deep Learning Approaches for Speech Enhancement Toward Robust ASR

Xingyu Shen, Ph.D.

Concordia University, 2026

Speech enhancement (SE) aims to suppress noise and reverberation while preserving speech information, improving quality, intelligibility, and automatic speech recognition (ASR) robustness. In multichannel speech enhancement (MCSE), microphone arrays provide spatial diversity for filtering and dereverberation. Classical methods such as minimum variance distortionless response (MVDR) beamforming are principled and interpretable, but depend on reliable spatial statistics and array assumptions; their performance can degrade under covariance-estimation error, array perturbation, nonstationary interference, and deployment mismatch. Deep learning improves spectro-temporal modeling and learnable spatial processing, yet designing MCSE front ends that generalize across array geometries, model long context, and reliably benefit frozen ASR back ends remains difficult.

This thesis develops short-time Fourier transform (STFT)-domain deep-learning approaches for robust and deployable speech enhancement. Four contributions are presented. First, topology-robust spatial front ends represent microphones as graph nodes and learn inter-channel interactions in the complex STFT domain. The work progresses from complex-valued graph convolution to graph-attention-based convex spatial combining with real, nonnegative, sum-to-one channel weights. A covariance-free inference variant uses MVDR-inspired teacher distributions and learnable temperature scaling to reduce sensitivity to covariance estimation.

Second, physics-informed and lightweight spatial-filtering architectures improve robustness and efficiency. These include compact dynamic spatial filtering with residual spectral mapping, an end-to-end MVDR-inspired framework with physics-inspired regularization and residual refinement, and a non-learned multi-band relative contrastive loss that aligns training with perceptual band structure without increasing inference cost.

Third, a dual-path state-space framework with cross-domain interaction improves long-context modeling in noisy and reverberant conditions. It captures short- and long-range dependencies at practical computational cost while coordinating magnitude restoration with complex-spectrum refinement.

Finally, this thesis studies recognition-compatible ASR front ends under frozen recognizers. A Parallel Time–Band Mixer (PTBM) models intra-band temporal context and cross-band structure without within-block recurrence, while learned observation fusion (LOF) adaptively combines noisy and enhanced complex spectra to reduce ASR-sensitive artifacts without development-set coefficient tuning.

Experiments on simulated and real noisy and reverberant benchmarks show consistent improvements in speech quality, intelligibility, robustness to geometry variation and array perturbation, and favorable computational efficiency. With frozen recognizers, the proposed front ends also improve downstream recognition accuracy.

Acknowledgments

I would first like to acknowledge my own perseverance throughout this Ph.D. journey. There were seasons of uncertainty, exhaustion, and quiet doubt, when progress seemed slow and the destination felt far away. Yet I kept working, learning, and trying again. What once felt like a long winter has finally given way to light, not by chance, but through patience, discipline, and the decision not to give up. This dissertation is also a record of that persistence.

I would like to express my deepest gratitude to my supervisor, Prof. Wei-Ping Zhu, for his invaluable guidance, patience, and continuous support throughout my Ph.D. studies. Under his supervision, I was introduced to the field of speech enhancement and learned not only its fundamental principles, but also the rigor, discipline, and perseverance required for research. His insight, encouragement, and high standards have profoundly shaped my academic development and will continue to influence my future career.

I am also deeply grateful to Prof. Benoit Champagne of McGill University, Canada, for his guidance and support throughout my research. His advice helped me build a solid theoretical foundation in speech processing, and his careful revisions of my manuscripts greatly improved my academic writing. I am especially grateful for his emphasis on rigor and precision, which has profoundly influenced the way I think and work as a researcher.

I would also like to express my sincere gratitude to the Concordia University members of my doctoral advisory and examining committee, Prof. Arash Mohammadi, Prof. William Lynch, and Prof. Maria Amer, who followed my progress throughout my Ph.D. studies and provided valuable feedback, insightful comments, and constructive suggestions at different stages of this work. I am equally grateful to Prof. Sreeraman Rajan, the External Examiner, for his careful reading of this

dissertation and for his constructive comments and suggestions, which helped improve and refine the final version of this thesis.

I am grateful to my labmates, Runze Wang and Keyu Pan, for their generous support and encouragement in both my research and daily life over the years. I sincerely wish them every success in their future academic careers.

My heartfelt thanks go to my girlfriend, Siyao Zeng, for her love, understanding, trust, and unwavering support throughout these years. Despite the long distance between Montreal and Beijing, she remained a steady source of comfort and encouragement through countless conversations across time zones. Her patience, trust, and emotional support gave me strength during the most difficult moments, and I am deeply grateful for everything she has done for me.

Finally, I would like to express my profound gratitude to my parents for their unconditional love, endless support, and constant belief in me. They have always been my strongest source of strength and motivation. Without their sacrifices, encouragement, and trust, this journey would not have been possible.

Contents

List of Figures	xii
List of Tables	xvi
List of Abbreviations	xix
1 Introduction	1
1.1 Multichannel Speech Enhancement and Its Applications	2
1.2 Traditional Methods for Multichannel Speech Enhancement	3
1.3 Deep Learning for Multichannel Speech Enhancement	5
1.3.1 Neural Beamforming and End-to-End Spatial Filtering	5
1.3.2 Topology-Robust Spatial Reasoning with Graph and Attention Models	6
1.3.3 Long-Context Modeling Beyond RNNs	6
1.3.4 Enhancement Front Ends for Robust ASR	7
1.4 Motivations and Objectives of the Research	7
1.4.1 Motivations	8
1.4.2 Objectives	9
1.5 Organization of the Thesis	10
2 Background and Preliminaries	11
2.1 Multichannel Signal Model in the STFT Domain	11
2.2 Problem Formulation for Multichannel Speech Enhancement	14
2.2.1 Target Definitions	14

2.2.2	Model Interfaces and Spatial Cues	16
2.3	Classical Spatial Filtering	17
2.3.1	Representative Beamformers	18
2.3.2	Spatial Covariance Matrices and Stabilization	19
2.4	Structured Neural Architectures Used in This Thesis	21
2.4.1	Microphone-Graph Neural Networks and Graph Attention	21
2.4.2	State-Space Sequence Models for Scalable Long-Context Modeling	23
2.4.3	Conformer Encoders as Temporal Refinement Modules	25
2.5	Experimental Protocols	26
2.5.1	Datasets	27
2.5.2	Evaluation Metrics	30
2.5.3	Computational Complexity Measures	32
2.6	Conclusion	33
3	Graph-Based Spatial Topology Learning for Multichannel Speech Enhancement	34
3.1	Introduction	34
3.2	Complex-Valued GiGCN-TPARN for Topology-Aware Spatial Feature Fusion	36
3.2.1	Proposed Method	36
3.2.2	Experimental Setup	39
3.2.3	Experimental Results and Analysis	40
3.3	MVDR-Consistent Graph-Attention Spatial Filtering with Local Conformer	42
3.3.1	Proposed Method	42
3.3.2	Experimental Setup	46
3.3.3	Experimental Results and Analysis	48
3.4	Stable Covariance-Free Graph-Attention Spatial Filtering with Learnable Temperature	50
3.4.1	Proposed Method	51
3.4.2	Experimental Setup	60
3.4.3	Experimental Results and Analysis	61
3.5	Conclusion	67

4	Physics-Informed Lightweight Deep Beamforming with Residual Refinement and Perceptual Objectives	69
4.1	Introduction	69
4.2	Dynamic Spatial Filtering with Residual Spectral Mapping	71
4.2.1	Proposed Method	71
4.2.2	Experimental Setup	75
4.2.3	Results and Analysis	76
4.3	PhysMVNet: Physics-Informed End-to-End MVDR with Helmholtz Regularization and Three-Band Residual Spectral Mapping	78
4.3.1	Proposed Method	78
4.3.2	Experimental Setup	83
4.3.3	Results and Analysis	84
4.4	A Non-Learned Multi-Band Relative Contrastive Loss	88
4.4.1	Proposed Method	88
4.4.2	Experimental Setup	91
4.4.3	Results and Analysis	92
4.5	Conclusion	95
5	Dual-Path State-Space Modeling with Cross-Domain Interaction	97
5.1	Introduction	97
5.2	Proposed Method	98
5.2.1	Overall Architecture	98
5.2.2	Encoder	99
5.2.3	Dual-Path S5 Model	101
5.2.4	Cross-Domain Interaction Module	104
5.2.5	Decoder	106
5.2.6	Loss Function	107
5.3	Experimental Setup	108
5.3.1	Datasets	108

5.3.2	Parameter Setup and Training Strategy	109
5.3.3	Evaluation Metrics	111
5.4	Results and Analysis	111
5.4.1	Effect of Dual-Path Models	111
5.4.2	Effect of Temporal Sequence Models	112
5.4.3	Effect of Cross-Domain Interaction Modules	113
5.4.4	Effect of Feature Source for Cross-Domain Interaction	113
5.4.5	Robustness to Microphone Permutation and Geometry	114
5.4.6	Results on Seen and Unseen Speaker Test Sets	114
5.4.7	Results on CHiME-4 Dataset	116
5.4.8	Results on ConferencingSpeech 2021 Development Set	117
5.5	Conclusion	117
6	Efficient Front-End Speech Enhancement for Robust ASR via Parallel Time–Band Mixing and Learned Observation Fusion	119
6.1	Introduction	119
6.2	Proposed Method	121
6.2.1	System Overview	121
6.2.2	Band-Split Tokenizer	121
6.2.3	Parallel Time–Band Mixer Backbone	122
6.2.4	Learned Observation Fusion for Recognition Stability	126
6.2.5	Frozen Recognizer Interface and Learning Objective	127
6.2.6	Complexity Analysis	128
6.3	Experimental Setup	128
6.3.1	Datasets and Evaluation Protocols	128
6.3.2	Parameter Setup and Training Strategy	131
6.3.3	Baselines and Back-End Recognizers	133
6.4	Results and Analysis	135
6.4.1	DNS Challenge: Efficiency Under Reverberation	136

6.4.2	CHiME-4 Real 1-Channel: Generalization to Real Recordings	137
6.4.3	Ablations on Band Interaction and Fusion Granularity	137
6.4.4	CHiME-4 Real 1-Channel: Cross-Back-End Transfer	139
6.4.5	Cross-Lingual Real-World Mandarin: Portability Beyond English	139
6.4.6	Direct Comparison to Bridging Modules on CHiME-4 (Channel-5 Protocol)	140
6.4.7	SNR-Stratified Analysis on LibriSpeech + DNS	141
6.4.8	Out-of-Domain Stress Tests: Aurora-4 and VoiceBank-DEMAND	142
6.4.9	Mechanism Analysis of LOF	143
6.4.10	Summary of Findings	144
6.5	Conclusion	144
7	Conclusion and Future Work	146
7.1	Concluding Remarks	146
7.2	Future Work	149
	Appendix A Publications from the Thesis Research	151
	Bibliography	153

List of Figures

Figure 2.1	Multi-layer GCN schematic	22
Figure 2.2	Graph attention and multi-head neighborhood aggregation	24
Figure 2.3	Representative encoder–graph–decoder pipeline for graph-based multichannel speech enhancement. “ M ” denotes tensors that retain an explicit microphone/node axis. The graph block performs spatial fusion in a compact latent space; skip connections support U-Net-style reconstruction. A channel-1 spectrum is routed in parallel and combined at the output via complex multiplication (masking interface, eq. (19)).	25
Figure 2.4	State-space modeling viewpoints for long-context sequence modeling	26
Figure 2.5	Conformer encoder overview and Conformer block	27
Figure 3.1	Overall architecture of GiGCN–TPARN. The top row shows the complete enhancement pipeline, and the lower row expands the complex-valued GiGCN and decoupled-LSTM components. The TPARN block is shown schematically as the attention-based frequency–time–space fusion stage; its stream-fusion interface is defined in Eqs. (54)–(55).	37
Figure 3.2	Overall pipeline of the proposed graph-attention spatial filter with a local Conformer and complex-mask refinement. The graph-attention spatial combiner corresponds to Eq. (62), and the cRM-based refinement corresponds to Eq. (68). In the source figure, P denotes the microphone count ($P = M$ in the text), and the label $h_r(f, t)$ corresponds to the reference-node embedding $\mathbf{h}_r^{(L_s)}(t, f)$	42

Figure 3.3 Overall architecture of the stabilized covariance-free spatial filtering system.

At inference, the spatial front end performs covariance-free convex spatial combining, producing the reference stream $Y_r(t, f)$ in Eq. (69); SCM-based teacher construction and alignment are used during training only. In the source figure, P denotes the microphone count ($P = M$ in the text), the reference-row attention corresponds to the final-layer weights $a_{rj}(t, f)$, and the stabilized convex-combining weights are denoted by $\tilde{a}_{rj}(t, f)$ in the text.

52

Figure 3.4 Spatial front end with stability-oriented modules. Dual-projection graph attention produces convex channel weights. In the source figure, P denotes the microphone count, the reference-row attention corresponds to the final-layer weights $a_{rj}(t, f)$, and the stabilized convex-combining weights are written in the text as $\tilde{a}_{rj}(t, f)$. The scorer block implicitly contains the internal maps $\mathbf{U}_q^{(\ell)}$ and $\mathbf{U}_k^{(\ell)}$ from Eq. (73); these are internal query/key interaction projections rather than additional graph layers. A learnable temperature stabilizes the final-layer distribution, low-band cross-frequency coupling reduces low-frequency attention jitter, and the same temperature modulates scaled diagonal loading in the training-only teacher construction.

53

Figure 4.1 Overall architecture of DSF-RSM. The DSF module generates TF-varying convex spatial weights for a filter-and-sum spatial front end; the beamformed intermediate spectrum corresponds to $Y_{\text{beam}}(t, f)$ in Eq. (101). The RSM module predicts an additive residual in the complex (real/imaginary) domain, and the final refined spectrum corresponds to $\hat{S}(t, f)$ formed by Eqs. (106)–(107).

71

Figure 4.2	Overview of the proposed PhysMVNet architecture. (a) End-to-end pipeline from STFT through SFG, learnable beamforming, training-time Helmholtz regularization, and RSM to iSTFT; (b) Spatial Filter Generation (SFG) block; (c) learnable beamforming block; (d) Residual Spectral Mapping (RSM) block. The beamformed intermediate spectrum shown before the regularization and RSM branches corresponds to $Y_{\text{beam}}(t, f)$ in Eq. (117), the final refined output corresponds to $\hat{S}(t, f)$ in Eq. (123), and the Helmholtz branch is a training-only branch computed from Y_{beam} using the fixed stencil L and scale k_f in Eq. (118).	79
Figure 5.1	Overall architecture of the proposed dual-path model for multichannel speech enhancement. To preserve the original diagram, the schematic input label Y corresponds to the real-valued multichannel input representation $\mathbf{X}_{\text{RI}}$ in the text. The internal branch labels $\mathbf{F}_{\text{CME}}$ and $\mathbf{F}_{\text{CSR}}$ denote the CME and CSR branch features before cross-domain interaction, and $\mathbf{F}_{\text{CME}}^{\text{out}}$ and $\mathbf{F}_{\text{CSR}}^{\text{out}}$ denote the post-interaction features passed to the decoders.	98
Figure 5.2	Structure of the S3Conv-based encoder. The encoder input corresponds to $\mathbf{X}_{\text{RI}}$, and the encoder output corresponds to $\mathbf{F}_{\text{enc}}$ obtained after the spatial/TF fusion and instance-normalization steps in Eqs. (138)–(143).	100
Figure 5.3	Overview of the S5 module, which captures both short-term and long-term dependencies. The labels $\mathbf{u}_k$, $\mathbf{y}_k$, and $\mathbf{z}_k$ denote the flattened per-frame input vector, the S5 output before the lightweight refinement path, and the post-refinement block output, respectively, consistent with Eqs. (148)–(149). The schematic label CRelu corresponds to the generic pointwise nonlinearity $\sigma(\cdot)$ used in the text.	102
Figure 5.4	Diagram of the cross-domain interaction module. The two inputs correspond to the first-stage CME and CSR branch features. In the text, the projected single-stream interaction features are denoted by $\mathbf{F}_{\text{CME}}$ and $\mathbf{F}_{\text{CSR}}$ in Eq. (150), and the post-interaction outputs are denoted by $\mathbf{F}_{\text{CME}}^{\text{out}}$ and $\mathbf{F}_{\text{CSR}}^{\text{out}}$ in Eq. (154).	104

Figure 6.1 Overall architecture of the proposed PTBM–LOF front end under a frozen ASR back end. The schematic mask label M in the source figure corresponds to the predicted complex mask $\widehat{M}$ in the text. The enhanced spectrum before learned observation fusion is denoted by $\widehat{S}$, and the fused spectrum passed to the recognizer is denoted by Y , consistent with Eqs. (174) and (176). Solid boxes and arrows denote the inference path from the noisy waveform to the fused waveform passed to the recognizer. Dashed boxes or arrows denote training-only components, including spectral supervision on $\widehat{S}$ and Y and an auxiliary CTC consistency loss using a frozen CTC teacher. Only the front-end parameters are updated during training; the ASR back end and the CTC teacher remain frozen. 122

Figure 6.2 One PTBM block. After pre-normalization, temporal mixing and per-frame cross-band mixing run in parallel, corresponding to Eqs. (166) and (169), respectively. The scalar mixing gate is $g^{(\ell)}$ in Eq. (171), followed by the pre-normalized residual FFN in Eq. (173). 123

List of Tables

Table 2.1	Representative dataset families used for speech enhancement in this thesis. Chapter-specific training or split details are given in the corresponding experimental chapters.	30
Table 2.2	Main dataset families used for ASR front-end evaluation in this thesis.	31
Table 3.1	Ablation experiments and performance comparison on the CHiME-3 dataset.	41
Table 3.2	Ablation experiments and performance comparison on the L3DAS22 dataset.	41
Table 3.3	Experimental results on spatialized WSJCAM0 and DNS Challenge 2020 corpus.	48
Table 3.4	Ablation on spatialized WSJCAM0. The default pipeline is S–T–M; Proposed denotes the full model.	49
Table 3.5	Performance comparison on spatialized WSJCAM0 and DNS-2020.	62
Table 3.6	WER (%) comparison on LibriCSS across overlap ratios.	63
Table 3.7	Ablation studies on WSJCAM0 (4-ch).	64
Table 3.8	Sensitivity to training-time DCA and PSC regularizers.	65
Table 3.9	Effect of attention window w and convolution kernel k	66
Table 3.10	SI-SDR (dB) under geometry perturbations and channel failures.	66
Table 3.11	Capacity sweep on WSJCAM0 (4-ch).	66
Table 4.1	Comparative results of DSF-RSM on 6-channel simulated CHiME-3/4.	77
Table 4.2	Computational complexity and model size of DSF-RSM.	77
Table 4.3	Comparative results of PhysMVNet on 6-channel simulated CHiME-3/4.	85
Table 4.4	Computational complexity and model size of PhysMVNet.	86

Table 4.5	Ablation study on simulated CHiME-3/4.	86
Table 4.6	Results under extremely low SNR (-10 dB).	87
Table 4.7	Results under 2 cm microphone-array perturbation.	87
Table 4.8	Results on VoiceBank-DEMAND.	93
Table 4.9	PESQ / STOI scores on the TIMIT test set (BLSTM backbone).	93
Table 4.10	Ablation on MBRCL components (VoiceBank-DEMAND, BLSTM).	94
Table 5.1	Symbol definitions and values (Chapter 5).	110
Table 5.2	Effect of dual-path configurations.	111
Table 5.3	Comparison of temporal sequence models in performance and computational complexity.	112
Table 5.4	Effect of cross-domain interaction modules.	113
Table 5.5	Ablation on feature source for the interaction module.	113
Table 5.6	Robustness to channel permutation and geometry.	114
Table 5.7	Results on the seen-speaker test set (TIMIT + NOISEX-92 with simulated RIRs).	115
Table 5.8	Results on the unseen-speaker test set (TIMIT + NOISEX-92).	115
Table 5.9	DNSMOS on the unseen-speaker test set (TIMIT + NOISEX-92).	115
Table 5.10	Comparative results on 6-channel simulated CHiME-4.	116
Table 5.11	DNSMOS and ASR results on the CHiME-4 real test set.	117
Table 5.12	Results on the ConferencingSpeech 2021 development set.	117
Table 6.1	Leading-order MAC complexity per second (front end only).	128
Table 6.2	Summary of datasets and evaluation protocols.	130
Table 6.3	Hyperparameter settings for the proposed system.	134
Table 6.4	DNS Challenge test results with frozen Whisper back ends. MACs (GMAC/s) are for the front end only.	136
Table 6.5	CHiME-4 1-channel real-recording results with frozen Whisper back ends (WER, %).	137
Table 6.6	Additional ablations on CHiME-4 real recordings (Whisper-Medium).	138
Table 6.7	Cross-back-end evaluation on the CHiME-4 real 1-channel track.	139

Table 6.8	In-house Mandarin per-set CER breakdown	140
Table 6.9	CHiME-4 Channel-5 protocol comparison against bridging-module systems (WER, %). Lower is better.	140
Table 6.10	LibriSpeech test-clean mixed with DNS noises at predefined SNR bins . . .	141
Table 6.11	Performance comparison on out-of-domain stress tests	142

List of Abbreviations

ASR Automatic speech recognition

Aux-CTC Auxiliary connectionist temporal classification consistency objective (training only)

BAK Background-noise quality score

BLSTM Bidirectional long short-term memory

BM Bridging module

BN Batch normalization

BSRNN Band-split recurrent neural network

CBA Cross-band attention mixer

CER Character error rate

CFC Cross-frequency coupling

CME Coarse magnitude estimation

CNN Convolutional neural network

COLA Constant overlap-add

cRM Complex ratio mask

CSR Complex spectral refinement

CTC Connectionist temporal classification

CTF Convolutional transfer function

DCA Directional consistency alignment

DFT Discrete Fourier transform

DNN Deep neural network

DNS Deep Noise Suppression

DNSMOS Deep Noise Suppression Mean Opinion Score

DNSMOS-OVRL Deep Noise Suppression Mean Opinion Score overall quality score

DS Delay-and-sum

DSF Dynamic spatial filtering

DSF-RSM Dynamic Spatial Filtering with Residual Spectral Mapping

DWConv Depthwise convolution

EMA Exponential moving average

ESTOI Extended Short-Time Objective Intelligibility

FFN Feed-forward network

FFT Fast Fourier transform

FLOPs Floating-point operations

FOA First-order Ambisonics

FRCRN Frequency recurrent convolutional recurrent network

GAT Graph attention network

GATv2 Graph attention network v2

GCN Graph convolutional network

GELU Gaussian error linear unit

GEV Generalized eigenvalue beamformer

GiGCN Graph-in-Graph Convolutional Network

GMAC/s Giga multiply–accumulate operations per second

GNN Graph neural network

GR Grouped recurrent neural network

GRU Gated recurrent unit

GTCRN Grouped temporal convolutional recurrent network

IPD Inter-channel phase difference

iSTFT Inverse short-time Fourier transform

KL Kullback–Leibler divergence

LN Layer normalization

LOF Learned observation fusion

LSTM Long short-term memory

LTl Linear time-invariant

LWR Layer-wise resampling

MAC Multiply–accumulate

MBRCL Multi-Band Relative Contrastive Loss

MCSE Multichannel speech enhancement

MHA Multi-head attention

MIMO Multiple-input multiple-output

MLP Multilayer perceptron

MMSE Minimum mean-square error

MOS Mean opinion score

MRSTFT Multi-resolution short-time Fourier transform

MSE Mean squared error

MSR Margin satisfaction rate

MTF Multiplicative transfer function

MVDR Minimum variance distortionless response

MWF Multichannel Wiener filter

NB-PESQ Narrow-band Perceptual Evaluation of Speech Quality

nGED Normalized gradient-envelope deviation

OVRL Overall quality score

PCM Phase-constrained magnitude loss

PESQ Perceptual Evaluation of Speech Quality

PMSQE Perceptual Metric for Speech Quality Evaluation

PPS Pre-downsampling and post-upsampling

PSC Projection consistency loss

PTBM Parallel Time–Band Mixer

PReLU Parametric rectified linear unit

PWConv Pointwise convolution

Q-Net Quality-Net

ReLU Rectified linear unit

RIR Room impulse response

RNN Recurrent neural network

RNN-T Recurrent neural network transducer

RSM Residual spectral mapping

RT60 Reverberation time for a 60-dB decay

RTF Relative transfer function

S3Conv Spatial–spectral convolution module

S4 Structured state-space model

S4D Diagonal structured state-space model

S5 Simplified State Space for Sequence Modeling

SBP Sub-band pruning

SCM Spatial covariance matrix

SDR Signal-to-distortion ratio

SE Speech enhancement

SFG Spatial filter generation

SI-SDR Scale-invariant signal-to-distortion ratio

SI-SDR_i SI-SDR improvement

SI-SNR Scale-invariant signal-to-noise ratio

SIG Signal quality score

SISO Single-input single-output

SNR Signal-to-noise ratio

SSL Self-supervised learning

SSL-Feat Self-supervised-learning feature-space objective

SSM State-space sequence model

STFT Short-time Fourier transform

STOI Short-Time Objective Intelligibility

TCM Temporal ConvMixer

TF Time–frequency

TPARN Triple-Path Attentive Recurrent Network

VAD Voice activity detection

WB-PESQ Wide-band Perceptual Evaluation of Speech Quality

WER Word error rate

WPE Weighted prediction error

Chapter 1

Introduction

Reliable speech communication and acquisition in real acoustic environments remain challenging because recorded speech is often degraded by background noise, reverberation, and competing speakers. Speech enhancement (SE) aims to suppress such degradations while preserving linguistic content and speaker characteristics, thereby improving speech quality and intelligibility. In multichannel speech enhancement (MCSE), multiple microphones capture correlated observations of the same acoustic scene, making it possible to exploit inter-microphone phase, level, and coherence cues for spatial filtering and dereverberation. Compared with single-channel enhancement, this spatial diversity gives MCSE a clear advantage in far-field communication and noise-robust automatic speech recognition (ASR) applications [1, 2].

Classical beamforming and dereverberation provide principled and interpretable solutions for MCSE, but their performance can degrade when spatial statistics are inaccurate, acoustic scenes are dynamic, or array geometry differs from the assumed configuration [1, 3]. Deep learning has greatly expanded the design space of MCSE through data-driven spectro-temporal modeling and learnable spatial modules [4–7]. The central question addressed in this thesis is therefore not only how to improve enhancement performance, but also how to design speech-enhancement front ends that remain robust, efficient, and reliable under realistic deployment constraints and downstream task requirements.

Accordingly, this thesis focuses on robust and deployable deep-learning-based speech enhancement front ends under practical acoustic, array, and computational constraints, with particular emphasis on multichannel speech enhancement and its role in robust ASR. Its main technical developments involve topology-robust spatial reasoning, explicit and lightweight spatial filtering, scalable long-context modeling, and recognition-compatible front-end design under frozen ASR back ends.

Concretely, this thesis addresses four deployment-facing challenges that frequently limit robustness and reproducibility. First, real devices and ad hoc capture often involve varying microphone geometries and channel counts, motivating topology-aware spatial reasoning rather than implicit channel fusion. Second, many beamforming pipelines rely on covariance estimation and matrix inversion, which can be brittle under nonstationarity, limited sample support, and array perturbations. Third, nonstationary interference and reverberation tails require scalable long-context modeling beyond short-context recurrent patterns. Finally, when an enhancement module is placed before a frozen ASR recognizer, reducing noise alone is not sufficient since the enhanced signal may contain artifacts or may remove weak cues that the recognizer relies on. This motivates recognition-compatible front-end design and evaluation, rather than optimization based only on signal-level metrics.

1.1 Multichannel Speech Enhancement and Its Applications

Typical MCSE application settings include teleconferencing, meeting transcription, smart-home voice interaction, in-cabin speech interfaces, and assistive listening. These settings impose practical requirements that strongly shape front-end design. In teleconferencing and meeting transcription, spatial processing improves clarity for remote participants and provides cleaner input for diarization and ASR, especially under far-field capture and overlapping speech. In smart-home devices and conversational assistants, MCSE supports reliable wake-word detection and command recognition under household noise, competing loudspeakers, and distance variations. In automotive and in-cabin voice interfaces, rapidly changing noise conditions, directional interferers, and engine or road noise make spatial processing important for consistent user experience. In assistive listening devices and hearing aids, spatial filtering enhances a target talker and reduces listening effort in noisy social

environments, where perceptual artifacts and algorithmic latency are tightly constrained.

Although these application settings are diverse, they share several practical operating conditions. Acoustic environments are often highly variable, with nonstationary noise, competing talkers, transient events, and reverberation that changes with room conditions and user movement. Microphone geometry may also vary across devices or sessions, especially in ad hoc arrays and multi-device capture, which challenges methods that assume a fixed array layout. In addition, deployment imposes constraints on computational cost, memory footprint, and latency, motivating efficient architectures with controlled complexity and robust behavior under acoustic and array mismatch.

In practice, MCSE is often part of a larger speech-processing pipeline rather than an isolated signal-processing module. A particularly important case is its use as a front end for downstream tasks such as diarization and ASR. In such settings, enhancement quality cannot be judged solely by signal-level metrics, because improvements in perceptual or reconstruction scores do not always translate into lower recognition error, especially when the ASR back end is frozen and cannot adapt to the distribution shift induced by front-end enhancement [8,9]. These application and deployment considerations motivate the following review of traditional and learning-based MCSE methods, and they provide the basis for the research motivations and objectives stated later in Section 1.4.

1.2 Traditional Methods for Multichannel Speech Enhancement

To place the above application and deployment issues in context, we briefly review traditional MCSE methods. Rooted in array signal processing and statistical filtering, these methods derive spatial filters from explicit assumptions on acoustic propagation, spatial covariance structure, and interference models. Although deep learning has greatly expanded the design space of MCSE, classical methods remain important as interpretable references, strong baselines, and sources of inductive bias for modern learning-based systems.

Most conventional multichannel enhancement pipelines operate in the STFT domain under a narrowband approximation. A typical system performs time–frequency analysis, applies spatial filtering to exploit inter-microphone structure, and may add post-filtering or spectral shaping. Linear beamforming forms an enhanced output by linearly combining microphone signals with complex

weights. Fixed beamformers such as delay-and-sum rely on array geometry and target direction-of-arrival information. They are simple and numerically stable, but their spatial selectivity is limited when interferers are close to the target or when reverberation dominates [2, 10]. Superdirective and related designs improve directivity by exploiting spatial coherence models, but can become sensitive to microphone mismatch and diffuse-noise modeling errors [2].

Adaptive beamformers further exploit spatial statistics to improve interference suppression. Representative criteria include minimum variance distortionless response (MVDR) beamforming, generalized eigenvalue (GEV) beamforming, and multichannel Wiener filtering (MWF) [3, 10]. These methods depend on spatial covariance matrices of speech and interference components, whose estimation becomes difficult under limited sample support, nonstationary scenes, and competing speakers. Practical systems therefore often rely on stabilization techniques such as recursive averaging, speech-presence weighting, and diagonal loading. When covariance estimates are inaccurate or ill-conditioned, adaptive beamformers may suffer from target distortion, residual interference, or unstable behavior, especially under array perturbations and geometry mismatch.

Reverberation introduces an additional challenge because late reflections overlap with speech and behave like structured interference. Weighted prediction error (WPE) dereverberation reduces late reverberation by multichannel linear prediction in the STFT domain and is often combined with beamforming to produce dereverberated and spatially filtered outputs [11, 12]. Such modular pipelines are attractive for interpretability and engineering control, but they also expose multiple sensitivity points, including dereverberation parameter choices and the reliability of spatial-statistics estimation in dynamic acoustic conditions.

Despite these limitations, classical multichannel methods retain two enduring values for modern MCSE. First, they make spatial assumptions and constraints explicit, for example through distortionless response and minimum-variance suppression. Second, they provide reusable computational primitives, such as covariance estimation, steering proxies, and linear spatial filtering, that can be embedded into learning-based systems to improve stability and interpretability. These principles continue to motivate the neural approaches reviewed next.

1.3 Deep Learning for Multichannel Speech Enhancement

Building on these classical formulations, deep learning has expanded MCSE from hand-crafted spatial filtering pipelines to trainable systems that jointly model spatial, spectral, and temporal structure. Compared with purely model-based methods, neural MCSE systems can learn complex interference patterns and nonstationary dynamics and can be optimized end-to-end under task-relevant objectives. This shift, however, introduces new challenges, including reliable exploitation of inter-microphone coherence, robust generalization across array geometries, and control of model complexity for deployment.

1.3.1 Neural Beamforming and End-to-End Spatial Filtering

A mainstream direction in neural MCSE is to integrate classical beamforming structure with learned components. A widely adopted paradigm estimates time–frequency masks using neural networks and then uses these masks to compute spatial covariance matrices that parameterize MVDR, GEV, or MWF-style beamformers [13–15]. This “mask $\rightarrow$ covariance $\rightarrow$ beamformer” formulation retains inductive bias and interpretability while leveraging data-driven mask prediction. Subsequent work has improved robustness by stabilizing covariance estimation, addressing numerical issues in matrix inversion, and designing training strategies that account for the interaction between mask errors and spatial filtering [5, 8, 16].

More recent work increasingly favors end-to-end formulations that learn spatial filtering behavior more directly. Differentiable beamforming layers allow gradients to flow through spatial filter design, enabling joint optimization of feature extraction, spatial filtering, and reconstruction [17]. Time-domain filter-and-sum networks provide another end-to-end alternative by learning adaptive multichannel filters directly on waveforms, which can be attractive for bounded-latency settings [18]. In the STFT domain, direct regression of complex beamforming weights has also been explored, offering flexible spatial filtering at the cost of requiring careful architectural and regularization choices to ensure generalization [19, 20]. These developments motivate MCSE architectures that combine spatial structure, trainable refinement, and deployability under practical constraints [6, 7].

1.3.2 Topology-Robust Spatial Reasoning with Graph and Attention Models

A core difficulty in practical multichannel learning is that microphone configurations and geometries can vary across devices and recording setups. Methods that implicitly assume a fixed channel ordering or a fixed array geometry may degrade when microphones are permuted, missing, or distributed, which is common in ad hoc arrays and multi-device capture. Graph neural networks (GNNs) provide a principled representation by modeling microphones as nodes and inter-channel relations as edges, enabling topology-aware feature aggregation that is less tied to a specific geometry [21]. In this setting, spatial reasoning becomes message passing on a microphone graph, and learned aggregation can capture inter-channel coherence patterns that are informative for enhancement.

Graph attention further improves flexibility by allowing adaptive weighting of inter-microphone interactions. Recent work has applied graph-attention-based designs to multichannel enhancement and dereverberation, particularly under ad hoc arrays where spatial relations may be uncertain or time-varying [22, 23]. Transformer-style frequency–time attention has also been shown to be effective for multichannel enhancement, and efficient attentive architectures have been proposed to improve performance under constrained complexity [24, 25]. These trends motivate topology-robust and interpretable spatial front ends for MCSE.

1.3.3 Long-Context Modeling Beyond RNNs

While spatial processing is essential in MCSE, output quality is also shaped by temporal modeling. Nonstationary interference and reverberation tails exhibit long-range dependencies, making long-context modeling crucial for suppressing residual interference and controlling artifacts. Dual-path processing captures long context efficiently by alternating modeling along different axes or segments, thereby reducing the effective sequence length per module [26]. Transformer and Conformer families further improve long-context modeling via attention combined with local convolutional inductive bias [27].

State-space models (SSMs) have emerged as a compelling alternative to attention and recurrence for long-range sequence modeling. Modern SSM formulations such as S4D and S5 offer efficient

long-context modeling with favorable scaling behavior, and they have rapidly expanded into speech and audio applications [28, 29]. These developments motivate scalable long-context backbones for multichannel enhancement.

1.3.4 Enhancement Front Ends for Robust ASR

Beyond signal-level evaluation, a growing body of work considers enhancement front ends from the viewpoint of downstream ASR robustness. Under frozen back ends, enhancement may reduce noise yet still harm recognition by removing weak but recognizer-relevant cues or by introducing artifacts to which the recognizer is sensitive [8, 30]. Recent studies have therefore revisited front-end design specifically for ASR robustness, including enhancement adaptation under fixed back ends and structured synthesis or blending mechanisms that reduce ASR-sensitive artifacts [30–32]. Efficiency and parallelizability are also central for deployable ASR front ends. Band-split enhancement architectures provide a strong low-distortion backbone, and structured time–band modeling has been explored to reduce serial dependencies and improve throughput [33, 34]. These trends motivate recognition-compatible front-end design for robust ASR.

Taken together, the methods reviewed in this section show that strong enhancement performance alone is not sufficient for deployment. Practical front ends must also remain robust to topology variation, spatial-statistics errors, long-context demands, and frozen-ASR constraints. These issues motivate the research objectives stated next.

1.4 Motivations and Objectives of the Research

This section distills the main design tensions identified above and states the research objectives that guide the developments in the remainder of the thesis.

1.4.1 Motivations

Physical interpretability vs. robustness under acoustic and array mismatch: Classical array-processing methods provide interpretable spatial filtering rules under explicit statistical assumptions, but they can be sensitive to covariance-estimation error, array perturbations, and rapidly varying acoustic scenes. Deep learning improves modeling flexibility, but purely data-driven designs may weaken physical interpretability. This tension motivates MCSE methods that preserve useful spatial structure while reducing brittleness to acoustic, statistical, and array mismatch.

Fixed array assumptions vs. topology variation: Many multichannel front ends implicitly assume fixed channel ordering and fixed array geometry. In practice, microphone layouts, channel counts, and ordering may vary across devices or sessions. This motivates topology-aware spatial reasoning that represents inter-microphone relations explicitly and generalizes more robustly across array conditions.

Covariance-based spatial filtering vs. stable inference: Many multichannel pipelines depend on covariance estimation and matrix inversion, which can be brittle under finite-sample support, ill-conditioning, and nonstationary scenes. This motivates learning-based spatial front ends that remain physically grounded while reducing reliance on fragile inference steps.

Short-context models vs. long-range temporal structure: Nonstationary interference and reverberation tails span long time ranges. Capturing these dependencies is important for suppressing residual interference and controlling artifacts without distorting speech. This motivates sequence models with stronger long-context capability and practical computational scaling.

Signal-level quality vs. downstream ASR robustness: When multichannel enhancement is used as a front end for ASR, improving signal-level metrics does not guarantee improved recognition, especially under frozen recognizers. Enhancement can change the acoustic distribution seen by the recognizer by introducing artifacts or removing weak recognizer-relevant cues. This motivates recognition-compatible enhancement designs that explicitly control artifacts and preserve information needed by downstream ASR.

1.4.2 Objectives

The overarching goal of this research is to develop robust, deployable, and interpretable multichannel speech enhancement front ends under practical acoustic, array, and computational constraints, while also enabling reliable gains for downstream ASR under frozen recognizers. Accordingly, this thesis pursues four objectives:

- **Topology-robust spatial reasoning on microphone graphs.** Develop topology-aware spatial front ends by modeling microphones as graph nodes and learning inter-channel interactions in the complex STFT domain, progressing from complex-valued graph convolution to graph-attention-based convex spatial combining with real, nonnegative, sum-to-one weights. Further reduce inference-time dependence on covariance estimation through covariance-free spatial inference guided by MVDR-inspired teacher distributions and stabilized by learnable temperature control. Validate robustness under topology variation, array-geometry changes, and array perturbations.
- **Physics-informed and lightweight spatial filtering with perceptually motivated objectives.** Design deployable spatial-filtering architectures that improve robustness and efficiency, including compact dynamic spatial filtering paired with residual spectral mapping, and end-to-end MVDR-style learning with physics-inspired regularization for stability under acoustic mismatch, array-geometry perturbation, and spatial-statistics errors. Develop an analytic, non-learned, architecture-agnostic training objective to better align optimization with perceptual structure and artifact control without increasing inference cost.
- **Scalable long-context modeling via state-space sequence backbones.** Enable long-context modeling at practical computational cost by developing dual-path state-space backbones and cross-domain interaction mechanisms that coordinate magnitude-oriented estimation with complex-domain refinement. Provide controlled ablations and complexity analyses to isolate the benefits of the state-space parameterization, dual-path processing, and cross-domain interaction.
- **Recognition-compatible front-end design for robust ASR under frozen back ends.** Improve recognition robustness by developing a parallel-friendly band-split enhancement backbone and an ASR-oriented learned observation fusion mechanism that blends noisy and enhanced spectra

directly on the complex STFT grid without development-set coefficient tuning. Evaluate recognition performance under frozen back ends across noisy, reverberant, and domain-mismatched conditions, and relate the results to artifact behavior and fusion dynamics.

1.5 Organization of the Thesis

The remainder of this thesis is organized as follows. Chapter 2 establishes the background and preliminaries used throughout the thesis, including the multichannel STFT-domain signal model and notation, problem formulations for MCSE (target definitions and model interfaces), classical spatial filtering as a physically interpretable reference, and the experimental protocols, evaluation metrics, and complexity measures adopted throughout the thesis. Chapter 3 develops topology-robust spatial front ends on microphone graphs, progressing from complex-domain graph message passing to an explicit per-time–frequency convex spatial combining interface, and then to a stabilized covariance-free inference variant with lightweight temperature control and MVDR-inspired training guidance. Chapter 4 focuses on deployable spatial filtering under acoustic and array mismatch together with training objectives: it first introduces a lightweight dynamic spatial-filtering front end with residual spectral mapping, then extends this framework to a physics-informed end-to-end spatial-filtering formulation with STFT-domain regularization and residual refinement, and finally presents an analytic non-learned perceptually motivated objective that improves training robustness without increasing inference cost. Chapter 5 develops a long-context multichannel enhancement approach based on dual-path state-space modeling and cross-domain interaction. Chapter 6 studies efficient and recognition-compatible enhancement front ends for robust ASR under frozen recognizers, centered on a parallel-friendly band-split backbone and learned observation fusion. Finally, Chapter 7 concludes the thesis and discusses future research directions.

Chapter 2

Background and Preliminaries

This chapter establishes the shared notation, signal model, and evaluation conventions used in the remainder of the thesis. We first present a multichannel short-time Fourier transform (STFT) signal model and the narrowband approximation that underlies both classical spatial filtering and learning-based front ends. We then define the enhancement targets, model interfaces, and spatial cues used throughout the thesis. Next, we summarize representative classical spatial filters and the covariance quantities needed to describe them. We also briefly review three neural architecture families that appear repeatedly in the proposed methods, namely microphone-graph networks, state-space sequence models, and Conformer-style temporal refiners. Finally, we outline the main experimental protocols, datasets, metrics, and complexity measures used in the experimental chapters.

2.1 Multichannel Signal Model in the STFT Domain

We consider a single target speaker captured by an M -channel microphone array. Let $s[\tau]$ denote the sampled clean anechoic source speech emitted by the target, $h_m[\tau]$ the room impulse response (RIR) from the source to the m -th microphone, $m \in \{1, \dots, M\}$, and $x_m[\tau]$ the observed waveform at microphone m . The observed signal is written as

$$x_m[\tau] = (s * h_m)[\tau] + v_m[\tau], \quad m = 1, \dots, M, \quad (1)$$

where $*$ denotes linear convolution, and $v_m[\tau]$ collects all disturbance components at channel m , including additive background noise, interfering sources, and, depending on the target definition, late reverberation. Here “additive” only describes how the disturbance enters the waveform and does not imply any specific statistical assumption about the noise, such as Gaussian, white, or stationary behavior, unless explicitly stated in the simulation setup.

Notation. Throughout the thesis, $(\cdot)^\top$ denotes transpose, $(\cdot)^H$ denotes Hermitian transpose, and $(\cdot)^*$ denotes complex conjugation. For a complex scalar z , $\Re\{z\}$ and $\Im\{z\}$ denote its real and imaginary parts, and $\angle(z)$ denotes its phase. We use j for the imaginary unit ($j^2 = -1$) and ϵ for a small positive constant used for numerical stability when needed.

Using an analysis segment of length L samples, hop size R samples, and an N -point DFT, and omitting an explicit window term for notational simplicity, the complex STFT of channel m at frame index t and frequency-bin index f is written as

$$X_m(t, f) = \sum_{\kappa=0}^{L-1} x_m[tR + \kappa] e^{-j\frac{2\pi}{N}f\kappa}, \quad (2)$$

where κ indexes samples within the extracted segment, t indexes frames, and $f \in \{0, 1, \dots, N-1\}$ indexes DFT bins. For real-valued waveforms, the STFT is conjugate symmetric, so unless otherwise stated we restrict processing to the nonnegative-frequency bins

$$\mathcal{F}_+ \triangleq \{0, 1, \dots, N/2\}, \quad (3)$$

where N is assumed to be even throughout this thesis. We also write $F \triangleq |\mathcal{F}_+| = N/2 + 1$ for the number of processed nonnegative-frequency bins. We denote the STFT of the clean source speech $s[\tau]$ by $S_{\text{clean}}(t, f)$ and the STFT of the disturbance $v_m[\tau]$ by $V_m(t, f)$, computed with the same STFT parameters. Here, $S_{\text{clean}}(t, f)$ denotes the source spectrum prior to speech enhancement.

Applying the STFT to (1) yields, in general, a cross-frame relation with inter-band leakage. Following standard practice in STFT-domain dereverberation and beamforming, we use a convolutive transfer function (CTF) approximation and model propagation as a short convolution over STFT

frames within each frequency bin:

$$X_m(t, f) \approx \sum_{\ell=0}^{L_h} H_m(\ell, f) S_{\text{clean}}(t - \ell, f) + V_m(t, f), \quad (4)$$

where ℓ is the frame-lag index, L_h is the maximum frame lag of the CTF, and $H_m(\ell, f) \in \mathbb{C}$ is the corresponding CTF coefficient. Thus the CTF has $L_h + 1$ taps indexed by $\ell = 0, \dots, L_h$. We introduce (4) mainly to make explicit that long room impulse responses (RIRs) induce cross-frame memory. In the later spatial-filtering developments, we primarily use the narrowband approximation below and absorb residual approximation error into the disturbance term.

For compactness, we stack the per-channel STFT coefficients at each time–frequency bin into vectors

$$\begin{aligned} \mathbf{X}(t, f) &= [X_1(t, f), \dots, X_M(t, f)]^T \in \mathbb{C}^M, \\ \mathbf{V}(t, f) &= [V_1(t, f), \dots, V_M(t, f)]^T \in \mathbb{C}^M, \end{aligned} \quad (5)$$

and define $\mathbf{H}(\ell, f) = [H_1(\ell, f), \dots, H_M(\ell, f)]^T \in \mathbb{C}^M$. Equation (4) can then be written as

$$\mathbf{X}(t, f) \approx \sum_{\ell=0}^{L_h} \mathbf{H}(\ell, f) S_{\text{clean}}(t - \ell, f) + \mathbf{V}(t, f). \quad (6)$$

Many classical and learning-based multichannel methods further adopt a narrowband multiplicative transfer-function approximation by retaining only the zero-lag term [1]:

$$\mathbf{X}(t, f) \approx \mathbf{H}(0, f) S_{\text{clean}}(t, f) + \mathbf{V}(t, f). \quad (7)$$

Equation (7) is the simplified signal model used throughout the thesis when discussing STFT-domain spatial filtering and learning-based front ends. Late reverberation, transfer-function approximation error, and other deviations from the narrowband model are treated as part of the disturbance term $\mathbf{V}(t, f)$.

In a free-field (anechoic) setting, one may approximate $h_m[\tau] \approx \delta[\tau - \tau_m]$, where $\delta[\cdot]$ is the Kronecker delta and τ_m is the relative sample delay with $\tau_1 = 0$ for channel 1. Then

$$H_m(0, f) = e^{-j\frac{2\pi}{N}f\tau_m},$$

so $\mathbf{H}(0, f)$ reduces to the familiar steering vector. For simulation-based experiments, RIRs can be generated by the image-source method implemented in `pyroomacoustics` [35].

2.2 Problem Formulation for Multichannel Speech Enhancement

Based on the multichannel STFT model in Section 2.1, the enhancement task is to estimate a desired single-channel target spectrum from multichannel observations. From this section onward, $S(t, f)$ denotes the target spectrum defined by the evaluation protocol, and $\hat{S}(t, f)$ denotes the model output. This separates the source spectrum $S_{\text{clean}}(t, f)$ introduced above from the target used for enhancement training and evaluation.

2.2.1 Target Definitions

The desired output in multichannel enhancement depends on the evaluation protocol. Unless otherwise stated, channel 1 is used whenever a single fixed channel must be selected. Three target choices are used in this thesis.

One choice, used only in synthetically generated mixtures where the clean source is known by construction, is the dry source spectrum itself:

$$S(t, f) = S_{\text{clean}}(t, f). \quad (8)$$

This means that the system is asked to recover the anechoic speech before room propagation is introduced. It is useful when the goal is to remove both additive corruption and room coloration, but it is not available for real recordings.

A more common choice is the clean speech image at one channel. For channel $m \in \{1, \dots, M\}$, define

$$S_m(t, f) \triangleq \text{STFT}\{(s * h_m)[\tau]\}(t, f). \quad (9)$$

Unless otherwise stated, this thesis uses the channel-1 image as the target, that is,

$$S(t, f) = S_1(t, f). \quad (10)$$

This choice preserves the acoustic transfer to channel 1 while suppressing additive noise and interference.

Under the same narrowband approximation as in (7), we have

$$S_m(t, f) \approx H_m(0, f) S_{\text{clean}}(t, f). \quad (11)$$

In particular,

$$S(t, f) = S_1(t, f) \approx H_1(0, f) S_{\text{clean}}(t, f). \quad (12)$$

When this target is used, it is convenient to rewrite the narrowband model in first-channel-normalized relative transfer function (RTF) form. Assuming $H_1(0, f) \neq 0$, define

$$\mathbf{d}(f) \triangleq \left[1, \frac{H_2(0, f)}{H_1(0, f)}, \dots, \frac{H_M(0, f)}{H_1(0, f)} \right]^\top \in \mathbb{C}^M, \quad d_1(f) = 1. \quad (13)$$

Substituting (12) into (7) yields

$$\mathbf{X}(t, f) \approx \mathbf{d}(f) S(t, f) + \mathbf{V}(t, f). \quad (14)$$

For later covariance definitions, we also define the corresponding multichannel target-image vector

$$\mathbf{S}(t, f) \triangleq \mathbf{d}(f) S(t, f) \in \mathbb{C}^M, \quad \mathbf{X}(t, f) \approx \mathbf{S}(t, f) + \mathbf{V}(t, f). \quad (15)$$

When dereverberation is considered, the RIR can be decomposed into early and late parts by

$$h_m^{\text{early}}[\tau] = \begin{cases} h_m[\tau], & 0 \leq \tau < N_e, \\ 0, & \text{otherwise,} \end{cases} \quad h_m^{\text{late}}[\tau] \triangleq h_m[\tau] - h_m^{\text{early}}[\tau], \quad (16)$$

where N_e is the early-window length in samples. The corresponding early target at channel 1 is

$$S(t, f) = \text{STFT}\{(s * h_1^{\text{early}})[\tau]\}(t, f). \quad (17)$$

In this case, the late-reverberation component is treated as part of the disturbance. The same first-channel-normalized model form in (14) applies after replacing the full channel-1 image and full RTF by their early-component counterparts.

Unless otherwise stated in the experimental chapters, this thesis uses the channel-1 target image when the goal is denoising with minimal image distortion, and the early target when late reverberation should be suppressed. The anechoic target is used only in synthetically generated mixtures where removal of room coloration is part of the intended evaluation.

2.2.2 Model Interfaces and Spatial Cues

A learning-based front end is represented by a parametric estimator $\mathcal{E}_\theta$ with parameters θ . At time–frequency bin (t, f) , it produces an estimate $\hat{S}(t, f)$ of the chosen target $S(t, f)$:

$$\hat{S}(t, f) = \mathcal{E}_\theta(\mathcal{X}_{t,f}), \quad (18)$$

where $\mathcal{X}_{t,f}$ denotes the multichannel STFT context available when estimating bin (t, f) . Depending on the architecture, this context may be a single time–frequency snapshot, a local chunk, a sliding window, or a full utterance in offline processing. The local M -channel snapshot at bin (t, f) is denoted by $\mathbf{X}(t, f) \in \mathbb{C}^M$. Many systems in this thesis also expose an intermediate single-channel spectrum $Z(t, f) \in \mathbb{C}$ that is subsequently refined.

One option is complex spectral mapping, which directly predicts $\hat{S}(t, f) \in \mathbb{C}$. This interface is conceptually simple, but the network must model magnitude and phase over a wide dynamic range. Another common interface is complex ratio masking, which predicts a complex mask $\widehat{M}(t, f) \in \mathbb{C}$ and applies it to a complex input $Z(t, f)$:

$$\hat{S}(t, f) = \widehat{M}(t, f) Z(t, f). \quad (19)$$

When $Z(t, f)$ is a noisy or spatially filtered spectrum, the above masking allows $Z(t, f)$ to carry the dominant spectral scale while the predicted mask focuses on relative correction. A third interface is residual spectral refinement, in which the network predicts a complex residual $\Delta(t, f) \in \mathbb{C}$ and

forms

$$\widehat{S}(t, f) = Z(t, f) + \Delta(t, f). \quad (20)$$

This interface encourages conservative correction when $Z(t, f)$ is already reliable.

In multichannel systems, the choice of $Z(t, f)$ depends on how spatial information is exploited. A general linear spatial operator maps the multichannel snapshot to a single-channel spectrum:

$$Z(t, f) = \mathbf{w}^H(t, f) \mathbf{X}(t, f), \quad (21)$$

where $\mathbf{w}(t, f) \in \mathbb{C}^M$ is a complex weight vector. A stability-oriented special case used repeatedly in this thesis predicts real, nonnegative, sum-to-one weights and performs a convex combination:

$$Z(t, f) = \sum_{m=1}^M \alpha_m(t, f) X_m(t, f), \quad \alpha_m(t, f) \geq 0, \quad \sum_{m=1}^M \alpha_m(t, f) = 1. \quad (22)$$

This yields a gain-bounded and interpretable mixing rule. Later, graph-attention coefficients are also denoted by α_{ij} ; the two uses are distinguished by their indices.

Learning-based multichannel models also depend on how spatial cues are represented. Common real-valued feature constructions include stacked real and imaginary parts, such as $[\Re\{X_m(t, f)\}, \Im\{X_m(t, f)\}]$, and magnitude features such as $\log(|X_m(t, f)|^2 + \epsilon)$. A standard phase-based cue is the inter-channel phase difference (IPD) between channel m and channel 1:

$$\text{IPD}_{m,1}(t, f) = \angle(X_m(t, f) X_1^*(t, f)). \quad (23)$$

In practice, $\sin(\text{IPD}_{m,1})$ and $\cos(\text{IPD}_{m,1})$ are often used to avoid phase wrapping. Additional cues such as inter-channel level differences or coherence-like features may also be used when appropriate.

2.3 Classical Spatial Filtering

Classical array processing remains important in this thesis for two reasons. First, several later chapters compare against beamforming baselines. Second, beamforming-inspired quantities are

reused in some learning-based designs as training guidance or auxiliary structure [1, 2, 10]. The goal of this section is to summarize the representative criteria and covariance quantities needed in this thesis.

Under the narrowband model in (14), the mixture snapshot is approximated as $\mathbf{X}(t, f) \approx \mathbf{d}(f)S(t, f) + \mathbf{V}(t, f)$. When the early target is used, the same form applies with $S(t, f)$ interpreted as the early channel-1 target image. Spatial filtering aims to preserve the target component aligned with $\mathbf{d}(f)$ while suppressing the disturbance component.

2.3.1 Representative Beamformers

Using the general spatial operator in (21), a beamformer forms a single-channel spectrum by linearly combining channels. In classical designs, the weight vector is determined from steering or RTF information together with spatial second-order statistics. For exposition, we write the beamforming vector as frequency-dependent but locally time-invariant over the interval used to estimate those statistics.

A simple data-independent baseline is delay-and-sum (DS) beamforming. Assume relative delays $\{\tau_m\}$ of the target across channels are available, with $\tau_1 = 0$ for channel 1. One convenient DS choice is

$$\mathbf{w}_{\text{DS}}(f) = \frac{1}{M} [e^{-j\frac{2\pi}{N}f\tau_1}, \dots, e^{-j\frac{2\pi}{N}f\tau_M}]^T. \quad (24)$$

Because the operator in (21) applies $\mathbf{w}^H$, the effective coefficient on channel m is $w_m^*(f)$; hence (24) compensates the phase lag associated with delay τ_m and then averages the aligned channels.

A widely used distortion-controlled design is minimum variance distortionless response (MVDR) beamforming, which minimizes output disturbance power subject to a distortionless constraint along the target direction:

$$\mathbf{w}_{\text{MVDR}}(f) = \arg \min_{\mathbf{w}(f) \in \mathbb{C}^M} \mathbf{w}^H(f) \Phi_{\mathbf{V}\mathbf{V}}(f) \mathbf{w}(f) \quad \text{s.t.} \quad \mathbf{w}^H(f) \mathbf{d}(f) = 1, \quad (25)$$

where $\Phi_{\mathbf{V}\mathbf{V}}(f)$ denotes the disturbance spatial covariance matrix (SCM), defined in Section 2.3.2.

If $\Phi_{\mathbf{V}\mathbf{V}}(f)$ is invertible, the closed-form solution is

$$\mathbf{w}_{\text{MVDR}}(f) = \frac{\Phi_{\mathbf{V}\mathbf{V}}^{-1}(f)\mathbf{d}(f)}{\mathbf{d}^H(f)\Phi_{\mathbf{V}\mathbf{V}}^{-1}(f)\mathbf{d}(f)}. \quad (26)$$

Another representative scheme is generalized eigenvalue (GEV) beamforming, which maximizes a signal-to-disturbance ratio proxy:

$$\mathbf{w}_{\text{GEV}}(f) = \arg \max_{\mathbf{w}(f) \in \mathbb{C}^M} \frac{\mathbf{w}^H(f)\Phi_{\mathbf{S}\mathbf{S}}(f)\mathbf{w}(f)}{\mathbf{w}^H(f)\Phi_{\mathbf{V}\mathbf{V}}(f)\mathbf{w}(f)}, \quad (27)$$

where $\Phi_{\mathbf{S}\mathbf{S}}(f)$ is the target-image SCM. The optimizer is the principal generalized eigenvector and is defined only up to a nonzero complex scale, so practical implementations require an explicit normalization to control output amplitude.

The multichannel Wiener filter (MWF) follows a minimum mean-square error (MMSE) criterion aimed at the desired channel-1 image. Let $\mathbf{u}_1 \in \mathbb{R}^M$ denote the unit vector whose first entry is one and whose remaining entries are zero. A common form is

$$\mathbf{w}_{\text{MWF}}(f) = \Phi_{\mathbf{X}\mathbf{X}}^{-1}(f)\Phi_{\mathbf{S}\mathbf{S}}(f)\mathbf{u}_1, \quad (28)$$

where $\Phi_{\mathbf{X}\mathbf{X}}(f)$ is the mixture SCM. Compared with MVDR, MWF may accept controlled target attenuation when this reduces mean-square error.

2.3.2 Spatial Covariance Matrices and Stabilization

The beamformers above depend on spatial covariance matrices, which encode second-order spatial statistics at each frequency bin. They are also relevant later in the thesis because covariance quality strongly affects classical beamforming baselines and several training-time beamforming-inspired guides.

For a multichannel random vector $\mathbf{Y}(t, f) \in \mathbb{C}^M$, the SCM at frequency f is defined as

$$\Phi_{\mathbf{Y}\mathbf{Y}}(f) = \mathbb{E}[\mathbf{Y}(t, f)\mathbf{Y}^H(t, f)] \in \mathbb{C}^{M \times M}. \quad (29)$$

Its diagonal entries represent per-channel power, while the off-diagonal entries encode inter-channel coherence and relative phase structure. We use $\Phi_{\mathbf{X}\mathbf{X}}(f)$, $\Phi_{\mathbf{S}\mathbf{S}}(f)$, and $\Phi_{\mathbf{V}\mathbf{V}}(f)$ for mixture, target-image, and disturbance SCMs, respectively. Here $\Phi_{\mathbf{S}\mathbf{S}}(f)$ refers to the SCM of the target-image vector defined in (15).

Since the expectation in (29) is unavailable, SCMs are estimated from finite data. Over a frame set $\mathcal{T}$, a standard sample estimate is

$$\hat{\Phi}_{\mathbf{X}\mathbf{X}}(f) = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \mathbf{X}(t, f) \mathbf{X}^H(t, f), \quad (30)$$

where $|\mathcal{T}|$ is the number of frames used for averaging. In real mixtures, target and disturbance SCMs are often approximated by time–frequency weighting with a soft speech mask $\rho(t, f) \in [0, 1]$:

$$\hat{\Phi}_{\mathbf{S}\mathbf{S}}(f) = \frac{\sum_{t \in \mathcal{T}} \rho(t, f) \mathbf{X}(t, f) \mathbf{X}^H(t, f)}{\sum_{t \in \mathcal{T}} \rho(t, f) + \epsilon}, \quad \hat{\Phi}_{\mathbf{V}\mathbf{V}}(f) = \frac{\sum_{t \in \mathcal{T}} (1 - \rho(t, f)) \mathbf{X}(t, f) \mathbf{X}^H(t, f)}{\sum_{t \in \mathcal{T}} (1 - \rho(t, f)) + \epsilon}. \quad (31)$$

When an RTF estimate is needed under the rank-one model in (14), a common proxy is obtained from the principal eigenvector of the target SCM:

$$\tilde{\mathbf{d}}(f) = \text{eig}_{\max}(\hat{\Phi}_{\mathbf{S}\mathbf{S}}(f)), \quad \mathbf{d}(f) = \frac{\tilde{\mathbf{d}}(f)}{\tilde{d}_1(f)}, \quad (32)$$

where $\text{eig}_{\max}(\cdot)$ returns the eigenvector associated with the largest eigenvalue and the normalization enforces $d_1(f) = 1$, assuming $\tilde{d}_1(f) \neq 0$. If the early target is used, the same construction applies to the corresponding early target-image SCM.

Estimated SCMs may be ill-conditioned under limited sample support, strong channel correlation, or low signal-to-noise ratio (SNR). A standard stabilization technique is diagonal loading:

$$\hat{\Phi}_{\mathbf{V}\mathbf{V}}^{\text{load}}(f) = \hat{\Phi}_{\mathbf{V}\mathbf{V}}(f) + \delta_{\text{dl}} \mathbf{I}, \quad (33)$$

where $\delta_{\text{dl}} > 0$ is a loading factor and $\mathbf{I}$ is the $M \times M$ identity matrix. In practice, Hermitian symmetrization and eigenvalue flooring are also often used before inversion. More specialized covariance estimation or dereverberation details are introduced only in the later chapters where they

are explicitly needed.

2.4 Structured Neural Architectures Used in This Thesis

This section reviews three neural architecture families that will be used in later chapters. Each family provides an inductive bias along a different axis of the enhancement problem: microphone-graph models for cross-channel spatial reasoning under topology variation, state-space sequence models for long-context temporal dynamics, and Conformer-style blocks for local-to-mid-range temporal refinement after an explicit spatial stage. The goal here is not to provide a full architectural survey, but to establish the notation and modeling intuition needed later in the thesis.

2.4.1 Microphone-Graph Neural Networks and Graph Attention

Graph neural networks (GNNs) model the microphone array as a graph and perform cross-channel fusion by message passing [36–38]. This viewpoint is useful when channel ordering, channel count, or array geometry may vary across devices or recordings. Provided that the graph structure and node features are permuted consistently, graph message passing is permutation equivariant; it is therefore better matched to topology-robust multichannel modeling than channel-index-tied concatenation.

We denote the microphone graph by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \dots, M\}$ is the node set and $\mathcal{E}$ is the edge set. At each time–frequency bin (t, f) , channel i is associated with a node feature vector $\mathbf{x}_i(t, f) \in \mathbb{R}^D$. The hidden representation at graph layer k is denoted by $\mathbf{h}_i^{(k)}(t, f) \in \mathbb{R}^{D_k}$, with $\mathbf{h}_i^{(0)}(t, f) = \mathbf{x}_i(t, f)$.

Figure 2.1 provides a conceptual illustration of graph convolution: node features are repeatedly updated by neighborhood aggregation under a fixed graph, and stacked layers expand the receptive field from local neighbors to multi-hop neighborhoods. In this thesis, the same principle is used to propagate spatial evidence across channels rather than across semantic graph nodes.

Many GNN layers can be written in the generic message-passing form

$$\mathbf{h}_i^{(k+1)} = \phi^{(k)} \left(\mathbf{h}_i^{(k)}, \sum_{j \in \mathcal{N}(i)} \psi^{(k)}(\mathbf{h}_i^{(k)}, \mathbf{h}_j^{(k)}, \mathbf{e}_{ij}) \right), \quad (34)$$

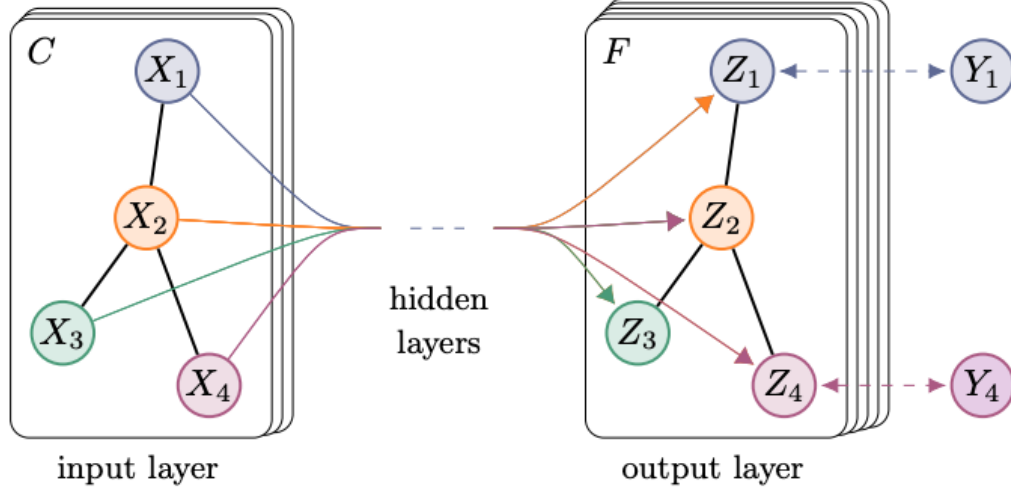


Figure 2.1: Multi-layer GCN schematic (cropped from the left panel of Fig. 1 in [37]). Node features are repeatedly updated by neighborhood aggregation under a fixed graph; stacked layers expand the receptive field to multi-hop neighborhoods. Here, X_i denotes the input feature vector of node i , Z_i an output node embedding, and Y_i an observed node label. The label C denotes the input feature dimension, while the label F denotes the number of output feature maps/classes.

where $\mathcal{N}(i) \triangleq \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}$ is the neighbor set of node i , $\psi^{(k)}(\cdot)$ computes messages, $\phi^{(k)}(\cdot)$ updates the node embedding, and $\mathbf{e}_{ij}$ denotes an optional edge feature. For notational simplicity, the dependence on (t, f) is omitted in (34)–(38); these operations are applied independently at each time–frequency bin unless stated otherwise.

A commonly used specialization is the graph convolutional network (GCN) layer [37]. With adjacency matrix $\mathbf{A}$, self-loop augmented adjacency $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, and degree matrix $\tilde{\mathbf{D}}$ whose diagonal entries satisfy $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$, the layer is

$$\mathbf{H}^{(k+1)} = \sigma\left(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(k)} \mathbf{W}^{(k)}\right), \quad (35)$$

where $\mathbf{H}^{(k)} \in \mathbb{R}^{M \times D_k}$ stacks node embeddings, $\mathbf{W}^{(k)} \in \mathbb{R}^{D_k \times D_{k+1}}$ is trainable, and $\sigma(\cdot)$ is an elementwise nonlinearity. The normalized propagation operator performs degree-normalized neighborhood averaging with self-loops. In a microphone graph, (35) therefore implements a fixed, topology-aware cross-channel mixing rule once $\mathcal{G}$ has been specified.

Graph attention replaces this fixed mixing rule with feature-dependent neighbor weights [38].

Unless otherwise stated, the attention neighborhood includes the node itself. A standard single-head attention mechanism computes

$$e_{ij} = \text{LeakyReLU}\left(\mathbf{a}^\top [\mathbf{W}\mathbf{h}_i \parallel \mathbf{W}\mathbf{h}_j]\right), \quad (36)$$

where $\mathbf{W}$ is a trainable projection, $\mathbf{a}$ is a trainable vector, and $\parallel$ denotes concatenation. Softmax over the neighborhood gives

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{r \in \mathcal{N}(i)} \exp(e_{ir})}, \quad \alpha_{ij} \geq 0, \quad \sum_{j \in \mathcal{N}(i)} \alpha_{ij} = 1, \quad (37)$$

and the node update becomes

$$\mathbf{h}'_i = \sigma \left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij} \mathbf{W}\mathbf{h}_j \right). \quad (38)$$

For each receiver node i , the coefficients $\{\alpha_{ij}\}_{j \in \mathcal{N}(i)}$ form a simplex over its neighbors, so each attention head performs a convex aggregation of transformed neighbor features. This property is particularly relevant later in the thesis because it connects graph attention to the explicit convex spatial combining interface introduced in Chapter 3.

Figure 2.2 is used to make that connection visually explicit: the left panel illustrates attention scoring and normalization, while the right panel emphasizes that each head produces simplex-constrained weights before head fusion. Figure 2.3 then shows how a microphone-graph module can be embedded into a representative enhancement front end. Shared-weight per-channel encoders preserve an explicit channel axis, the graph block performs topology-aware spatial fusion in a compact latent space, and the decoder converts the fused representation to an enhancement output such as a mask or residual correction. The figure is intended to clarify the architectural role of the graph module rather than to prescribe a single implementation.

2.4.2 State-Space Sequence Models for Scalable Long-Context Modeling

Temporal modeling is critical for enhancement under nonstationary interference and long reverberation tails. State-space sequence models (SSMs) capture long-range temporal dependence

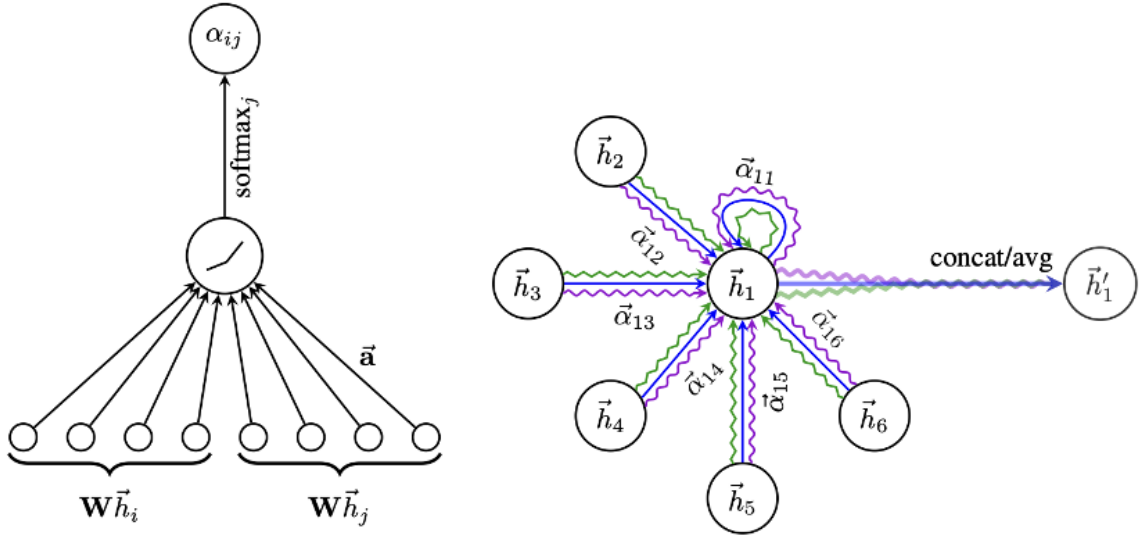


Figure 2.2: Graph attention and multi-head neighborhood aggregation [38]. (Left) The attention logit e_{ij} and normalized coefficient α_{ij} correspond to Eqs. (36) and (37), respectively. (Right) Each head yields simplex-constrained weights, and head outputs are merged by concatenation or averaging.

by maintaining a latent state that evolves over time [28, 29, 39]. Figure 2.4 summarizes the three viewpoints most relevant here: a continuous-time state-space description, the interpretation of memory through state dynamics, and an efficient discrete-time realization suitable for neural sequence modeling.

To avoid collision with the waveform notation $x_m[n]$, we denote the SSM latent state by $\mathbf{z}(\xi)$. A continuous-time linear time-invariant state-space system is

$$\dot{\mathbf{z}}(\xi) = \mathbf{A}\mathbf{z}(\xi) + \mathbf{B}\mathbf{u}(\xi), \quad \mathbf{y}(\xi) = \mathbf{C}\mathbf{z}(\xi) + \mathbf{D}\mathbf{u}(\xi), \quad (39)$$

where ξ is continuous time, $\mathbf{z}(\xi) \in \mathbb{R}^{N_z}$ is the latent state, $\mathbf{u}(\xi) \in \mathbb{R}^{D_u}$ is the input, and $\mathbf{y}(\xi) \in \mathbb{R}^{D_y}$ is the output. The matrices satisfy $\mathbf{A} \in \mathbb{R}^{N_z \times N_z}$, $\mathbf{B} \in \mathbb{R}^{N_z \times D_u}$, $\mathbf{C} \in \mathbb{R}^{D_y \times N_z}$, and $\mathbf{D} \in \mathbb{R}^{D_y \times D_u}$. The matrix $\mathbf{A}$ determines the intrinsic memory timescales of the model.

Discretizing with step size $\Delta > 0$ yields

$$\mathbf{z}_k = \bar{\mathbf{A}}\mathbf{z}_{k-1} + \bar{\mathbf{B}}\mathbf{u}_k, \quad \mathbf{y}_k = \bar{\mathbf{C}}\mathbf{z}_k + \bar{\mathbf{D}}\mathbf{u}_k, \quad (40)$$

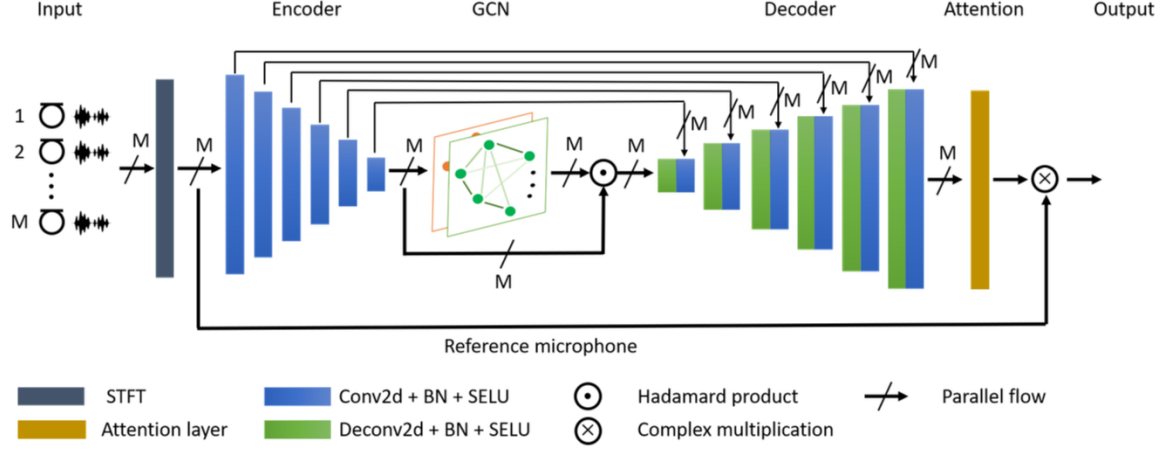


Figure 2.3: Representative encoder–graph–decoder pipeline for graph-based multichannel speech enhancement. “ M ” denotes tensors that retain an explicit microphone/node axis. The graph block performs spatial fusion in a compact latent space; skip connections support U-Net-style reconstruction. A channel-1 spectrum is routed in parallel and combined at the output via complex multiplication (masking interface, eq. (19)).

with

$$\bar{\mathbf{A}} = \exp(\mathbf{A}\Delta), \quad \bar{\mathbf{B}} = \int_0^\Delta \exp(\mathbf{A}\tau)\mathbf{B} d\tau, \quad \bar{\mathbf{C}} = \mathbf{C}, \quad \bar{\mathbf{D}} = \mathbf{D}. \quad (41)$$

Under the zero-order-hold convention used here, discretization changes the state-transition and input matrices, while the output matrices are kept unchanged; the bars on $\bar{\mathbf{C}}$ and $\bar{\mathbf{D}}$ simply mark that they belong to the discrete-time realization. In STFT-based enhancement, the discrete index k typically corresponds to the frame index t , and Δ is associated with the hop duration. Modern SSM layers exploit structured parameterizations and efficient scan or convolution implementations, making them attractive for the long-context modeling developed later in Chapter 5.

2.4.3 Conformer Encoders as Temporal Refinement Modules

The Conformer combines self-attention with a convolutional inductive bias and is widely used for speech sequence modeling [27]. In this thesis, Conformer-style blocks are used mainly as temporal refinement modules after an explicit spatial stage. That is, spatial processing first produces a single-channel spectrum $Z(t, f)$, and the Conformer then refines temporal context to suppress residual interference and reduce artifacts while preserving speech structure.

Figure 2.5 gives an overview of the Conformer encoder architecture. The left panel shows a

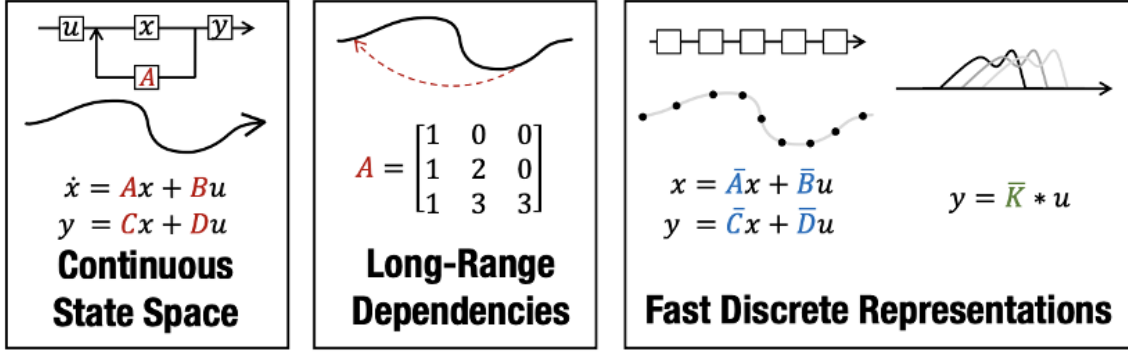


Figure 2.4: State-space modeling viewpoints for long-context sequence modeling [39]: continuous-time state space, long-range dependencies through state dynamics, and efficient discrete-time realizations. In the discrete-time view, $\mathbf{u}_k$, $\mathbf{z}_k$, and $\mathbf{y}_k$ denote the input, latent state, and output vectors, respectively, as in Eqs. (40)–(41). The bars on the discrete-time matrices indicate discretized system matrices.

typical ASR-oriented Conformer encoder stack, while the right panel shows the internal structure of a Conformer block itself. The present thesis mainly reuses the block-level design on the right: multi-head self-attention supplies wider temporal context, and the convolution module preserves local continuity that is important for artifact-free enhancement.

Unless otherwise stated, we use the Conformer block as a temporal refinement module operating at the STFT frame rate after spatial fusion, rather than the full ASR-specific encoder front end with aggressive subsampling. This preserves the time–frequency resolution required for enhancement while still leveraging local-to-global temporal modeling.

2.5 Experimental Protocols

Following common practice in speech enhancement and robust-ASR research, this thesis combines controlled simulated mixtures for supervised training and ablation with established real or semi-real benchmarks for robustness evaluation [40–44]. This design helps separate two questions: whether a method works under controlled ground-truth supervision, and whether it remains reliable under device, room, and domain mismatch. Unless otherwise stated, all audio is resampled to 16 kHz. For intrusive speech-enhancement metrics, the reference signal is aligned to the enhanced output at the same sampling rate and time support.

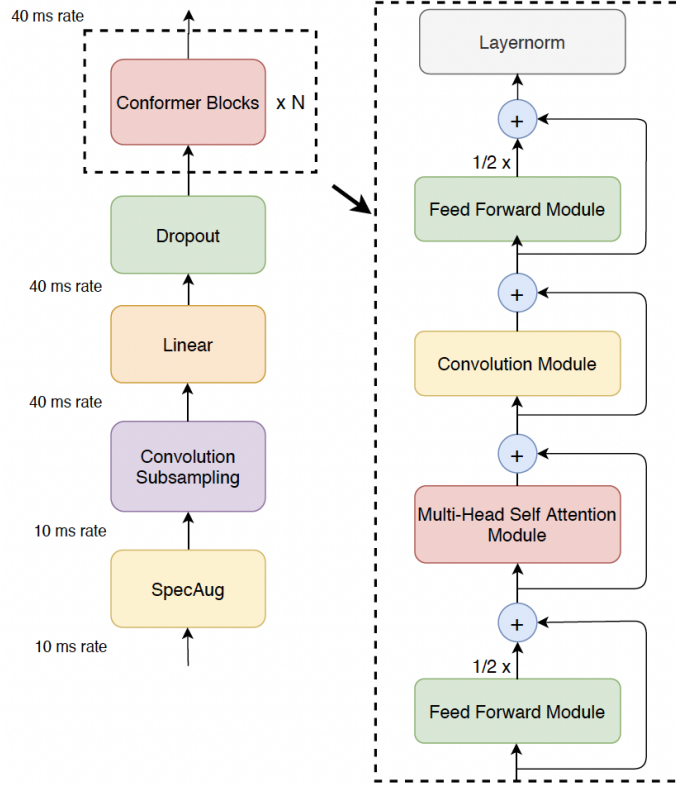


Figure 2.5: Conformer encoder overview and Conformer block [27]. (Left) A typical ASR-oriented Conformer encoder stack. (Right) A Conformer block combining macaron-style feed-forward modules, multi-head self-attention, and a convolution module with residual connections and layer normalization.

2.5.1 Datasets

From Single-Channel Sources to Multichannel Mixtures

Many corpora provide single-channel clean speech, single-channel noise sources, or both. To construct multichannel mixtures, we spatialize these sources with source-to-channel RIRs. For a single-channel mixture,

$$x[n] = s[n] + \gamma v[n], \quad (42)$$

where $s[n]$ is clean speech, $v[n]$ is noise or interference, and γ is chosen to achieve a target signal-to-noise ratio SNR in dB. If Ω denotes the sample set used for energy measurement, we define

$$P_s = \sum_{n \in \Omega} s^2[n], \quad P_v = \sum_{n \in \Omega} v^2[n]. \quad (43)$$

A standard scaling rule is

$$\gamma = \sqrt{\frac{P_s}{P_v 10^{\text{SNR}/10}}}. \quad (44)$$

For multichannel simulation, scaling is usually defined with respect to channel 1 unless the experimental protocol states otherwise. Let $h_m^{(s)}[n]$ and $h_m^{(v)}[n]$ denote the RIRs from the speech and disturbance sources to channel m . Define

$$P_{s,1} = \sum_{n \in \Omega} ((s * h_1^{(s)})[n])^2, \quad P_{v,1} = \sum_{n \in \Omega} ((v * h_1^{(v)})[n])^2.$$

The channel-1 scaling rule becomes

$$\gamma = \sqrt{\frac{P_{s,1}}{P_{v,1} 10^{\text{SNR}/10}}}. \quad (45)$$

The resulting M -channel mixture is

$$x_m[n] = (s * h_m^{(s)})[n] + \gamma (v * h_m^{(v)})[n], \quad m = 1, \dots, M. \quad (46)$$

When multiple interferers are present, the disturbance term becomes a sum of independently spatialized and scaled sources. The experimental chapters instantiate this generic synthesis procedure with different source pools, array layouts, and SNR or reverberation ranges.

Speech Enhancement Datasets

The speech-enhancement chapters in this thesis use three types of data: controlled simulation for supervised training and ablation, standardized multichannel benchmarks for realism and device or environment mismatch, and single-channel mismatch sets for out-of-domain checks. The choice of dataset in each later chapter follows the methodological question being studied, for example topology robustness, deployable spatial filtering under mismatch, long-context modeling, or objective transfer.

TIMIT is used as a clean-speech source for controlled simulation and for seen-speaker versus

unseen-speaker evaluation [45]. The primary simulated multichannel corpus in this thesis combines TIMIT utterances, DNS Challenge noises, and image-source simulated RIRs generated with `pyroomacoustics` [35, 40]. Additional stress-test conditions are created with structured noises from NOISEX-92, especially `babble` and `factory1`, under prescribed SNR settings [46]. Here, “stress test” refers to a deliberately challenging evaluation condition used to probe robustness, such as structured nonstationary noise, low or fixed SNR values, or a mismatch from the main training distribution.

DNS Challenge resources are used both as a large noise pool for simulation and as a standardized benchmark resource in later front-end evaluations [40]. NOISEX-92 is used mainly for controlled stress tests [46]. CHiME-3 and CHiME-4 provide real and simulated multichannel noisy speech recorded with a 6-channel tablet array in everyday acoustic environments, and are important in this thesis because they expose methods to realistic background noise, device effects, and simulation-to-real mismatch [41, 42]. ConferencingSpeech 2021 is used to test transfer to far-field meeting and video-conferencing scenarios with long-form conversational structure and realistic reverberation [43]. L3DAS22 is used to test whether learned spatial reasoning transfers across microphone representations rather than only across one fixed linear-array geometry [47]. VoiceBank-DEMAND, although single-channel, remains useful as an out-of-domain mismatch set [44].

Table 2.1 summarizes the dataset families most directly used for speech-enhancement evaluation in this thesis.

ASR Front-End Evaluation Datasets

The ASR-oriented chapters use benchmarks chosen to isolate front-end behavior under frozen recognizers, to test robustness under real mismatch, and to examine portability across languages and acoustic conditions. Accordingly, later chapters restate only the protocol-specific details and the reason a particular subset of benchmarks is used for a given method.

In the ASR front-end studies, the ASR back end is kept fixed within each benchmark so that word error rate (WER) or character error rate (CER) differences primarily reflect front-end behavior rather than recognizer retraining. DNS Challenge-derived synthetic mixtures are used to test recognition robustness under controlled additive-noise and reverberant conditions [40]. CHiME-4

Table 2.1: Representative dataset families used for speech enhancement in this thesis. Chapter-specific training or split details are given in the corresponding experimental chapters.

Dataset	Channels	Composition	Role in Thesis
TIMIT [45]	1	Clean read speech from 630 speakers	Clean-speech source for controlled simulation
DNS Challenge [40]	1	Large clean-speech and noise pools	Primary noise source; standardized synthetic benchmark
NOISEX-92 [46]	1	Structured evaluation noises such as babble and factory	Controlled stress tests with challenging structured noises at fixed SNRs
CHiME-3/4 [41, 42]	6	Real and simulated tablet-array recordings	Core multichannel real-world benchmark
ConferencingSpeech 2021 [43]	8	Far-field meeting scenarios	Long-form far-field robustness
L3DAS22 [47]	4	FOA virtual 3D audio environments	Spatial-representation generalization
VoiceBank-DEMAND [44]	1	Clean speech mixed with DEMAND noises	Out-of-domain mismatch evaluation

real data provide a stringent generalization test because they include real background noise, device effects, and environment mismatch that are difficult to reproduce synthetically [42]. LibriSpeech is used as a clean read-speech source for controlled mixture analyses and SNR-stratified ASR studies [48]. Aurora-4 is used mainly as an evaluation-only robust-ASR stress test [49]. AISHELL-1 is used in Mandarin front-end evaluations and portability studies [50, 51].

Table 2.2 summarizes the main ASR front-end evaluation datasets.

2.5.2 Evaluation Metrics

Speech enhancement can be evaluated by subjective listening tests, objective signal-level metrics, non-intrusive perceptual predictors, or downstream task metrics. Because repeated human evaluation is expensive and time-consuming, this thesis primarily reports objective measures. The ASR-oriented datasets above already introduce WER and CER, so this section first defines these task metrics and then summarizes the enhancement metrics used in later chapters.

For downstream ASR, word error rate (WER) is the primary English metric. Let N_{ref} be the number of reference words, and let N_{sub} , N_{del} , and N_{ins} be the numbers of substitutions, deletions,

Table 2.2: Main dataset families used for ASR front-end evaluation in this thesis.

Dataset	Channels	Condition / Scale	Role in ASR Evaluation
DNS Challenge [40]	1	Synthetic noisy speech with and without reverberation	Controlled additive-noise robustness benchmark
CHiME-4 [42]	6	Real noisy tablet-array recordings	Real-world mismatch evaluation
LibriSpeech [48]	1	Large-scale clean read speech	Source for controlled SNR analysis
Aurora-4 [49]	1	Additive noise and channel mismatch	Evaluation-only stress test
AISHELL-1 [50]	1	Mandarin speech corpora	Cross-lingual portability studies

and insertions after optimal alignment. Then

$$\text{WER} = \frac{N_{\text{sub}} + N_{\text{del}} + N_{\text{ins}}}{N_{\text{ref}}} \times 100\%. \quad (47)$$

For Mandarin ASR, character error rate (CER) is defined analogously at the character level. If C_{ref} is the number of reference characters and C_{sub} , C_{del} , and C_{ins} are character-level substitutions, deletions, and insertions, then

$$\text{CER} = \frac{C_{\text{sub}} + C_{\text{del}} + C_{\text{ins}}}{C_{\text{ref}}} \times 100\%. \quad (48)$$

Lower WER and CER indicate better recognition accuracy.

Scale-invariant signal-to-distortion ratio (SI-SDR) measures reconstruction fidelity while removing overall gain ambiguity [52]. Let $\mathbf{s} \in \mathbb{R}^{N_{\text{samp}}}$ be the clean reference waveform and $\hat{\mathbf{s}} \in \mathbb{R}^{N_{\text{samp}}}$ the enhanced waveform, where N_{samp} is the number of samples. Define the target projection and residual error as

$$\mathbf{s}_{\text{target}} = \frac{\langle \hat{\mathbf{s}}, \mathbf{s} \rangle}{\|\mathbf{s}\|_2^2} \mathbf{s}, \quad \mathbf{e} = \hat{\mathbf{s}} - \mathbf{s}_{\text{target}}, \quad (49)$$

where $\langle \cdot, \cdot \rangle$ is the inner product and $\|\cdot\|_2$ is the Euclidean norm. Then

$$\text{SI-SDR} = 10 \log_{10} \frac{\|\mathbf{s}_{\text{target}}\|_2^2}{\|\mathbf{e}\|_2^2}. \quad (50)$$

Some chapters also report scale-invariant signal-to-noise ratio (SI-SNR) or signal-to-distortion ratio

(SDR) when those metrics are part of the referenced evaluation protocol.

Perceptual Evaluation of Speech Quality (PESQ) is standardized in ITU-T Recommendation P.862 and predicts subjective quality in a form correlated with mean opinion score (MOS) by comparing a reference signal and a degraded signal after auditory-domain processing [53, 54]. When needed, narrow-band PESQ (NB-PESQ) and wide-band PESQ (WB-PESQ) denote the corresponding PESQ modes. Short-Time Objective Intelligibility (STOI) is an intrusive intelligibility measure based on preservation of short-time temporal-envelope structure, and Extended Short-Time Objective Intelligibility (ESTOI) extends this idea to improve behavior under modulated noise and other challenging conditions [55, 56].

For real noisy recordings without a clean reference, Deep Noise Suppression Mean Opinion Score (DNSMOS) provides non-intrusive MOS-like quality predictions directly from the waveform [57, 58]. DNSMOS P.835 is a P.835-style DNSMOS model that separately predicts speech quality, background-noise quality, and overall quality, following the ITU-T P.835 subjective-test decomposition [59, 60].

Later chapters report only the subset of metrics that is directly relevant to the chapter-specific claim. Enhancement-centric chapters emphasize intrusive quality, intelligibility, and reconstruction metrics when clean references are available, whereas ASR-oriented chapters emphasize WER or CER under fixed back ends. Real-recording sections without clean references use non-intrusive quality predictors such as DNSMOS.

2.5.3 Computational Complexity Measures

Besides enhancement quality and recognition accuracy, deployable front ends should also be computationally efficient. Throughout this thesis, we report model size and operation counts under a specified input configuration. Wall-clock runtime depends strongly on hardware and software implementation and is therefore not used as the primary complexity measure here.

Let N_{param} denote the number of trainable scalar parameters in the enhancement network, excluding fixed transforms such as STFT or inverse STFT (iSTFT). If parameters are stored in b bytes,

the parameter memory footprint is

$$\text{Mem}_{\text{param}} = b N_{\text{param}} \quad \text{bytes.} \quad (51)$$

Let N_{MAC} denote the number of multiply–accumulate (MAC) operations required for one forward pass under a specified input shape. For a network composed of layers $\{\ell\}$,

$$N_{\text{MAC}} = \sum_{\ell} N_{\text{MAC}}^{(\ell)}. \quad (52)$$

In practice, this thesis reports MACs using code-based profilers such as `ptflops` under the same input configuration used for evaluation [61]. When floating-point operations (FLOPs) are also reported, we use the common convention that one MAC corresponds approximately to two FLOPs.

2.6 Conclusion

This chapter established the shared STFT-domain signal model, notation, target definitions, and model interfaces used in the remainder of the thesis. It also summarized representative classical spatial filters, the main neural architecture families reused later, and the evaluation protocols adopted across the experimental studies in later chapters. With these common preliminaries in place, the thesis next turns to topology-robust spatial learning on microphone graphs in Chapter 3.

Chapter 3

Graph-Based Spatial Topology Learning for Multichannel Speech Enhancement

3.1 Introduction

Multichannel speech enhancement (MCSE) exploits spatial diversity in microphone arrays: each microphone observes the same target speech through a distinct acoustic transfer path, whereas interference, background noise, and reverberation exhibit different spatial structure. Spatial cues such as inter-channel phase or level differences and spatial coherence therefore provide degrees of freedom beyond single-channel processing for interference suppression and dereverberation [1, 2, 10].

Chapter 2 reviewed two complementary approaches to MCSE. On the signal-processing side, classical beamformers, including minimum variance distortionless response (MVDR), generalized eigenvalue (GEV), and multichannel Wiener filtering (MWF), are principled and interpretable, but their effectiveness hinges on reliable steering or relative transfer function (RTF) information and well-conditioned spatial covariance matrices (SCMs); in practice, SCM estimation can be unstable under low signal-to-noise ratio (SNR), short effective context, and rapidly nonstationary scenes [1, 3, 62]. On the learning side, modern neural MCSE systems have progressed rapidly, yet many still assume an implicit channel ordering, a fixed reference microphone, or both, and cross-channel fusion is often realized as latent feature mixing tied to a particular channel-index convention [7, 24, 63–65]. Both assumptions become fragile under deployment mismatch: microphone topology

(channel permutation, missing channels, or varying channel count) and array geometry (pose or spacing perturbations) are rarely perfectly controlled.

This chapter addresses that gap by developing topology-robust spatial front ends based on graph modeling, while progressively making the spatial operation more explicit and more deployment-friendly. Graph neural networks provide permutation-equivariant message passing over microphones-as-nodes and data-dependent cross-channel weighting via attention [21–23, 36–38]. The objective is to retain the core benefit of multichannel spatial processing, namely identifying reliable microphones and combining them coherently, while avoiding brittle fixed-ordering assumptions and inference-time SCM inversion whenever possible. The chapter develops this idea in three stages. We begin with complex-domain graph-based spatial feature fusion, which improves topology robustness while preserving phase-sensitive information. We then introduce an explicit per-time–frequency convex weighting interface that exposes an interpretable “filter-and-sum” spatial stream and provides a gain-bounded reference for downstream refinement. Finally, we strengthen this explicit interface with stability-oriented extensions that keep inference covariance-free while improving robustness in more challenging regimes through lightweight inference-time stabilizers and training-only teacher regularization.

The remainder of this chapter therefore progresses from implicit graph-based spatial fusion to an explicit and stabilized convex spatial interface. Section 3.2 introduces complex-domain topology-aware graph fusion and establishes baseline robustness on CHiME-3 and L3DAS22. Section 3.3 reformulates spatial processing as an explicit per-time–frequency convex channel-weighting operator and details the end-to-end system design and evaluation. Section 3.4 strengthens stability and generalization under mismatch by introducing lightweight inference-time stabilizers and training-only teacher regularization, with comprehensive analysis on both spatialized benchmarks and real LibriCSS recordings. Section 3.5 concludes the chapter and motivates the deployability-oriented developments in Chapter 4.

3.2 Complex-Valued GiGCN–TPARN for Topology-Aware Spatial Feature Fusion

This section addresses the limitation that early multichannel neural baselines often rely on implicit channel fusion tied to a fixed channel ordering. We make spatial interaction explicit by treating microphones as graph nodes and performing graph-based processing in the complex STFT domain, thereby preserving phase-sensitive information. Throughout this chapter, we follow the STFT notation and target conventions established in Chapter 2.

3.2.1 Proposed Method

Overall architecture

Figure 3.1 summarizes the Graph-in-Graph Convolutional Network–Triple-Path Attentive Recurrent Network (GiGCN–TPARN) pipeline. Given an M -channel recording, we compute the multichannel STFT and treat microphones as nodes on a fully connected graph so that cross-channel interaction is explicit and permutation equivariant. The spatial module is a complex-valued GiGCN block that performs topology-aware message passing on STFT features while preserving phase-sensitive information. The graph-refined spatial representation is then passed to a temporal module with two long short-term memory (LSTM) branches, one configured for short-term context and one for longer-span context, as shown schematically in Fig. 3.1. Finally, TPARN fuses frequency-, time-, and spatial-stream representations via attention, and the channel-fusion output layer shown in the figure maps the fused features to the enhanced complex spectrum $\hat{S}(t, f)$, followed by inverse STFT (iSTFT) reconstruction.

The design separates responsibilities across modules. GiGCN focuses on spatial reasoning, namely which microphones provide mutually consistent evidence in each time–frequency (TF) region. The decoupled LSTMs model temporal continuity over short and longer effective horizons. TPARN then fuses frequency, time, and space explicitly rather than entangling all interactions in a single monolithic backbone. The figure shows TPARN as a compact fusion block; its internal attention interface is specified in Section 3.2.1 through the stream-fusion formulation in Eqs. (54)–(55), rather than as a separate expanded subfigure.

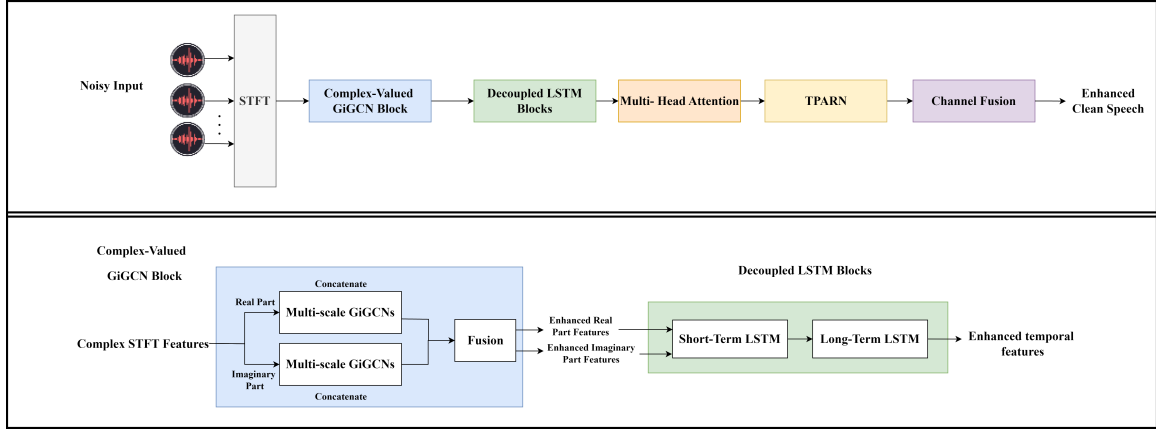


Figure 3.1: Overall architecture of GiGCN-TPARN. The top row shows the complete enhancement pipeline, and the lower row expands the complex-valued GiGCN and decoupled-LSTM components. The TPARN block is shown schematically as the attention-based frequency–time–space fusion stage; its stream-fusion interface is defined in Eqs. (54)–(55).

Complex-domain processing and GiGCN spatial fusion

GiGCN-TPARN processes multichannel STFT features in the complex domain to preserve phase-sensitive spatial cues. Each STFT coefficient is decomposed as $X_m(t, f) = \Re\{X_m(t, f)\} + j \Im\{X_m(t, f)\}$, where $\Re\{\cdot\}$ and $\Im\{\cdot\}$ denote the real and imaginary parts.

To preserve complex coupling while using standard real-valued layers, complex linear maps are implemented by structured real matrices. Let $\mathbf{W} = \mathbf{W}_{\Re} + j\mathbf{W}_{\Im} \in \mathbb{C}^{d_{\text{out}} \times d_{\text{in}}}$ be a complex weight matrix and $\mathbf{z} = \mathbf{z}_{\Re} + j\mathbf{z}_{\Im} \in \mathbb{C}^{d_{\text{in}}}$ be a complex input vector. Then

$$\begin{bmatrix} \Re\{\mathbf{W}\mathbf{z}\} \\ \Im\{\mathbf{W}\mathbf{z}\} \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{\Re} & -\mathbf{W}_{\Im} \\ \mathbf{W}_{\Im} & \mathbf{W}_{\Re} \end{bmatrix} \begin{bmatrix} \mathbf{z}_{\Re} \\ \mathbf{z}_{\Im} \end{bmatrix}, \quad (53)$$

where $\mathbf{W}_{\Re}, \mathbf{W}_{\Im} \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ and $\mathbf{z}_{\Re}, \mathbf{z}_{\Im} \in \mathbb{R}^{d_{\text{in}}}$. Equation (53) enforces the complex multiplication rule in real form and prevents the model from treating real and imaginary parts as unrelated channels.

We define a microphone graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{1, \dots, M\}$ and a fully connected edge set including self-loops. This avoids geometry-specific assumptions and allows the model to learn which inter-channel relations are informative under the training distribution. GiGCN-TPARN

adopts the Graph-in-Graph Convolutional Network (GiGCN) concept [66] to increase expressiveness for complex STFT features: the nested structure enriches both within-node feature interaction and cross-node message passing. To capture spatial correlations at multiple scales, the spatial block uses several GiGCN branches with different local feature-extraction kernels inside the GiGCN block (e.g., 3×3 , 5×5 , and 7×7), whose outputs are concatenated and fused. The resulting topology-aware spatial features are denoted by $\mathbf{H}_{\text{space}}(t, f) \in \mathbb{R}^{D_s}$, where D_s is the spatial feature dimension.

Decoupled temporal modeling and TPARN fusion

Spatial fusion is complemented by temporal and spectral modeling. The decoupled temporal module captures both short-term and longer-span temporal dependencies through dedicated LSTM processing, shown schematically in Fig. 3.1 as short-term and long-term LSTM stages. We denote the resulting temporal features collectively by $\mathbf{H}_{\text{time}}(t, f) \in \mathbb{R}^{D_t}$, where D_t is the temporal feature dimension. In parallel, a frequency-path encoder produces $\mathbf{H}_{\text{freq}}(t, f) \in \mathbb{R}^{D_f}$, where D_f is the frequency feature dimension.

TPARN fuses the frequency, time, and spatial streams using multi-head attention (MHA) [65]. To avoid confusing the three feature streams with the standard query, key, and value arguments of MHA, we write the TPARN block as a stream-fusion operator. The three streams are first projected to a common attention dimension D_a :

$$\begin{aligned}\mathbf{G}_{\text{freq}}(t, f) &= \mathbf{P}_{\text{freq}} \mathbf{H}_{\text{freq}}(t, f), \\ \mathbf{G}_{\text{time}}(t, f) &= \mathbf{P}_{\text{time}} \mathbf{H}_{\text{time}}(t, f), \\ \mathbf{G}_{\text{space}}(t, f) &= \mathbf{P}_{\text{space}} \mathbf{H}_{\text{space}}(t, f),\end{aligned}\tag{54}$$

where $\mathbf{P}_{\text{freq}} \in \mathbb{R}^{D_a \times D_f}$, $\mathbf{P}_{\text{time}} \in \mathbb{R}^{D_a \times D_t}$, and $\mathbf{P}_{\text{space}} \in \mathbb{R}^{D_a \times D_s}$ are learned projections. We then write the fusion step as

$$\mathbf{H}_{\text{TPARN}}(t, f) = \mathcal{A}_{\text{TPARN}}(\mathbf{G}_{\text{freq}}(t, f), \mathbf{G}_{\text{time}}(t, f), \mathbf{G}_{\text{space}}(t, f)),\tag{55}$$

where $\mathcal{A}_{\text{TPARN}}(\cdot)$ denotes the MHA-based TPARN fusion block and $\mathbf{H}_{\text{TPARN}}(t, f) \in \mathbb{R}^{D_{\text{fuse}}}$ is the

fused representation. The three arguments in Eq. (55) are the frequency, time, and spatial streams. They are not intended to be the raw (Q, K, V) arguments of a standard attention call; those internal query, key, and value projections are learned inside $\mathcal{A}_{\text{TPARN}}$. A lightweight reconstruction head then predicts the enhanced complex spectrum $\hat{S}(t, f) \in \mathbb{C}$ by regressing its real and imaginary parts.

Training objective

As defined in Chapter 2, $s[n] \in \mathbb{R}$ denotes the protocol-defined target waveform (e.g., a reference-microphone target image) and $\hat{s}[n] \in \mathbb{R}$ the enhanced waveform. Also, $S(t, f)$ and $\hat{S}(t, f)$ represent their complex STFTs, with $f \in \mathcal{F}_+$ and $F \triangleq |\mathcal{F}_+|$. GiGCN-TPARN uses a combined time-domain and frequency-domain mean-squared-error (MSE) objective:

$$\mathcal{L}_{\text{time}} = \frac{1}{N_{\text{samp}}} \sum_{n=1}^{N_{\text{samp}}} (s[n] - \hat{s}[n])^2, \quad \mathcal{L}_{\text{freq}} = \frac{1}{TF} \sum_{t=1}^T \sum_{f \in \mathcal{F}_+} |S(t, f) - \hat{S}(t, f)|^2, \quad (56)$$

where N_{samp} is the number of waveform samples used in the loss and T is the number of frames used in the STFT-domain loss. The final objective is $\mathcal{L} = \alpha \mathcal{L}_{\text{time}} + \beta \mathcal{L}_{\text{freq}}$, with $\alpha = \beta = 0.5$ unless stated otherwise.

3.2.2 Experimental Setup

Datasets. This section evaluates topology-aware graph fusion on two complementary benchmarks: CHiME-3 tests robustness to real acoustic and device mismatch, whereas L3DAS22 tests transfer to a different spatial representation. CHiME-3 provides 6-channel tablet-array recordings in everyday noisy environments and includes official clean references used for intrusive evaluation [41], making it a stringent test of topology-aware spatial fusion under real acoustics and device or environment mismatch. L3DAS22 provides four-channel first-order Ambisonics (FOA) recordings in simulated spatial scenes [47]. Using L3DAS22 is particularly helpful here because it changes the multichannel representation, namely FOA instead of a conventional linear array, and therefore probes whether complex-domain graph fusion learns transferable spatial relations rather than overfitting to a single array geometry.

Implementation details. All signals are sampled at 16 kHz. We compute the STFT with a 1024-sample window, a 512-sample hop, and a 1024-point FFT. The integrated model is trained using Adam with learning rate 10^{-4} , batch size 16, for 100 epochs. Mixtures are augmented with diverse noise types and SNR conditions; unless stated otherwise, SNRs are sampled in 0–20 dB.

For the L3DAS22 comparison, the official composite score is computed as

$$\text{Metric} = \frac{\text{STOI} + (1 - \text{WER})}{2}, \quad (57)$$

where WER is obtained by decoding the enhanced speech with a pretrained Wav2Vec2.0 back end [67].

3.2.3 Experimental Results and Analysis

Results on CHiME-3. Table 3.1 reports PESQ and STOI separately for the four CHiME-3 environments. GiGCN–TPARN gives the strongest STOI in all four environments and the strongest PESQ in CAF and PED, while STGCSEN gives the highest PESQ in BUS and STR. Because the table contains single-run scores without confidence intervals, the differences should be interpreted descriptively rather than as a formal significance test; the main conclusion is the consistent cross-condition improvement produced by topology-aware complex spatial fusion.

Ablation interpretation (CHiME-3). Using only a single GiGCN branch reduces both PESQ and STOI, suggesting that spatial correlations at different receptive fields provide complementary cues. Removing either the short-term or long-term LSTM branch degrades performance, consistent with the need to model both local dynamics and longer-span structure. Removing the spatial path in TPARN causes a clear drop, indicating that spatial cues provide information not recoverable by time–frequency modeling alone.

Results on L3DAS22 (FOA spatial scenes). Table 3.2 shows that GiGCN–TPARN attains the highest STOI and the lowest WER, resulting in the best composite Metric defined in (57). Because FOA encodes spatial information differently from conventional linear arrays, this result indicates that complex-domain graph fusion can exploit spatial structure beyond a single fixed array representation.

Table 3.1: Ablation experiments and performance comparison on the CHiME-3 dataset.

Model	PESQ				STOI			
	BUS	CAF	PED	STR	BUS	CAF	PED	STR
GiGCN-TPARN (ours)	1.942	1.863	1.899	1.925	0.938	0.910	0.927	0.935
Single 3×3 GiGCN block	1.781	1.757	1.787	1.803	0.919	0.902	0.911	0.931
Single 5×5 GiGCN block	1.797	1.768	1.792	1.808	0.921	0.905	0.913	0.933
Only short-term LSTM	1.784	1.760	1.788	1.805	0.923	0.907	0.914	0.934
Only long-term LSTM	1.777	1.754	1.782	1.798	0.917	0.900	0.909	0.929
Only frequency and time paths	1.773	1.748	1.775	1.793	0.916	0.899	0.908	0.928
Simplified channel fusion	1.786	1.763	1.791	1.807	0.918	0.903	0.912	0.932
Noisy (channel 1)	1.272	1.153	1.176	1.263	0.903	0.808	0.807	0.877
CNN [68]	1.775	1.540	1.618	1.751	0.920	0.885	0.882	0.913
LSTM-IPD [43]	1.908	1.496	1.512	1.730	0.921	0.847	0.835	0.890
UNet-GCN [21]	1.655	1.448	1.495	1.645	0.918	0.875	0.871	0.909
STGCSEN [69]	2.002	1.743	1.747	1.941	0.923	0.891	0.876	0.914

Table 3.2: Ablation experiments and performance comparison on the L3DAS22 dataset.

Model	WER	STOI	Metric
GiGCN-TPARN (ours)	0.065	0.941	0.938
Single 3×3 GiGCN block	0.069	0.936	0.934
Single 5×5 GiGCN block	0.070	0.935	0.932
Only short-term LSTM	0.068	0.937	0.935
Only long-term LSTM	0.071	0.934	0.931
Only frequency and time paths	0.072	0.932	0.930
Simplified channel fusion	0.070	0.933	0.932
Spatial-DCCRN [70]	0.071	0.931	0.931
SEU Speech [71]	0.101	0.902	0.900
JLESS [71]	0.174	0.836	0.831
CCA Speech [71]	0.240	0.831	0.796

Transition to an explicit and inspectable spatial operator. The preceding results demonstrate that topology-aware graph message passing in the complex STFT domain can effectively exploit inter-channel structure across both real recordings and simulated spatial scenes. Nevertheless, the spatial component at this stage remains predominantly a latent feature-fusion mechanism: spatial behavior is encoded implicitly in learned representations, without exposing an explicit per-time–frequency mixing rule comparable to beamforming weights. This limits interpretability and makes it difficult to directly enforce gain control and stability when spatial evidence is weak.

To address this limitation, the next section reformulates the spatial front end as an explicit

convex combining operator over microphone channels. Specifically, the model predicts simplex-constrained weights at each (t, f) and forms a bounded “filter-and-sum” reference stream that can be inspected, analyzed, and stabilized by design. This explicit spatial interface provides a bridge between classical array processing and neural refinement: subsequent temporal modeling and complex-domain correction operate on a bounded spatial reference rather than on entangled multi-stream features.

3.3 MVDR-Consistent Graph-Attention Spatial Filtering with Local Conformer

Building on the topology-aware but largely latent fusion in Section 3.2, this section exposes the spatial operation as an explicit per-time-frequency mixing rule. The key idea is to predict simplex-constrained channel weights on a microphone graph and to form a bounded reference stream by a convex “filter-and-sum” combination. This makes the spatial behavior directly inspectable, in the spirit of beamforming weights in Chapter 2, while keeping the subsequent refinement fully learnable and end-to-end trainable.

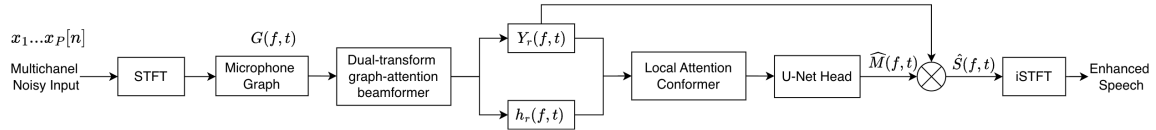


Figure 3.2: Overall pipeline of the proposed graph-attention spatial filter with a local Conformer and complex-mask refinement. The graph-attention spatial combiner corresponds to Eq. (62), and the cRM-based refinement corresponds to Eq. (68). In the source figure, P denotes the microphone count ($P = M$ in the text), and the label $h_r(f, t)$ corresponds to the reference-node embedding $\mathbf{h}_r^{(L_s)}(t, f)$.

3.3.1 Proposed Method

Figure 3.2 summarizes the end-to-end pipeline. Multichannel audio is mapped to the STFT domain and processed by a microphone-graph attention front end that outputs an explicit convex

spatially filtered reference stream $Y_r(t, f)$. This $Y_r(t, f)$ plays the role of the single-channel spatial reference $Z(t, f)$ in Chapter 2 (Eq. (22)), but we keep a dedicated symbol to emphasize that it is formed by the reference-row weights of a microphone graph. A bounded-context Conformer then refines this reference over time, and a lightweight U-Net predicts a complex ratio mask (cRM) $\widehat{M}(t, f)$ for residual magnitude and phase correction (the masking interface in Chapter 2, Eq. (19)) before waveform reconstruction. The system is trained end-to-end with a phase-aware reconstruction objective.

A core design choice is to use real, convex spatial weights in the spatial stage, while delegating explicit complex correction to the cRM head. This division of labor stabilizes the spatial stage through gain-bounded convex combining, while preserving complex fidelity in the final enhanced spectrum through the learned mask.

Spatial feature extraction on a microphone graph

We operate in the multichannel STFT domain. Let $X_m(t, f) \in \mathbb{C}$ be the STFT coefficient corresponding to microphone $m \in \{1, \dots, M\}$. For each (t, f) we construct a directed, fully connected microphone graph whose nodes are channels; self-loops are included, hence $\mathcal{N}(i) = \{1, \dots, M\}$ for all nodes.

At spatial layer $\ell \in \{0, 1, \dots, L_s - 1\}$, node i carries a real feature vector $\mathbf{h}_i^{(\ell)}(t, f) \in \mathbb{R}^D$, where D is the node-feature width and L_s is the number of spatial layers. Node features are initialized by embedding Cartesian STFT components:

$$\mathbf{h}_i^{(0)}(t, f) = \text{PReLU}\left(\mathbf{W}_{\text{in}}^{\text{sp}} [\Re\{X_i(t, f)\}, \Im\{X_i(t, f)\}]^T + \mathbf{b}_{\text{in}}^{\text{sp}}\right), \quad (58)$$

where $\mathbf{W}_{\text{in}}^{\text{sp}} \in \mathbb{R}^{D \times 2}$ and $\mathbf{b}_{\text{in}}^{\text{sp}} \in \mathbb{R}^D$.

The spatial module is a dual-projection graph-attention front end based on GATv2 [23]. We use two linear projections to form receiver-specific queries and sender-specific keys:

$$\mathbf{q}_i^{(\ell)}(t, f) = \mathbf{W}_1^{(\ell)} \mathbf{h}_i^{(\ell)}(t, f), \quad \mathbf{k}_j^{(\ell)}(t, f) = \mathbf{W}_2^{(\ell)} \mathbf{h}_j^{(\ell)}(t, f), \quad (59)$$

where $\mathbf{W}_1^{(\ell)}, \mathbf{W}_2^{(\ell)} \in \mathbb{R}^{d_{\text{att}} \times D}$ and d_{att} is the attention projection dimension.

A feed-forward scorer $F : \mathbb{R}^{2d_{\text{att}}} \rightarrow \mathbb{R}$ produces attention logits

$$e_{ij}^{(\ell)}(t, f) = F([\mathbf{q}_i^{(\ell)}(t, f), \mathbf{k}_j^{(\ell)}(t, f)]), \quad (60)$$

which are normalized over the sender index:

$$\alpha_{ij}^{(\ell)}(t, f) = \frac{\exp(e_{ij}^{(\ell)}(t, f))}{\sum_{\kappa \in \mathcal{N}(i)} \exp(e_{i\kappa}^{(\ell)}(t, f))}, \quad \alpha_{ij}^{(\ell)}(t, f) \in [0, 1], \quad \sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(\ell)}(t, f) = 1. \quad (61)$$

At the last spatial layer, we define the reference-row weights by

$$\alpha_{rj}(t, f) \triangleq \alpha_{rj}^{(L_s-1)}(t, f), \quad Y_r(t, f) = \sum_{j=1}^M \alpha_{rj}(t, f) X_j(t, f), \quad (62)$$

where $r \in \{1, \dots, M\}$ is the chosen reference index. Equation (62) is exactly a convex combining operator (Chapter 2, Eq. (22)) with $\alpha_j(t, f) \equiv \alpha_{rj}(t, f)$.

For representation learning, node features are updated via message passing:

$$\mathbf{h}_i^{(\ell+1)}(t, f) = \text{PReLU}\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(\ell)}(t, f) \mathbf{W}_v^{(\ell)} \mathbf{h}_j^{(\ell)}(t, f)\right), \quad (63)$$

where $\mathbf{W}_v^{(\ell)} \in \mathbb{R}^{D \times D}$ is the value-transformation matrix of layer ℓ : it maps each sender-node embedding into the message/value space before the attention-weighted average is taken, analogous to the value projection in standard attention. The pair $(Y_r(t, f), \mathbf{h}_r^{(L_s)}(t, f))$ is then forwarded to the temporal encoder.

Relation to MVDR magnitude profiles

The spatial weights in (62) are constrained to be real and convex. They should therefore not be interpreted as approximating complex MVDR coefficients directly. Rather, the relevant comparison is at the level of channel-importance profiles.

Let $\Phi_{nn}(f)$ denote the disturbance SCM (equivalently, the disturbance or noise covariance in

Chapter 2) and let $\mathbf{d}(f) \in \mathbb{C}^M$ denote the steering or RTF vector. The MVDR beamformer at frequency f is

$$\mathbf{w}_{\text{MVDR}}(f) = \frac{\Phi_{nn}^{-1}(f) \mathbf{d}(f)}{\mathbf{d}^H(f) \Phi_{nn}^{-1}(f) \mathbf{d}(f)} \in \mathbb{C}^M. \quad (64)$$

Define its numerator vector

$$\mathbf{v}(f) \triangleq \Phi_{nn}^{-1}(f) \mathbf{d}(f).$$

A normalized magnitude profile can then be written as

$$\pi_j(f) = \frac{|[\mathbf{v}(f)]_j|}{\sum_{m=1}^M |[\mathbf{v}(f)]_m|}, \quad j = 1, \dots, M. \quad (65)$$

The learned reference-row weights $\alpha_{rj}(t, f)$ are thus better viewed as a bounded, time-varying surrogate of the MVDR-style channel-importance profile (65), rather than as a direct approximation of the complex beamformer itself. This interpretation motivates the term “MVDR-consistent” used here: the learned spatial interface is encouraged to follow the same notion of channel reliability while remaining inference-time covariance-free and gain-bounded because of the convex constraint.

Temporal sequence modeling with bounded-context conformer

The temporal encoder adopts a Conformer-style block [27] and operates on the spatial outputs of Section 3.3.1. Let $Y_r(t, f) \in \mathbb{C}$ be the spatially filtered spectrum and $\mathbf{h}_r^{(L_s)}(t, f) \in \mathbb{R}^D$ the reference-node embedding. For streaming-style processing, computations are described within one chunk of T_{ck} frames; for notational simplicity, we write $t \in \{1, \dots, T_{\text{ck}}\}$ within each chunk.

For each frequency bin f , we form the time sequence

$$\mathbf{x}_t(f) = [\Re\{Y_r(t, f)\}, \Im\{Y_r(t, f)\}, \mathbf{h}_r^{(L_s)}(t, f)] \in \mathbb{R}^{D_{\text{in}}}, \quad D_{\text{in}} = 2 + D.$$

To bound latency and computation, self-attention is restricted to a symmetric local window of $\pm w$ frames. Using single-head notation for clarity, let d_{tmp} be the temporal projection dimension and define

$$\mathbf{q}_t(f) = \mathbf{W}_q^{(T)} \mathbf{x}_t(f), \quad \mathbf{k}_\tau(f) = \mathbf{W}_k^{(T)} \mathbf{x}_\tau(f), \quad \mathbf{v}_\tau(f) = \mathbf{W}_v^{(T)} \mathbf{x}_\tau(f),$$

where $\mathbf{W}_q^{(T)}, \mathbf{W}_k^{(T)}, \mathbf{W}_v^{(T)} \in \mathbb{R}^{d_{\text{tmp}} \times D_{\text{in}}}$. With the window set $\mathcal{W}(t) = \{\tau : \max(1, t - w) \leq \tau \leq \min(T_{\text{ck}}, t + w)\}$, the localized attention output is

$$\mathbf{y}_t(f) = \sum_{\tau \in \mathcal{W}(t)} \frac{\exp(\mathbf{q}_t^\top(f) \mathbf{k}_\tau(f) / \sqrt{d_{\text{tmp}}})}{\sum_{\kappa \in \mathcal{W}(t)} \exp(\mathbf{q}_t^\top(f) \mathbf{k}_\kappa(f) / \sqrt{d_{\text{tmp}}})} \mathbf{v}_\tau(f). \quad (66)$$

To reduce parameter cost in large linear maps, we apply low-rank factorization:

$$\mathbf{W}_{\text{proj}} \approx \mathbf{U} \mathbf{V}, \quad \mathbf{W}_{\text{proj}} \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}, \quad \mathbf{U} \in \mathbb{R}^{d_{\text{out}} \times r_{\text{LR}}}, \quad \mathbf{V} \in \mathbb{R}^{r_{\text{LR}} \times d_{\text{in}}}, \quad r_{\text{LR}} < \min(d_{\text{out}}, d_{\text{in}}), \quad (67)$$

where $\mathbf{W}_{\text{proj}}$ denotes any large dense projection matrix in the temporal Conformer block, such as the query/key/value projections, the attention output projection, or the feed-forward projections. In implementation, the factorization is applied only to these temporal projection matrices, not to the microphone-graph spatial projections. The scalar r_{LR} is the low-rank dimension.

Mask prediction and speech reconstruction

Given the spatial reference $Y_r(t, f)$ and the temporal features, a lightweight U-Net predicts a complex ratio mask $\widehat{M}(t, f) \in \mathbb{C}$ [72]. The enhanced spectrum is obtained by

$$\widehat{S}(t, f) = \widehat{M}(t, f) Y_r(t, f), \quad (68)$$

followed by iSTFT overlap-add reconstruction.

The system is trained end-to-end with a phase-aware reconstruction objective. For notational compactness, its explicit PCM form is given later in Section 3.4.1; the same loss form is used here.

3.3.2 Experimental Setup

Datasets. We evaluate on two spatialized multichannel datasets generated by convolving clean speech and noise with simulated room impulse responses using the image-source method. These two datasets are intentionally complementary: WSJCAM0 emphasizes reverberant speech under

controlled conditions, whereas DNS Challenge 2020 (DNS-2020) stresses stronger and more diverse additive interference. This pairing allows us to test whether the explicit convex spatial interface remains effective across reverberation-dominant and denoising-dominant regimes. A four-microphone circular array (radius 10 cm) is used, and all signals are resampled to 16 kHz. To avoid leakage, room geometries, impulse-response sets, and source–array configurations are disjoint across training, validation, and test; sources are at least 0.5 m from walls, source–array distance is sampled in $[0.75, 2]$ m, and reverberation time for a 60-dB decay (RT_{60}) is sampled in $[0.2, 1.2]$ s.

WSJCAM0 is considered because it provides controlled reverberant speech with moderate additive noise, making it suitable for isolating the behavior of the explicit convex spatial operator under reverberation. For WSJCAM0, clean speech is drawn from WSJCAM0 [73] and background noises follow the REVERB challenge protocol [74]; mixtures use SNR in $[5, 20]$ dB. DNS-2020 is included because its larger and more diverse noise pool stresses generalization to strongly nonstationary interference. For DNS-2020, we spatialize the DNS Challenge 2020 corpus [40]; segments are 3–6 s for training and 3–10 s for validation or testing, with SNR in $[-10, 10]$ dB. Together, the two datasets let us assess whether the explicit graph-attention spatial interface remains effective across both reverberation-dominant and noise-diverse conditions.

Implementation details. Inputs are transformed into the complex STFT domain with a 32 ms analysis window ($L = 512$ samples) and 75% overlap (hop $R = 128$ samples), yielding 257 nonnegative-frequency bins per frame. For streaming-style evaluation, we adopt chunked processing with chunk length $T_{\text{ck}} = 126$ frames and shift $T_{\text{sh}} = 63$ frames, followed by overlap-and-stitch on the waveform. Local self-attention uses a fixed window of $\pm w$ frames with $w = 4$. Model width is set to $D = 128$ across modules. Training uses randomly cropped 4 s segments with batch size 8 using Adam (initial learning rate 4×10^{-4} , halved after 5 epochs). Unless otherwise specified, we use the first microphone as the reference stream ($r = 1$), $L_s = 2$ spatial attention layers, and $L_t = 6$ local-Conformer blocks.

Table 3.3: Experimental results on spatialized WSJCAM0 and DNS Challenge 2020 corpus.

Method	WSJCAM0			DNS-2020			Complexity	
	SI-SDR (dB)	PESQ	STOI	SI-SDR (dB)	PESQ	STOI	Param (M)	GMAC/s
Unprocessed	-3.8	1.38	0.709	-7.6	1.38	0.638	–	–
FasNet-TAC [63]	8.2	2.93	0.947	4.7	2.26	0.865	2.9	55.3
ADCN [64]	12.0	3.42	0.973	7.8	2.84	0.923	9.3	72.8
TPARN [65]	10.4	3.43	0.969	8.4	2.75	0.919	3.2	71.8
ADCN-TPARN [64]	13.8	3.64	0.980	10.0	2.99	0.937	40.0	360.0
DeFT-AN [24]	15.7	3.63	0.981	9.9	3.01	0.924	2.7	95.6
Ours	16.3	3.69	0.982	10.3	3.05	0.927	3.1	45.4

3.3.3 Experimental Results and Analysis

Overall performance comparison

Table 3.3 compares the proposed model with recent multichannel baselines on spatialized WSJCAM0 and DNS-2020. On WSJCAM0, the proposed system improves SI-SDR over TPARN and yields consistent improvements in PESQ and STOI, indicating that explicit per-TF convex spatial weighting combined with bounded-context temporal encoding reduces distortion while improving perceived quality and intelligibility. On DNS-2020, the proposed model also improves all metrics over both TPARN and DeFT-AN.

The model remains computationally compact. Its measured complexity is 45.4 GMAC/s, about half that of DeFT-AN. This behavior is consistent with the expected $O(T_{\text{ck}} w d_{\text{tmp}})$ scaling of localized attention together with depthwise-separable convolution and low-rank projections.

Ablation studies

We consider four ablation scenarios on spatialized WSJCAM0: removal of sub-modules, module order, spatial-backbone choice, and local-attention window size. The corresponding results are summarized in Table 3.4.

(A) *Removal of sub-modules.* Removing the temporal block leads to the largest degradation, while removing the spatial block also causes a substantial drop; removing the mask produces a smaller but consistent loss. These trends align with the intended module roles: temporal modeling mitigates room-induced smearing and slowly varying interference; spatial mixing suppresses structured interferers before refinement; and the mask corrects residual complex errors on top of the

Table 3.4: Ablation on spatialized WSJCAM0. The default pipeline is S–T–M; **Proposed** denotes the full model.

Variations	SI-SDR (dB)	PESQ	STOI	Param (M)	GMAC/s
<i>(A) Remove Sub-Module</i>					
sub-S	12.1	3.40	0.959	2.2	31.3
sub-T	6.1	3.15	0.926	2.3	29.9
sub-M	14.7	3.57	0.976	3.1	44.6
<i>(B) Module Order</i>					
S–M–T	14.3	3.53	0.972	3.1	44.8
T–S–M	13.9	3.49	0.968	3.1	44.8
<i>(C) Spatial Backbone</i>					
GCN	12.6	3.46	0.963	2.8	39.1
GAT	13.4	3.51	0.969	2.9	40.1
GIN	13.8	3.52	0.970	2.9	40.5
GATv2 (standard)	14.5	3.58	0.975	3.0	44.3
<i>(D) Conformer (Local Window)</i>					
Global (MHA)	14.2	3.55	0.973	3.1	62.7
Local (± 2)	14.9	3.59	0.976	3.1	44.8
Local (± 6)	15.8	3.64	0.981	3.1	46.0
Proposed	16.3	3.69	0.982	3.1	45.4

spatially combined reference.

(B) Module order. The proposed model adopts S–T–M (spatial combining $\rightarrow$ temporal encoding $\rightarrow$ complex-mask refinement). Both alternative permutations underperform, indicating that early spatial combining provides a cleaner reference stream that is most effective when refined temporally before applying the final complex correction.

(C) Spatial backbone. Among GCN, GAT, and GIN, the standard GATv2 variant performs best, supporting the importance of asymmetric, data-dependent inter-channel relations under dynamic acoustics.

(D) Local attention window. Replacing global attention with localized windows improves accuracy while reducing computation. The default ± 4 setting offers a strong accuracy–efficiency balance, whereas larger windows provide only diminishing returns.

Transition to stability-oriented extensions under mismatch. The explicit convex spatial interface above turns graph attention into a gain-bounded, inspectable spatial operator and yields a stable reference stream for downstream refinement. However, when mismatch becomes more severe (e.g., lower SNR, stronger reverberation, and real-device variability), two practical issues become

prominent: (i) weight jitter and abrupt microphone switching, most noticeably in low-frequency bins where phase-derived cues are less reliable; and (ii) occasional degenerate patterns in which noisy logits drive the distribution toward near-hard selection, reducing robustness.

The next section strengthens the reliability of the convex weights while preserving the same overall S–T–M pipeline and maintaining covariance-free inference. It introduces lightweight inference-time stabilizers (learnable temperature scaling and low-band cross-frequency coupling) and uses beamforming-inspired structure only as training-time guidance (teacher alignment under stop-gradient), complemented by an STFT projection-consistency regularizer to discourage STFT-inconsistent artifacts. Together, these elements aim to reduce variability in the spatial interface without sacrificing deployability.

3.4 Stable Covariance-Free Graph-Attention Spatial Filtering with Learnable Temperature

This section extends the covariance-free convex graph-attention spatial filter introduced in Section 3.3 with stability- and generalization-oriented mechanisms. The objective is to retain the deployment advantages of the method in Section 3.3, namely an explicit convex spatial interface and no inference-time SCM estimation, while improving robustness under stronger mismatch and reducing attention instability, especially in low-frequency bands where spatial cues are often ambiguous. Two practical failure modes motivate the design: (i) weight jitter, namely abrupt microphone switching across adjacent TF bins; and (ii) degenerate near-hard selection, namely overly peaky simplex weights driven by unreliable logits.

Crucially, the overall S–T–M pipeline is unchanged. All modifications in this section act on the spatial interface that produces the convex reference stream; temporal refinement and complex masking remain the same as in Section 3.3. At inference, the spatial stage stays covariance-free. SCM-derived quantities are used only for training-time regularization and never appear on the inference path.

3.4.1 Proposed Method

Overall architecture

We use the same chunked processing convention as in Section 3.3.1. Within a chunk, $t \in \{1, \dots, T_{\text{ck}}\}$ indexes STFT frames and $f \in \mathcal{F}_+$ indexes nonnegative-frequency bins (Chapter 2, Eq. (3)). Let $X_m(t, f) \in \mathbb{C}$ denote the STFT coefficient of microphone $m \in \{1, \dots, M\}$.

The spatial front end operates on a fully connected directed microphone graph with self-loops and produces a gain-bounded spatial reference stream by convex combination:

$$Y_r(t, f) = \sum_{j=1}^M \tilde{a}_{rj}(t, f) X_j(t, f), \quad \tilde{a}_{rj}(t, f) \geq 0, \quad \sum_{j=1}^M \tilde{a}_{rj}(t, f) = 1, \quad (69)$$

where r is the reference index and $\tilde{a}_{rj}(t, f)$ are the final stabilized simplex weights predicted by the graph-attention stack.

This convex interface yields an explicit gain bound. By triangle inequality, for every TF bin,

$$|Y_r(t, f)| \leq \sum_{j=1}^M \tilde{a}_{rj}(t, f) |X_j(t, f)| \leq \max_{1 \leq j \leq M} |X_j(t, f)|. \quad (70)$$

Thus the spatial stage cannot amplify the magnitude beyond the maximum input magnitude at that bin, which directly mitigates instability when the weights are uncertain.

Compared with the method in Section 3.3, we introduce four additional elements that target the two failure modes above while keeping inference covariance-free: (i) a learnable per-frequency temperature $\tau(f)$ applied to the final attention layer to control sharpness of the output distribution; (ii) a low-band cross-frequency coupling (CFC) step that smooths low-frequency reference-row weights across adjacent frequency bins; (iii) a training-only MVDR-inspired teacher-alignment term, constructed from masked SCM estimates under a stop-gradient operator; and (iv) a training-only STFT projection-consistency term that discourages STFT-inconsistent artifacts produced by masking and refinement. Among these, (i) and (ii) are lightweight inference-time stabilizers, whereas (iii) and (iv) are used only during training and therefore do not affect inference complexity.

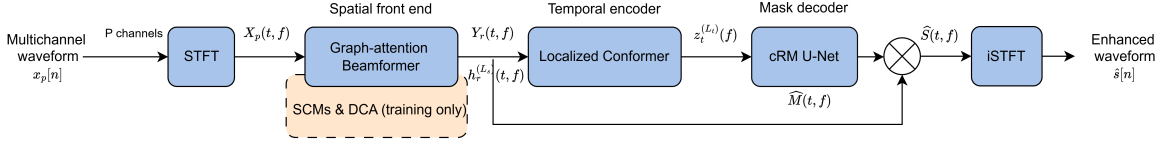


Figure 3.3: Overall architecture of the stabilized covariance-free spatial filtering system. At inference, the spatial front end performs covariance-free convex spatial combining, producing the reference stream $Y_r(t, f)$ in Eq. (69); SCM-based teacher construction and alignment are used during training only. In the source figure, P denotes the microphone count ($P = M$ in the text), the reference-row attention corresponds to the final-layer weights $a_{rj}(t, f)$, and the stabilized convex-combining weights are denoted by $\tilde{a}_{rj}(t, f)$ in the text.

Spatial graph-attention weight prediction

Let L_s denote the number of spatial attention layers. Node i at layer ℓ carries an embedding $\mathbf{h}_i^{(\ell)}(t, f) \in \mathbb{R}^D$. We initialize node features from Cartesian STFT components:

$$\mathbf{h}_i^{(0)}(t, f) = \text{PReLU}\left(\mathbf{W}_{\text{in}}^{\text{sp}} [\Re\{X_i(t, f)\}, \Im\{X_i(t, f)\}]^T + \mathbf{b}_{\text{in}}^{\text{sp}}\right), \quad (71)$$

where $\mathbf{W}_{\text{in}}^{\text{sp}} \in \mathbb{R}^{D \times 2}$ and $\mathbf{b}_{\text{in}}^{\text{sp}} \in \mathbb{R}^D$.

Each spatial layer computes attention logits by a dual-projection interaction. We form query and key projections

$$\mathbf{q}_i^{(\ell)}(t, f) = \mathbf{W}_1^{(\ell)} \mathbf{h}_i^{(\ell)}(t, f), \quad \mathbf{k}_j^{(\ell)}(t, f) = \mathbf{W}_2^{(\ell)} \mathbf{h}_j^{(\ell)}(t, f), \quad (72)$$

with $\mathbf{W}_1^{(\ell)}, \mathbf{W}_2^{(\ell)} \in \mathbb{R}^{d_{\text{att}} \times D}$, where d_{att} is the attention width. Logits are computed by an element-wise interaction scorer

$$e_{ij}^{(\ell)}(t, f) = \mathbf{u}^{(\ell)\top} \text{LeakyReLU}\left((\mathbf{U}_q^{(\ell)} \mathbf{q}_i^{(\ell)}(t, f)) \odot (\mathbf{U}_k^{(\ell)} \mathbf{k}_j^{(\ell)}(t, f))\right), \quad (73)$$

where $\odot$ denotes element-wise multiplication, $\mathbf{U}_q^{(\ell)}, \mathbf{U}_k^{(\ell)} \in \mathbb{R}^{d_{\text{att}} \times d_{\text{att}}}$, and $\mathbf{u}^{(\ell)} \in \mathbb{R}^{d_{\text{att}}}$. The matrices $\mathbf{U}_q^{(\ell)}$ and $\mathbf{U}_k^{(\ell)}$ are the internal query/key interaction maps inside the scorer block of Fig. 3.4; they are not additional graph layers.

Attention weights are obtained by softmax over the sender index j . Intermediate layers use a

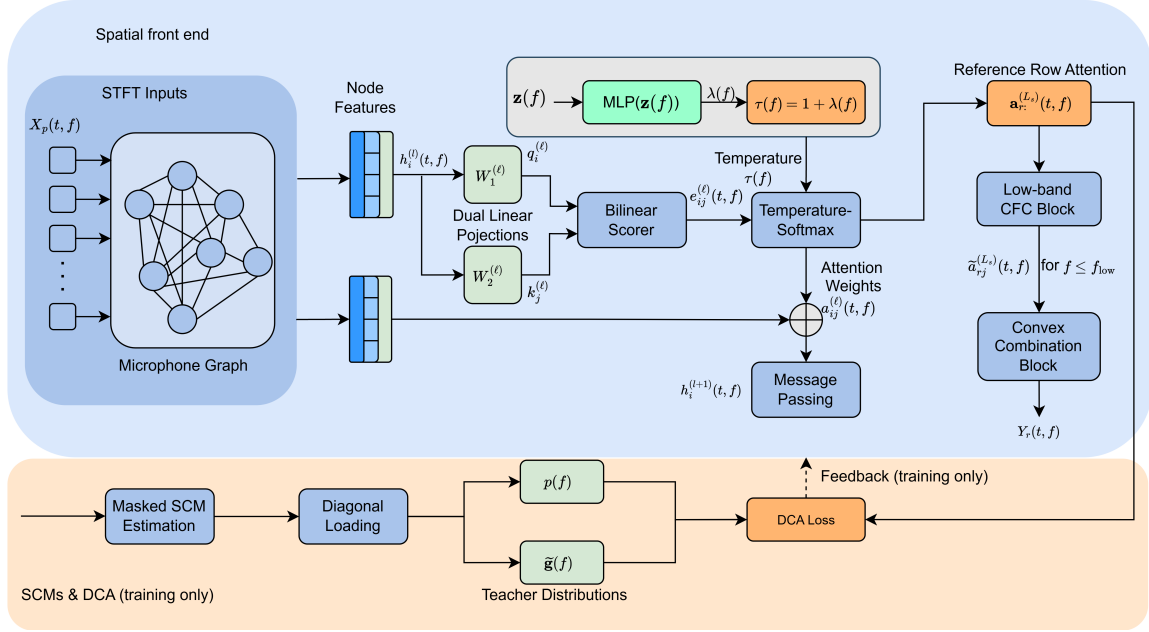


Figure 3.4: Spatial front end with stability-oriented modules. Dual-projection graph attention produces convex channel weights. In the source figure, P denotes the microphone count, the reference-row attention corresponds to the final-layer weights $a_{rj}(t, f)$, and the stabilized convex-combining weights are written in the text as $\tilde{a}_{rj}(t, f)$. The scorer block implicitly contains the internal maps $\mathbf{U}_q^{(\ell)}$ and $\mathbf{U}_k^{(\ell)}$ from Eq. (73); these are internal query/key interaction projections rather than additional graph layers. A learnable temperature stabilizes the final-layer distribution, low-band cross-frequency coupling reduces low-frequency attention jitter, and the same temperature modulates scaled diagonal loading in the training-only teacher construction.

standard softmax, whereas the final layer uses the temperature-scaled softmax defined in (75). Thus, for every receiver i and layer ℓ ,

$$a_{ij}^{(\ell)}(t, f) \geq 0, \quad \sum_{j=1}^M a_{ij}^{(\ell)}(t, f) = 1.$$

Given $a_{ij}^{(\ell)}(t, f)$, node embeddings are updated by attention-weighted aggregation

$$\mathbf{h}_i^{(\ell+1)}(t, f) = \text{PReLU} \left(\sum_{j=1}^M a_{ij}^{(\ell)}(t, f) \mathbf{W}_v^{(\ell)} \mathbf{h}_j^{(\ell)}(t, f) \right), \quad (74)$$

where $\mathbf{W}_v^{(\ell)} \in \mathbb{R}^{D \times D}$ is the value-transformation matrix that converts each sender-node embedding into a message vector before attention-weighted aggregation. The spatial reference weights used in

(69) are taken from the reference row of the final layer, i.e., $a_{rj}(t, f) \triangleq a_{rj}^{(L_s-1)}(t, f)$, and may be further stabilized by low-band CFC to produce $\tilde{a}_{rj}(t, f)$.

Learnable temperature and low-band cross-frequency coupling

Final-layer temperature. We introduce a learnable per-frequency temperature $\tau(f) \in [1, \tau_{\max}]$ and apply it only to the final spatial layer. This avoids circular dependence because $\tau(f)$ is predicted from summary statistics of the final-layer logits before temperature scaling, while all earlier-layer attentions are computed without $\tau(f)$. Concretely,

$$a_{ij}^{(\ell)}(t, f) = \begin{cases} \frac{\exp(e_{ij}^{(\ell)}(t, f))}{\sum_{\kappa=1}^M \exp(e_{i\kappa}^{(\ell)}(t, f))}, & \ell = 0, 1, \dots, L_s - 2, \\ \frac{\exp(e_{ij}^{(\ell)}(t, f)/\tau(f))}{\sum_{\kappa=1}^M \exp(e_{i\kappa}^{(\ell)}(t, f)/\tau(f))}, & \ell = L_s - 1. \end{cases} \quad (75)$$

Larger $\tau(f)$ smooths the final-layer simplex distribution and therefore reduces near-hard selection when logits are unreliable.

To predict $\tau(f)$ from the current chunk, we compute per-frequency summary statistics $\mathbf{z}(f) \in \mathbb{R}^3$:

$$\mathbf{z}(f) = [z_{\text{eng}}(f), z_{\text{disp}}(f), z_{\text{ent}}(f)]^T, \quad (76)$$

with a small constant $\varepsilon_z > 0$ and

$$\begin{aligned} z_{\text{eng}}(f) &\triangleq \frac{1}{T_{\text{ck}}} \sum_{t=1}^{T_{\text{ck}}} \log(\varepsilon_z + \|\mathbf{X}(t, f)\|_2^2), \\ z_{\text{disp}}(f) &\triangleq \frac{1}{T_{\text{ck}}} \sum_{t=1}^{T_{\text{ck}}} \text{std}(\mathbf{e}_{r\cdot}(t, f)), \\ z_{\text{ent}}(f) &\triangleq \frac{1}{T_{\text{ck}}} \sum_{t=1}^{T_{\text{ck}}} H(\text{softmax}(\mathbf{e}_{r\cdot}(t, f))). \end{aligned} \quad (77)$$

Here $\mathbf{X}(t, f) = [X_1(t, f), \dots, X_M(t, f)]^T$, $\mathbf{e}_{r\cdot}(t, f) \triangleq [e_{r1}^{(L_s-1)}(t, f), \dots, e_{rM}^{(L_s-1)}(t, f)]^T$ denotes the reference-row logits at the final spatial layer before temperature scaling, $\text{std}(\cdot)$ is the standard deviation over the channel index, and $H(\mathbf{p}) \triangleq -\sum_j p_j \log p_j$ is Shannon entropy. Intuitively,

$z_{\text{disp}}(f)$ and $z_{\text{ent}}(f)$ summarize how peaked or uncertain the raw logits are, whereas $z_{\text{eng}}(f)$ provides a coarse energy context for that frequency.

We parameterize

$$\tau(f) = \text{clip}_{[1, \tau_{\max}]} \left(1 + \text{softplus} \left(\mathbf{v}_\tau^\top \text{LeakyReLU}(\mathbf{U}_\tau \mathbf{z}(f)) \right) \right), \quad (78)$$

where $\mathbf{U}_\tau \in \mathbb{R}^{d_\tau \times 3}$ and $\mathbf{v}_\tau \in \mathbb{R}^{d_\tau}$ are learnable parameters, d_τ is a small hidden width, $\text{softplus}(x) = \log(1 + e^x)$, and $\text{clip}_{[a, b]}(\cdot)$ clamps its argument to the interval $[a, b]$.

Low-band cross-frequency coupling. Even with temperature control, attention computed independently at each frequency can remain unstable at low frequencies, where phase-derived cues are often less reliable. We therefore smooth the reference-row weights across adjacent frequency bins for $f \leq f_{\text{low}}$.

Let $\mathbf{a}_{r:}(t, f) \triangleq [a_{r1}(t, f), \dots, a_{rM}(t, f)]^\top \in \mathbb{R}^M$ denote the reference-row simplex weights after temperature scaling. For $f \leq f_{\text{low}}$, we define

$$\tilde{\mathbf{a}}_{r:}(t, f) = (1 - \gamma_f) \mathbf{a}_{r:}(t, f) + \gamma_f \sum_{\Delta \in \{-1, 0, 1\}: f + \Delta \in \mathcal{F}_+} \bar{w}_\Delta(f) \mathbf{a}_{r:}(t, f + \Delta), \quad (79)$$

and set $\tilde{\mathbf{a}}_{r:}(t, f) = \mathbf{a}_{r:}(t, f)$ for $f > f_{\text{low}}$. Here f_{low} is a frequency-bin index, $\gamma_f \in [0, \gamma_{\max}]$ is a frequency-dependent coupling gate, and $\bar{w}_\Delta(f) \geq 0$ are boundary-normalized kernel weights satisfying $\sum_{\Delta} \bar{w}_\Delta(f) = 1$ over the valid neighboring bins. Thus Eq. (79) averages only existing adjacent bins near boundaries and leaves the vector on the probability simplex.

We use a symmetric kernel $w_{-1} = w_{+1} = \frac{1}{4}$ and $w_0 = \frac{1}{2}$, with boundary normalization

$$\bar{w}_\Delta(f) = \frac{w_\Delta \mathbb{I}[f + \Delta \in \mathcal{F}_+]}{\sum_{\Delta' \in \{-1, 0, 1\}} w_{\Delta'} \mathbb{I}[f + \Delta' \in \mathcal{F}_+]}, \quad \Delta \in \{-1, 0, 1\}, \quad (80)$$

where $\mathbb{I}[\cdot]$ is the indicator function. To emphasize stabilization near DC, we set

$$\gamma_f = \gamma_{\max} \max \left(0, 1 - \frac{f}{f_{\text{low}}} \right). \quad (81)$$

Because (79) is a convex combination of simplex vectors, $\tilde{\mathbf{a}}_{r:}(t, f)$ remains nonnegative and sums

to one; its components $\tilde{a}_{rj}(t, f)$ can therefore be used directly in (69).

Scaled diagonal loading for training-only teachers. Teacher construction in the next subsection uses an estimated disturbance SCM $\hat{\Phi}_{nn}(f)$ during training. To stabilize the teacher-side linear solve when $\tau(f)$ indicates uncertainty, we apply diagonal loading scaled by $\tau(f)$. Let

$$\mu_n(f) \triangleq \frac{\text{Re}\{\text{tr}(\hat{\Phi}_{nn}(f))\}}{M}, \quad \delta(f) \triangleq \eta_{\text{load}} \text{stopgrad}((\tau(f) - 1) \mu_n(f)) \geq 0,$$

where $\eta_{\text{load}} > 0$ is a scalar hyperparameter and $\text{stopgrad}(\cdot)$ blocks gradients through the teacher-construction path. The scalar $\mu_n(f)$ is the real-valued average disturbance power per microphone, so $\delta(f)\mathbf{I}$ has the same physical scale as the SCM. Because $\tau(f) \geq 1$, the factor $\tau(f) - 1$ increases the loading only when the final-layer attention is judged uncertain. Applying the stop-gradient to the full loading scale prevents the network from reducing the teacher loss by modifying $\tau(f)$ or the SCM-derived power through this stabilization path. The loaded SCM is

$$\hat{\Phi}_{nn}^{\text{load}}(f) = \hat{\Phi}_{nn}(f) + \delta(f) \mathbf{I}, \quad (82)$$

where $\mathbf{I}$ is the $M \times M$ identity matrix.

Training-time MVDR-inspired teacher alignment

We use beamforming-inspired structure only as training guidance, without adding inference-time SCM computation. During training, we construct per-frequency teacher distributions from masked SCM estimates and align the predicted convex weights to these teachers via KL regularization. All SCM-derived quantities in this subsection are detached from the computational graph by a stop-gradient operator and are not computed at inference. Because the training data in this section are simulated, the speech-dominance mask $\rho(t, f)$ used for teacher construction is obtained from oracle target and disturbance energies rather than from an additional learned mask estimator.

Let $\rho(t, f) \in [0, 1]$ be the oracle speech-dominance mask constructed from target and disturbance energies. Over the current chunk, we compute mask-weighted SCMs

$$\hat{\Phi}_{ss}(f) = \frac{\sum_{t=1}^{T_{\text{ck}}} \rho(t, f) \mathbf{X}(t, f) \mathbf{X}^H(t, f)}{\sum_{t=1}^{T_{\text{ck}}} \rho(t, f) + \varepsilon_{\text{scm}}}, \quad \hat{\Phi}_{nn}(f) = \frac{\sum_{t=1}^{T_{\text{ck}}} (1 - \rho(t, f)) \mathbf{X}(t, f) \mathbf{X}^H(t, f)}{\sum_{t=1}^{T_{\text{ck}}} (1 - \rho(t, f)) + \varepsilon_{\text{scm}}}, \quad (83)$$

where $\varepsilon_{\text{scm}} > 0$ avoids division by zero.

We extract a steering or relative transfer function (RTF) proxy as the principal eigenvector of $\hat{\Phi}_{ss}(f)$ and apply a numerically safe reference normalization:

$$\tilde{\mathbf{d}}(f) = \text{eig}_{\max}(\hat{\Phi}_{ss}(f)), \quad \hat{\mathbf{d}}(f) = \tilde{\mathbf{d}}(f) \frac{\tilde{d}_r^*(f)}{\max(|\tilde{d}_r(f)|^2, \varepsilon_d)}, \quad (84)$$

where $\varepsilon_d > 0$ is small and $\tilde{d}_r(f)$ is the r -th entry of $\tilde{\mathbf{d}}(f)$. We then compute an MVDR-numerator proxy by solving

$$\tilde{\mathbf{v}}(f) = (\hat{\Phi}_{nn}^{\text{load}}(f))^{-1} \hat{\mathbf{d}}(f), \quad (85)$$

implemented numerically as the linear solve $\hat{\Phi}_{nn}^{\text{load}}(f) \tilde{\mathbf{v}}(f) = \hat{\mathbf{d}}(f)$.

Because inference-time spatial weights are constrained to be real and convex, we align magnitude profiles rather than complex MVDR coefficients. We therefore define simplex-valued teacher distributions

$$\mathbf{p}(f) = \frac{|\tilde{\mathbf{v}}(f)| + \varepsilon_{\text{teach}}}{\mathbf{1}^\top (|\tilde{\mathbf{v}}(f)| + \varepsilon_{\text{teach}})}, \quad \tilde{\mathbf{g}}(f) = \frac{|\hat{\mathbf{d}}(f)| + \varepsilon_{\text{teach}}}{\mathbf{1}^\top (|\hat{\mathbf{d}}(f)| + \varepsilon_{\text{teach}})}, \quad (86)$$

where $|\cdot|$ is elementwise magnitude, $\mathbf{1} \in \mathbb{R}^M$ is the all-ones vector, and $\varepsilon_{\text{teach}} > 0$ prevents numerical zeros.

Since the teachers in (86) are estimated from chunk-level SCMs, we align them to the chunk-averaged predicted profile rather than forcing every frame to match the same teacher. Define

$$\bar{\mathbf{a}}_{r:}(f) \triangleq \frac{1}{T_{\text{ck}}} \sum_{t=1}^{T_{\text{ck}}} \tilde{\mathbf{a}}_{r:}(t, f), \quad (87)$$

where $\tilde{\mathbf{a}}_{r:}(t, f)$ is the stabilized reference-row simplex vector after temperature scaling and low-band CFC. Using the Kullback–Leibler divergence $\text{KL}(\mathbf{p} \parallel \mathbf{q}) \triangleq \sum_j p_j \log \frac{p_j}{q_j}$, the directional consistency alignment (DCA) regularizer is

$$\mathcal{L}_{\text{DCA}} = \lambda_{\text{dir}} \mathbb{E}_f [\text{KL}(\mathbf{p}(f) \parallel \tilde{\mathbf{a}}_{r:}(f))] + \lambda_{\text{rtf}} \mathbb{E}_f [\text{KL}(\tilde{\mathbf{g}}(f) \parallel \tilde{\mathbf{a}}_{r:}(f))], \quad (88)$$

where $\lambda_{\text{dir}}, \lambda_{\text{rtf}} \geq 0$ are scalar weights and $\mathbb{E}_f$ denotes simple averaging over the frequency bins in the current chunk.

Temporal refinement, masking, and total loss

After forming the spatial reference $Y_r(t, f)$ in (69), we use the same temporal refinement and complex masking structure as in Section 3.3, yielding

$$\hat{S}(t, f) = \widehat{M}(t, f) Y_r(t, f). \quad (89)$$

Let $S(t, f)$ denote the clean target spectrum under the protocol-defined target (Chapter 2, Section 2.2.1). Unless stated otherwise, $\mathbb{E}_{t,f}$ denotes simple averaging over the TF bins in the current training chunk.

Phase-constrained magnitude loss (PCM). Define the phase-rotated estimate

$$\tilde{S}(t, f) = \hat{S}(t, f) \exp(-j \angle S(t, f)), \quad (90)$$

so that the clean target becomes real and nonnegative in the rotated frame. This rotation separates the estimate into an in-phase component, which should match the target magnitude, and an orthogonal component, which represents phase error relative to the target. It therefore penalizes phase inconsistency without directly regressing a wrapped phase angle. We then use

$$\mathcal{L}_{\text{pcm}} = \mathbb{E}_{t,f} \left[(|S(t, f)| - \Re\{\tilde{S}(t, f)\})^2 + \beta_{\perp} (\Im\{\tilde{S}(t, f)\})^2 \right], \quad (91)$$

where $\beta_{\perp} \in [0, 1]$ controls the penalty on the component orthogonal to the clean phase.

Time-domain reconstruction loss. Let $s[n]$ and $\hat{s}[n]$ be the clean and enhanced waveforms for the current training segment of length N_{samp} samples. We use the waveform-domain ℓ_1 loss

$$\mathcal{L}_{\text{time}} = \frac{1}{N_{\text{samp}}} \sum_{n=1}^{N_{\text{samp}}} |\hat{s}[n] - s[n]|. \quad (92)$$

The reconstruction term is then

$$\mathcal{L}_{\text{rec}} = \alpha_{\text{rec}} \mathcal{L}_{\text{pcm}} + (1 - \alpha_{\text{rec}}) \mathcal{L}_{\text{time}}, \quad \alpha_{\text{rec}} \in [0, 1]. \quad (93)$$

In the experiments of this section, α_{rec} is a fixed validation-selected hyperparameter, not a learned parameter; we use $\alpha_{\text{rec}} = 0.7$ as stated in Section 3.4.2.

STFT projection consistency (PSC). Let $\mathcal{T}$ and $\mathcal{T}^{-1}$ denote the STFT and iSTFT operators with identical window and hop parameters. Define the consistency operator

$$\mathcal{P}(X) \triangleq \mathcal{T}(\mathcal{T}^{-1}(X)), \quad (94)$$

which maps an arbitrary complex TF array to the STFT of its overlap-add reconstruction and thus projects it toward the set of STFT-consistent representations under the chosen analysis or synthesis configuration. We penalize the projection error

$$\mathcal{L}_{\text{psc}} = \mathbb{E}_{t,f} \left[|\hat{S}(t, f) - \mathcal{P}(\hat{S})(t, f)| \right]. \quad (95)$$

Overall objective. The full training objective is

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda_{\text{psc}} \mathcal{L}_{\text{psc}} + \mathcal{L}_{\text{DCA}}, \quad (96)$$

where $\lambda_{\text{psc}} > 0$ is a scalar weight. Both $\mathcal{L}_{\text{psc}}$ and $\mathcal{L}_{\text{DCA}}$ are used only to update the model parameters during training. After training, the objective terms, SCM estimation, and teacher construction are removed; inference therefore uses only the learned spatial front end, temporal encoder, and mask decoder, with no extra operations from these losses.

3.4.2 Experimental Setup

Datasets. We evaluate on two simulated (spatialized) corpora and one real multichannel corpus. The simulated sets are used for controlled analysis of stability under SNR and reverberation mismatch, whereas LibriCSS is included to test transfer to real acoustics and downstream ASR. All signals are sampled at 16 kHz. For the spatialized corpora, multichannel mixtures are rendered by convolving clean speech and noise with multichannel RIRs generated using the image-source method in `pyroomacoustics` [35]. The splits use disjoint pools of rooms and array poses.

Spatialized WSJCAM0. Clean utterances are drawn from WSJCAM0 [73]. We simulate 4-channel mixtures using a circular array with radius 10 cm. SNR at channel 1 is sampled uniformly from $[5, 25]$ dB. This corpus is used because it provides a reverberation-dominant controlled setting in which the effect of the stabilizers can be measured against clean references.

Spatialized DNS Challenge. Speech and noise are sourced from DNS Challenge 2020 [40]. SNR is sampled from $[-10, 10]$ dB and reverberation time RT_{60} from $[0.2, 1.2]$ s. This corpus complements WSJCAM0 by emphasizing stronger and more nonstationary additive interference.

LibriCSS (real). LibriCSS [75] is recorded with a 7-channel circular array in real rooms and is grouped by overlap ratio (OS/OL/10–40%). It is included because real meeting recordings are necessary to test whether covariance-free spatial inference remains stable when simulation assumptions break.

Implementation details. We compute the STFT using a 32 ms analysis window ($L = 512$ samples), an 8 ms hop ($R = 128$ samples), and a 512-point FFT, yielding 257 nonnegative-frequency bins. The same configuration is used for iSTFT. Training uses random 4 s segments. Unless otherwise stated, channel 1 is used as the reference row, i.e., $r = 1$.

The spatial front end uses node width $D = 128$, spatial depth $L_s = 2$, attention width $d_{\text{att}} = 64$, temperature hidden width $d_\tau = 16$, and temperature clamp $\tau_{\text{max}} = 4.0$. Low-band CFC uses a cutoff frequency of 1.5 kHz, i.e., f_{low} is the corresponding frequency-bin index; we set $\gamma_{\text{max}} = 0.5$ and use the kernel $[0.25, 0.5, 0.25]$. The temporal encoder and decoder use the same architecture as the method in Section 3.3, with temporal embedding width $D_T = 128$, $L_t = 6$, local attention half-window $w = 4$, depthwise-convolution kernel size $k = 15$, and low-rank projection dimension

$r_{\text{LR}} = 64$ in the Lite configuration unless otherwise stated.

Training details. On the simulated training sets, the speech-dominance mask $\rho(t, f)$ in Eq. (83) is constructed from oracle target and disturbance energies at channel 1:

$$\rho(t, f) = \frac{|S(t, f)|^2}{|S(t, f)|^2 + |V_1(t, f)|^2 + 10^{-8}}.$$

The model is trained for 100 epochs with batch size 8 using AdamW. We use peak learning rate 4×10^{-4} , weight decay 10^{-4} , 5-epoch linear warmup, and cosine decay to 10^{-6} . Gradient clipping uses maximum norm 5.0. Mixed-precision training and exponential moving average (EMA) of model weights are enabled, with EMA decay 0.999. These settings are the same across WSJCAM0 and DNS Challenge 2020-based simulated training unless otherwise stated.

The loss and regularization hyperparameters are fixed to $\alpha_{\text{rec}} = 0.7$, $\beta_{\perp} = 0.3$, $\lambda_{\text{psc}} = 0.05$, $\lambda_{\text{dir}} = 0.05$, $\lambda_{\text{rtf}} = 0.02$, and $\eta_{\text{load}} = 0.1$. The numerical-stability constants are set to $\varepsilon_z = 10^{-8}$, $\varepsilon_{\text{scm}} = 10^{-6}$, $\varepsilon_d = 10^{-8}$, and $\varepsilon_{\text{teach}} = 10^{-6}$. These values are kept fixed across all experiments in this section unless a specific ablation states otherwise. For *Proposed-LITE*, the main capacity-related settings are the spatial tuple $(D, L_s, d_{\text{att}}) = (128, 2, 64)$ and the temporal tuple $(D_T, L_t, w, k, r_{\text{LR}}) = (128, 6, 4, 15, 64)$. The larger *Proposed-STD* keeps the same spatial, temperature, CFC, and loss settings, but increases temporal-refinement capacity as described in Section 3.4.3. For external baselines reported in the comparison tables, architectures and hyperparameters follow the cited publications or official implementations and are not re-tuned in this thesis.

3.4.3 Experimental Results and Analysis

Overall results on spatialized corpora

Table 3.5 summarizes results on two complementary spatialized benchmarks. WSJCAM0 is reverberation-dominant with moderate noise, whereas DNS-2020 contains stronger and more non-stationary interferers at lower SNR. This pairing evaluates both dereverberation- and denoising-oriented conditions under consistent multichannel rendering.

The WSJCAM0 block of Table 3.5 shows that both proposed variants outperform the earlier

Table 3.5: Performance comparison on spatialized WSJCAM0 and DNS-2020.

Model	WSJCAM0			DNS-2020			Params (M)	GMAC/s
	SI-SDR (dB)	PESQ	STOI	SI-SDR (dB)	PESQ	STOI		
Unprocessed	-4.0	1.93	0.707	-7.8	1.38	0.593	–	–
FasNet-TAC [63]	8.2	2.93	0.947	4.7	2.26	0.865	2.9	55.3
TPARN [65]	10.4	3.43	0.969	8.4	2.75	0.919	3.2	71.8
ADCN [64]	12.0	3.42	0.973	7.8	2.84	0.923	9.3	72.8
ADCN-TPARN [64]	13.8	3.64	0.980	10.6	3.08	0.936	40.0	360.0
DeFT-AN [24]	15.7	3.63	0.981	9.9	3.01	0.924	2.7	95.6
DeFTAN-II [25]	18.4	3.92	0.987	12.0	3.35	0.953	4.0	64.5
Proposed-LITE	17.1	3.76	0.985	11.2	3.15	0.934	3.3	47.2
Proposed-STD	18.8	3.96	0.989	12.5	3.41	0.957	7.1	86.0

graph-attention model and most neural baselines. *Proposed-STD* achieves the best SI-SDR, PESQ, and STOI among the compared systems. *Proposed-LITE* attains slightly lower scores while remaining compact in parameters and computation. Compared with DeFT-AN, *Proposed-LITE* improves SI-SDR from 15.7 to 17.1 dB while reducing computation from 95.6 to 47.2 GMAC/s. Compared with ADCN-TPARN, it improves SI-SDR from 13.8 to 17.1 dB with substantially fewer parameters and lower computation. These results are consistent with the intended role of the stabilized spatial front end: forming a bounded spatial reference that reduces the burden on subsequent temporal and complex refinement.

In the DNS-2020 block of Table 3.5, absolute scores decrease for all systems due to stronger and rapidly varying noise. Nevertheless, *Proposed-STD* remains competitive and slightly outperforms DeFTAN-II across SI-SDR, PESQ, and STOI. *Proposed-LITE* also improves over DeFT-AN while using substantially lower computation. The smaller margins relative to WSJCAM0 are consistent with the increased difficulty of enhancement at very low SNR and under highly nonstationary interferers.

Results on real recordings

We further evaluate generalization on real-world recordings using LibriCSS. This test set differs from the simulated training conditions because it contains real room responses, device characteristics, and conversational dynamics.

Table 3.6 shows that the proposed models consistently reduce WER across all overlap conditions. With the ASR back end unchanged, *Proposed-STD* improves over the MVDR baseline in every subset, for example from 8.4% to 7.7% on OS and from 21.7% to 20.4% on 40% overlap. A plausible reason is that covariance-based beamforming can be sensitive to mask or SCM estimation errors under real-room mismatch and rapidly changing interference statistics, whereas the proposed inference-time spatial stage avoids SCM estimation entirely and enforces a gain-bounded convex mixing rule at every TF bin.

These results are consistent with the design of the present stabilized variant: inference uses bounded convex mixing with frame-level updates, while SCM-derived quantities are used only as training-time teacher signals. The learnable temperature and low-band coupling further reduce weight variability in frequency regions where phase cues are typically less reliable.

Table 3.6: WER (%) comparison on LibriCSS across overlap ratios.

System	OS	OL	10%	20%	30%	40%
No separation	11.8	11.7	18.8	27.2	35.6	43.3
Mask (7ch)	12.0	11.6	15.6	20.2	25.6	29.4
MVDR (7ch)	8.4	8.3	11.6	15.8	18.7	21.7
FasNet-TAC	12.5	12.4	16.7	21.5	27.1	31.8
TPARN	10.1	10.0	13.9	18.3	22.7	26.8
Proposed-LITE	8.0	8.0	11.0	15.2	18.0	21.0
Proposed-STD	7.7	7.8	10.6	14.7	17.5	20.4

Ablation studies

Table 3.7 quantifies the contribution of each core component under the WSJCAM0 setting. All ablation variants are derived from *Proposed-LITE* and retrained under the same optimization protocol to ensure a fair comparison.

Main architectural components. To assess the contribution of the spatial front end, we remove the microphone-graph attention module and directly forward the raw reference-channel spectrum $X_r(t, f)$ to the subsequent stages; the spatial-embedding input is set to zeros so that the downstream interface remains unchanged. This causes a large SI-SDR drop, confirming that covariance-free convex spatial mixing is the dominant mechanism for exploiting multichannel diversity in reverberant mixtures.

Table 3.7: Ablation studies on WSJCAM0 (4-ch).

Variant	SI-SDR (dB)	PESQ	STOI	Params (M)	GMAC/s
w/o spatial combiner	12.1	3.40	0.959	2.2	31.3
w/o temporal encoder	11.2	3.15	0.926	2.3	29.9
w/o complex mask (cRM)	14.7	3.57	0.976	3.1	44.6
w/o dual projection	15.6	3.62	0.979	3.2	46.8
w/o temperature & teacher loading	16.7	3.73	0.984	3.3	47.2
w/o low-band CFC	16.8	3.74	0.984	3.3	47.1
Proposed-LITE	17.1	3.76	0.985	3.3	47.2

To assess the role of temporal refinement, we remove the localized Conformer blocks while keeping the spatial front end and the decoder backbone unchanged. In this variant, the decoder is driven only by spatially combined reference features, i.e., without temporal-encoder representations; implementation-wise, the first decoder projection is resized to match the reduced input dimension. The degradation indicates that the spatial front end provides a meaningful coarse reference, while bounded-context temporal modeling remains essential to suppress residual distortions and recover spectro-temporal detail.

Spatial and refinement mechanisms. To isolate the effect of complex refinement, we disable the complex ratio-mask stage by setting $\widehat{M}(t, f) \equiv 1$, so that the enhanced spectrum equals the convexly combined reference $Y_r(t, f)$. The resulting performance drop supports the conclusion that explicit complex refinement remains beneficial even after stable convex spatial combining.

To evaluate the benefit of receiver–sender asymmetry in attention, we simplify the dual-projection attention to a single shared projection for both query and key computation, while keeping the remaining attention and message-passing design unchanged. The associated loss suggests that modeling directional asymmetry improves the reliability of learned inter-channel relations for reference-conditioned spatial weighting.

Stability modules. Finally, we examine the proposed stabilizers. For *w/o temperature & teacher loading*, we use a fixed temperature and disable the associated scaled diagonal loading in teacher construction. For *w/o low-band CFC*, we disable low-band cross-frequency coupling so that attention weights are computed independently at each frequency. Both ablations yield consistent but smaller degradations, indicating that these components primarily improve robustness by reducing

attention jitter in frequency regions where spatial cues are less reliable.

Effect of training regularizers

Table 3.8 isolates the effect of training-time regularizers on WSJCAM0 while keeping the architecture and inference computation unchanged. The absolute differences are modest, so this table should be read as a sensitivity check rather than as evidence that either regularizer alone is the main source of improvement. The trends are nevertheless consistent with the intended roles of the terms: DCA encourages the learned convex weights to follow MVDR-inspired magnitude profiles, whereas PSC discourages STFT-inconsistent artifacts introduced by masking and temporal refinement. Both regularizers are disabled after training and therefore do not add inference operations.

Table 3.8: Sensitivity to training-time DCA and PSC regularizers.

Variant	SI-SDR (dB)	PESQ	STOI
w/o DCA (no λ_{dir} , no λ_{rtf})	16.6	3.74	0.984
Directional alignment only	16.9	3.75	0.984
Steering-magnitude prior only	16.8	3.74	0.984
w/o PSC	16.9	3.75	0.984
Proposed-LITE (full)	17.1	3.76	0.985

Temporal window and kernel

Table 3.9 studies how the local self-attention half-window w and the depthwise convolution kernel size k affect enhancement quality and computation on WSJCAM0. A smaller window limits localized self-attention from capturing mid-range temporal dependencies such as reverberation tails and slowly varying interference, leading to a consistent drop in SI-SDR. Increasing w yields a modest improvement, but the gain saturates. Adjusting k has a smaller effect around moderate kernel sizes.

Robustness to geometry and channels

Table 3.10 evaluates robustness under geometry perturbations and channel-failure conditions on WSJCAM0. The proposed model retains most of its performance under array-radius change,

Table 3.9: Effect of attention window w and convolution kernel k .

Setting	SI-SDR (dB)	PESQ	STOI	Params (M)	GMAC/s
$w = 2, k = 15$	16.7	3.74	0.984	3.3	46.6
Default ($w = 4, k = 15$)	17.1	3.76	0.985	3.3	47.2
$w = 6, k = 15$	17.3	3.77	0.985	3.3	47.8
$w = 4, k = 11$	16.9	3.75	0.984	3.3	46.1
$w = 4, k = 19$	17.2	3.76	0.985	3.3	48.3

yaw perturbation, channel permutation, and single-channel dropout, whereas the baseline degrades substantially.

Table 3.10: SI-SDR (dB) under geometry perturbations and channel failures.

Perturbation condition	TPARN	Proposed-LITE
Array radius $10 \rightarrow 8$ cm	9.7	16.7
Yaw perturbation $\pm 15^\circ$	9.9	16.8
One channel dropped	8.8	16.3
Random channel permutation	9.5	16.6

Capacity sweep on WSJCAM0

Table 3.11 examines how performance changes as the model is scaled from *Proposed-LITE* toward a larger configuration. Here D_T denotes the temporal-module embedding width, L_t the number of temporal blocks, and r_{LR} the low-rank projection dimension used in the temporal refinement stack. The results suggest that once a stable spatial reference is formed by the convex spatial front end, the dominant remaining bottleneck lies in spectro-temporal refinement. Increasing w expands the local attention context, increasing D_T and L_t increases temporal modeling capacity, and increasing r_{LR} raises the rank of the low-rank projections; each change can improve quality but also increases computation.

Table 3.11: Capacity sweep on WSJCAM0 (4-ch).

Config	SI-SDR (dB)	PESQ	STOI	Params (M)	GMAC/s
Proposed-LITE	17.1	3.76	0.985	3.3	47.2
+ $w = 6$	17.3	3.77	0.985	3.3	47.8
$D_T = 160, L_t = 8, r_{LR} = 80$	18.6	3.85	0.988	7.1	86.0
Proposed-STD	18.8	3.96	0.989	7.1	86.0

Implications for deployment and connection to Chapter 4. Across the above analyses, the key effect of temperature control and low-band coupling is not to change the form of the spatial operator, but to make its per-TF weights more reliable under uncertainty, including low SNR, strong reverberation, and device or geometry mismatch. This stabilized explicit convex interface therefore provides a convenient anchor point for the deployability-oriented developments in Chapter 4, where we further inject physics-guided stabilization into explicit TF-domain spatial filtering and address objective mismatch with portable analytic training objectives.

3.5 Conclusion

This chapter addressed a practical gap in multichannel speech enhancement: strong neural models can still be brittle to microphone ordering, channel-count variation, and array-geometry mismatch, partly because spatial fusion is often implicit and difficult to inspect. To close this gap, we developed a graph-based line of topology-robust spatial processing that progressively moves from latent feature mixing to an explicit and gain-bounded spatial interface.

Concretely, we first validated that complex-domain graph message passing provides topology-aware spatial feature fusion while preserving phase-sensitive cues, improving enhancement on both real recordings and simulated spatial scenes. We then introduced an explicit per-time-frequency convex spatial weighting interface predicted on a microphone graph, yielding an interpretable “filter-and-sum” style spatial reference stream that remains compatible with downstream temporal and spectral refinement. Finally, we strengthened stability and generalization without relying on inference-time spatial covariance estimation by incorporating lightweight stabilizers (a learnable per-frequency temperature and low-band cross-frequency coupling) together with a training-only MVDR-inspired teacher alignment. Across both simulated and real-world evaluations, the results support a consistent conclusion: explicitly constrained convex spatial weights provide a reliable and interpretable interface between multichannel acoustics and neural refinement, while improving robustness to topology changes and array-geometry perturbations.

With a topology-robust and explicit convex spatial interface established, the central challenge shifts from “how to exploit multichannel structure under channel or geometry variation” to “how

to remain reliable under deployment-grade mismatch and objective mismatch”. Realistic systems must tolerate small array perturbations, unstable phase cues at very low SNR, and the tendency of strong suppression to introduce perceptually salient artifacts even when distortion-oriented metrics appear favorable.

Chapter 4 builds on the explicit spatial-filtering viewpoint developed here and moves toward a more deployable blueprint. It strengthens robustness under practical mismatch by injecting physics-guided stabilization into explicit TF-domain spatial filtering, retains lightweight residual refinement to recover attenuated speech details, and further addresses objective mismatch through portable analytic training objectives that better align optimization with perceptual and utility-driven evaluation.

Chapter 4

Physics-Informed Lightweight Deep Beamforming with Residual Refinement and Perceptual Objectives

4.1 Introduction

Chapter 3 established topology-robust multichannel processing by explicitly modeling inter-channel relations and by exposing an inspectable, gain-bounded convex spatial interface. This chapter takes the next step toward a deployable multichannel enhancement front end. For deployment, the spatial stage should remain explicit and implementation-friendly, remain stable under realistic mismatch, and support training objectives that better reflect perceptual and utility-driven evaluation.

These requirements expose three recurring gaps. First, small array perturbations (device-to-device tolerances, mounting offsets, or minor displacement during use) can alter relative propagation delays and inter-channel phase relations, thereby destabilizing coherent spatial filtering. Because this sensitivity becomes more pronounced at higher frequencies, even modest geometric mismatch can lead to non-negligible phase inconsistency and reduced robustness in deployable multichannel systems. Second, at extremely low SNR, spatial cues, especially phase-derived cues, become unreliable. Both covariance-driven beamforming and data-driven spatial filtering may then drift

toward overly aggressive suppression, creating spectral holes or temporal fluctuations. Third, objective mismatch remains a practical bottleneck: fidelity-oriented losses (e.g., SI-SDR or uniformly weighted STFT reconstruction losses) do not necessarily translate into consistent gains in perceptual or intelligibility metrics such as PESQ and STOI, even when a system already performs strongly on distortion-oriented measures [52, 54, 76]. Meanwhile, learned perceptual surrogates often require auxiliary networks and introduce additional training complexity [77, 78].

To address these gaps, this chapter develops a unified, deployment-oriented blueprint: explicit TF-domain spatial filtering for interference suppression, followed by a lightweight spectral refinement stage that restores details that spatial suppression tends to attenuate, complemented by objective design that better aligns optimization with perceptual band structure. We first present Dynamic Spatial Filtering with Residual Spectral Mapping (DSF-RSM), which predicts TF-varying convex spatial weights from compact real-valued spatial features (magnitudes and inter-channel phase differences) and performs residual correction in the complex domain. We then introduce PhysMVNet, which preserves the same explicit filter-and-sum interface but improves robustness under mismatch through a physics-inspired STFT-domain regularizer and a frequency-structured residual restoration strategy, targeting regimes where deployable systems most often fail (array perturbation and extremely low SNR). Finally, we address objective mismatch with a portable, non-learned analytic objective module, the multi-band relative contrastive loss (MBRCL), which can be attached to different architectures without auxiliary assessment networks.

The chapter proceeds in three parts following this blueprint. Section 4.2 details DSF-RSM, including its methodology and evaluation on multichannel benchmarks. Section 4.3 develops PhysMVNet and analyzes robustness under extremely low SNR and microphone-array perturbations. Section 4.4 then shifts the focus from architecture to optimization and presents MBRCL, designed to improve perceptual outcomes without auxiliary learned quality models. The chapter concludes by summarizing how explicit spatial operators, stability-oriented inductive bias, and objective alignment jointly improve deployability, and transitions to Chapter 5, where the main limitation shifts from spatial robustness to long-context temporal modeling and coordinated magnitude–phase refinement.

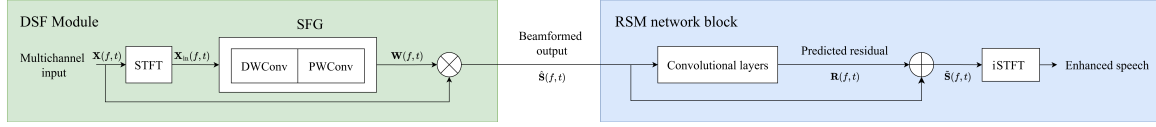


Figure 4.1: Overall architecture of DSF-RSM. The DSF module generates TF-varying convex spatial weights for a filter-and-sum spatial front end; the beamformed intermediate spectrum corresponds to $Y_{\text{beam}}(t, f)$ in Eq. (101). The RSM module predicts an additive residual in the complex (real/imaginary) domain, and the final refined spectrum corresponds to $\hat{S}(t, f)$ formed by Eqs. (106)–(107).

4.2 Dynamic Spatial Filtering with Residual Spectral Mapping

This section targets a practical but often under-emphasized constraint in multichannel speech enhancement: many applications require low latency and limited compute, yet still demand reliable spatial suppression in nonstationary and reverberant environments. DSF-RSM follows a deployability-driven design principle: the spatial stage remains an explicit, lightweight filter-and-sum front end, while a small post-filtering module restores speech details that may be attenuated by aggressive spatial suppression.

4.2.1 Proposed Method

Overall architecture

Figure 4.1 illustrates DSF-RSM. Given multichannel STFT coefficients, the dynamic spatial filtering (DSF) module predicts TF-varying convex spatial weights from compact real-valued spatial features, namely magnitudes and inter-channel phase differences. These weights are applied through a filter-and-sum operation to form the beamformed complex reference spectrum $Y_{\text{beam}}(t, f)$. Because spatial filtering alone may introduce spectral attenuation and coloration, a residual spectral mapping (RSM) network further refines $Y_{\text{beam}}(t, f)$ by predicting an additive complex residual. The enhanced waveform is then reconstructed by iSTFT.

Dynamic spatial filtering module

Let $\mathbf{X}(t, f) = [X_1(t, f), \dots, X_M(t, f)]^T \in \mathbb{C}^M$ denote the multichannel STFT vector at TF bin (t, f) . To avoid directly processing complex-valued inputs, we construct compact real-valued

spatial features from magnitudes and inter-channel phase differences (IPDs).

Magnitude and IPD features. Define the per-channel magnitude vector

$$\mathbf{x}_{\text{mag}}(t, f) \triangleq (|X_1(t, f)|, \dots, |X_M(t, f)|)^T \in \mathbb{R}^M.$$

For $1 \leq m < m' \leq M$, we define the (principal-value) IPD as

$$\Delta\phi_{m,m'}(t, f) = \angle(X_m(t, f) X_{m'}(t, f)^*). \quad (97)$$

Unlike the reference-based IPD in Chapter 2, here we use all unordered microphone pairs to avoid fixing a single reference channel inside the DSF feature extractor. Collecting all pairwise IPDs yields $\Delta\phi(t, f) \in \mathbb{R}^{\frac{M(M-1)}{2}}$. We concatenate magnitude and IPD features into a per-bin input vector

$$\mathbf{x}_{\text{in}}(t, f) = \left[\mathbf{x}_{\text{mag}}(t, f)^T, \Delta\phi(t, f)^T \right]^T \in \mathbb{R}^{D_{\text{in}}}, \quad D_{\text{in}} = M + \frac{M(M-1)}{2}.$$

Stacking $\mathbf{x}_{\text{in}}(t, f)$ over all TF bins forms a channel-first tensor $\mathbf{F}_{\text{in}} \in \mathbb{R}^{D_{\text{in}} \times T \times F}$ for convolutional processing, where T and F denote the numbers of STFT frames and nonnegative-frequency bins, respectively.

Depthwise-separable spatial filter generator. The spatial filter generator (SFG) is a lightweight CNN built with depthwise-separable convolutions: a depthwise 3×3 convolution captures local TF patterns within each feature channel, followed by a pointwise 1×1 projection that mixes channels. In our implementation, the depthwise stage uses a depth multiplier q , so that the intermediate map has $H = qD_{\text{in}}$ feature channels:

$$\mathbf{H}_{\text{dsf}} = \sigma(\text{DepthwiseConv}_{3 \times 3}(\mathbf{F}_{\text{in}})), \quad \mathbf{H}_{\text{dsf}} \in \mathbb{R}^{H \times T \times F}, \quad (98)$$

where $\sigma(\cdot)$ is a pointwise nonlinearity (e.g., ReLU). A pointwise projection then produces per-microphone logits $\mathbf{Z} \in \mathbb{R}^{M \times T \times F}$:

$$\mathbf{Z} = \text{PointwiseConv}_{1 \times 1}(\mathbf{H}_{\text{dsf}}), \quad \mathbf{Z} \in \mathbb{R}^{M \times T \times F}. \quad (99)$$

To stabilize the filter-and-sum stage, we convert logits into a convex TF weighting via a softmax over microphones:

$$a_m(t, f) = \frac{\exp(Z_m(t, f))}{\sum_{m'=1}^M \exp(Z_{m'}(t, f))}, \quad a_m(t, f) \geq 0, \quad \sum_{m=1}^M a_m(t, f) = 1. \quad (100)$$

This convex constraint guarantees gain-bounded mixing and improves robustness when spatial evidence is weak.

Filter-and-sum beamforming. Once the spatial weights are obtained, the beamformed reference at each TF bin is

$$Y_{\text{beam}}(t, f) = \sum_{m=1}^M a_m(t, f) X_m(t, f). \quad (101)$$

Equation (101) is an explicit convex spatial combiner in the sense of Eq. (22), with the advantage that the spatial behavior remains directly inspectable.

Residual spectral mapping network

Spatial filtering may attenuate speech components and introduce spectral coloration. The residual spectral mapping (RSM) module therefore refines $Y_{\text{beam}}(t, f)$ by predicting an additive complex residual. Let $S(t, f)$ denote the protocol-defined target spectrum introduced in Chapter 2; in the multichannel enhancement experiments of this chapter, this is typically the target spectrum associated with the chosen reference channel. We define the oracle residual target for supervision as

$$R(t, f) = S(t, f) - Y_{\text{beam}}(t, f). \quad (102)$$

In implementation, the network predicts $\hat{R}(t, f)$ and forms the refined spectrum by residual addition.

Since $Y_{\text{beam}}(t, f)$ is complex, we operate on its real and imaginary parts:

$$Y_{\text{beam}}(t, f) = Y_{\text{real}}(t, f) + j Y_{\text{imag}}(t, f). \quad (103)$$

The RSM network consists of three 3×3 convolutional layers (64/64/1 filters), with batch normalization and ReLU in the hidden layers and a linear output layer. We instantiate two parallel branches with the same CNN structure to predict residuals for the real and imaginary parts, respectively.

For clarity, the oracle residual targets in real and imaginary parts are

$$R_{\text{real}}(t, f) = S_{\text{real}}(t, f) - Y_{\text{real}}(t, f), \quad (104)$$

$$R_{\text{imag}}(t, f) = S_{\text{imag}}(t, f) - Y_{\text{imag}}(t, f). \quad (105)$$

Let $\hat{R}_{\text{real}}(t, f)$ and $\hat{R}_{\text{imag}}(t, f)$ denote the network predictions. The final enhanced spectrum is obtained by residual addition:

$$\hat{S}_{\text{real}}(t, f) = Y_{\text{real}}(t, f) + \hat{R}_{\text{real}}(t, f), \quad (106)$$

$$\hat{S}_{\text{imag}}(t, f) = Y_{\text{imag}}(t, f) + \hat{R}_{\text{imag}}(t, f). \quad (107)$$

Equivalently, $\hat{S}(t, f) = \hat{S}_{\text{real}}(t, f) + j \hat{S}_{\text{imag}}(t, f)$.

Training objective

We minimize a complex-domain reconstruction loss implemented on real and imaginary parts, together with an ℓ_1 penalty that encourages compact residual corrections. All summations below are over time frames and nonnegative-frequency bins:

$$\mathcal{L}_{\text{rec}} = \sum_{t, f} \left((\hat{S}_{\text{real}}(t, f) - S_{\text{real}}(t, f))^2 + (\hat{S}_{\text{imag}}(t, f) - S_{\text{imag}}(t, f))^2 \right), \quad (108)$$

$$\mathcal{L}_{\text{res}} = \sum_{t, f} \left(|\hat{R}_{\text{real}}(t, f)| + |\hat{R}_{\text{imag}}(t, f)| \right), \quad (109)$$

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda_{\text{res}} \mathcal{L}_{\text{res}}, \quad \lambda_{\text{res}} = 0.001, \quad (110)$$

where λ_{res} is selected on validation data to balance refinement capacity and overfitting.

Computational efficiency

Depthwise-separable convolutions substantially reduce computation compared with standard 2D convolutions. For a standard convolution mapping $D_{\text{in}} \rightarrow D_{\text{out}}$ with kernel size $k \times k$, the cost scales as $\mathcal{O}(D_{\text{in}}D_{\text{out}}k^2FT)$. With depth multiplier q (hence intermediate width $H = qD_{\text{in}}$), depthwise-separable convolution replaces this with a depthwise step $\mathcal{O}(qD_{\text{in}}k^2FT)$ plus a point-wise step $\mathcal{O}(qD_{\text{in}}D_{\text{out}}FT)$. Accordingly, the relative cost (depthwise-separable versus standard) is approximately

$$\frac{qD_{\text{in}}k^2 + qD_{\text{in}}D_{\text{out}}}{D_{\text{in}}D_{\text{out}}k^2} = q \left(\frac{1}{D_{\text{out}}} + \frac{1}{k^2} \right).$$

For the common case $q = 1$, this reduces to the standard ratio $\frac{1}{D_{\text{out}}} + \frac{1}{k^2}$. The lightweight RSM network further keeps overall complexity low, making DSF-RSM suitable for low-latency deployment on resource-constrained devices.

4.2.2 Experimental Setup

Datasets. We evaluate DSF-RSM primarily on the 6-channel simulated CHiME-3/4 benchmark [41, 42]. This choice is deliberate: DSF-RSM is proposed as a lightweight but explicit spatial-filtering front end, so a standardized multichannel benchmark with official clean references is needed to measure enhancement quality under realistic noise while still permitting intrusive evaluation. The CHiME benchmark is also appropriate because it combines realistic background-noise conditions with a practical tablet-array geometry, making it a useful test of whether a compact filter-and-sum design can remain competitive without heavy covariance-driven machinery. Although the broader CHiME benchmark family also contains real recordings, this section focuses on the simulated track because intrusive metrics are central to evaluating the quality–complexity trade-off of DSF-RSM.

Implementation details. We compute the STFT using a Hamming window of length 512 samples (32 ms) and hop size 256 samples (16 ms) at 16 kHz, yielding 257 nonnegative-frequency bins. The DSF module uses depthwise-separable 3×3 convolutions with depth multiplier $q = 1$, followed by

a 1×1 pointwise projection to M channel logits. The RSM branches use three 3×3 convolutional layers with 64, 64, and 1 output channels, respectively, as described above. We train for 100 epochs with Adam (initial learning rate 10^{-3}) using exponential learning-rate decay and gradient clipping (maximum norm 5). Training uses randomly cropped 4 s segments with batch size 2. Mixtures are augmented by sampling SNRs in $[-5, 10]$ dB and by using diverse noise segments; room simulations use reverberation times for a 60-dB decay (RT_{60}) in approximately 0.16–0.5 s, following the dataset protocol.

Evaluation metrics. We report NB-PESQ and WB-PESQ [54], STOI [76], SI-SDR [52], and SDR [79]. The first three quantify perceived quality and intelligibility, whereas SI-SDR and SDR capture the suppression–distortion trade-off. We also report FLOPs and parameter count because the main claim of this section includes deployment efficiency in addition to enhancement quality.

4.2.3 Results and Analysis

Speech enhancement performance

Table 4.1 reports results on 6-channel simulated CHiME-3/4. DSF-RSM performs best on most metrics, indicating that a lightweight, explicit filter-and-sum spatial stage can be highly effective when paired with residual complex refinement. Notably, DSF-RSM reaches NB-PESQ 3.51 and WB-PESQ 2.91, and achieves the highest STOI (0.988), suggesting that the RSM module helps recover speech details that are typically attenuated by aggressive spatial suppression. In SI-SDR and SDR, DSF-RSM attains 22.4/22.5 dB, reflecting strong interference suppression with limited speech distortion.

Computational complexity and model size

Table 4.2 compares computational cost and model size for methods that report these quantities. DSF-RSM achieves 22.0 G/s at 8 kHz and 43.5 G/s at 16 kHz, lower than McNet and slightly below SpatialNet, mainly due to the depthwise-separable SFG. The parameter count is 1.3 M (8 kHz) / 1.7 M (16 kHz), marginally higher than SpatialNet due to the complex-domain residual refinement, while remaining in the lightweight regime.

Table 4.1: Comparative results of DSF-RSM on 6-channel simulated CHiME-3/4.

Method	NB-PESQ	WB-PESQ	STOI	SI-SDR (dB)	SDR (dB)
Unprocessed	1.82	1.27	0.870	7.5	7.5
MNMF Beamforming [80]	–	–	0.940	–	16.2
FasNet+TAC [63]	2.53	1.91	0.944	16.0	16.6
CA Dense U-net [81]	–	2.44	–	–	18.6
NBDF [82]	2.74	2.31	0.950	15.7	16.6
FT-JNF [6]	3.20	2.61	0.967	17.6	18.3
McNet [83]	3.38	2.73	0.976	19.2	19.6
Wang et al. [4]	–	–	0.986	22.0	22.4
Oracle MVDR	2.49	1.94	0.970	17.3	17.7
SpatialNet [84]	3.49	2.88	0.983	22.1	22.3
DSF-RSM (prop.)	3.51	2.91	0.988	22.4	22.5

Table 4.2: Computational complexity and model size of DSF-RSM.

Method	8 kHz		16 kHz	
	FLOPs (G/s)	#Param (M)	FLOPs (G/s)	#Param (M)
NBDF [82]	19.5	1.2	38.9	1.2
FT-JNF [6]	19.5	1.2	38.9	1.2
McNet [83]	29.7	1.9	59.2	1.9
FasNet+TAC [63]	9.7	2.7	10.1	2.8
SpatialNet [84]	23.1	1.2	46.3	1.6
DSF-RSM (prop.)	22.0	1.3	43.5	1.7

Discussion and transition to PhysMVNet. DSF-RSM establishes a deployable baseline that is explicit, modular, and computationally compact: the DSF stage exposes a TF-dependent convex filter-and-sum interface, while RSM compensates for beamforming-induced complex spectral errors through residual correction. However, under harder mismatch regimes, most notably extremely low SNR and small array perturbations, the spatial-weight prediction stage can become the dominant source of instability, and spurious TF artifacts may emerge when phase-derived cues become unreliable.

The next section preserves the same “explicit spatial filtering + lightweight residual refinement” blueprint, but strengthens robustness by injecting a physics-inspired consistency regularizer in the STFT domain and adopting a more constrained MVDR-inspired convex parameterization, together with a frequency-structured residual strategy to improve restoration where suppression tends to be

overly aggressive.

4.3 PhysMVNet: Physics-Informed End-to-End MVDR with Helmholtz Regularization and Three-Band Residual Spectral Mapping

DSF-RSM demonstrates that an explicit “spatial filter + residual refinement” blueprint can be both effective and lightweight. This section develops PhysMVNet, which keeps the same deployable front-end structure while making the spatial filtering stage substantially more stable under mismatch, especially (i) extremely low SNR and (ii) microphone-array perturbations. PhysMVNet combines an MVDR-inspired, end-to-end learnable convex beamforming parameterization with a Helmholtz-inspired STFT-domain regularizer that discourages irregular local TF fluctuations. A three-band residual spectral mapping module further restores speech details while keeping refinement lightweight.

4.3.1 Proposed Method

Figure 4.2 summarizes PhysMVNet. Processing begins with STFT and spatio-spectral feature extraction (magnitudes and IPDs), followed by a lightweight SFG module. A learnable convex beamformer produces a beamformed complex reference spectrum $Y_{\text{beam}}(t, f)$. The Helmholtz-inspired term is a training loss branch computed directly from $Y_{\text{beam}}(t, f)$ using a fixed finite-difference stencil and a frequency-dependent scale; it does not require clean speech, microphone coordinates, or an additional inference input. Finally, a three-band RSM module predicts an additive complex correction, and the enhanced waveform is reconstructed via iSTFT.

STFT analysis and feature construction

Let $x_m[n]$ be the waveform at microphone $m \in \{1, \dots, M\}$. Applying STFT with window length L and hop size R_{hop} yields $X_m(t, f) \in \mathbb{C}$. Here R_{hop} is written explicitly to avoid collision with the residual notation used later; it corresponds to the hop-size symbol R introduced in

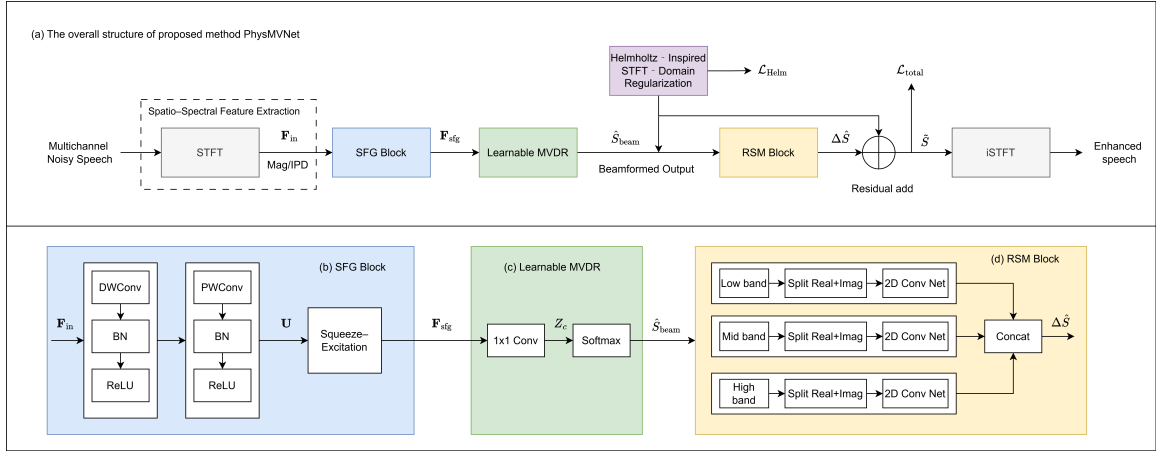


Figure 4.2: Overview of the proposed PhysMVNet architecture. (a) End-to-end pipeline from STFT through SFG, learnable beamforming, training-time Helmholtz regularization, and RSM to iSTFT; (b) Spatial Filter Generation (SFG) block; (c) learnable beamforming block; (d) Residual Spectral Mapping (RSM) block. The beamformed intermediate spectrum shown before the regularization and RSM branches corresponds to $Y_{beam}(t, f)$ in Eq. (117), the final refined output corresponds to $\hat{S}(t, f)$ in Eq. (123), and the Helmholtz branch is a training-only branch computed from Y_{beam} using the fixed stencil L and scale k_f in Eq. (118).

Chapter 2. Stacking channels forms

$$\mathbf{X}(t, f) = [X_1(t, f), \dots, X_M(t, f)]^T \in \mathbb{C}^M. \quad (111)$$

We extract magnitude features

$$\mathbf{x}_{mag}(t, f) = (|X_1(t, f)|, \dots, |X_M(t, f)|)^T \in \mathbb{R}^M,$$

and pairwise IPDs

$$\Delta\phi_{m,m'}(t, f) = \angle(X_m(t, f) X_{m'}^*(t, f)), \quad 1 \leq m < m' \leq M,$$

yielding $\frac{M(M-1)}{2}$ IPD channels. Concatenation forms $\mathbf{F}_{in} \in \mathbb{R}^{D_{in} \times T \times F}$ with $D_{in} = M + \frac{M(M-1)}{2}$.

Spatial filter generation (SFG)

The Spatial Filter Generation (SFG) block transforms $\mathbf{F}_{\text{in}} \in \mathbb{R}^{D_{\text{in}} \times T \times F}$ into a compact latent representation $\mathbf{F}_{\text{sfg}} \in \mathbb{R}^{H \times T \times F}$. To balance capacity and efficiency, we adopt depthwise-separable convolutions: a depthwise 3×3 kernel per input channel followed by a pointwise 1×1 projection, reducing parameters and multiplications while retaining a local TF receptive field [85].

Specifically, we apply a depthwise 3×3 convolution followed by a pointwise 1×1 projection, and then apply batch normalization and ReLU to the projected features. Denoting the intermediate feature map by $\mathbf{U} \in \mathbb{R}^{H \times T \times F}$, we write

$$\mathbf{U} = \text{ReLU}\left(\text{BN}\left(\text{PWConv}_{1 \times 1}\left(\text{DWConv}_{3 \times 3}(\mathbf{F}_{\text{in}})\right)\right)\right). \quad (112)$$

To emphasize informative channels, we apply a squeeze-and-excitation block [86] to $\mathbf{U}$. Global average pooling over (t, f) yields

$$z_h = \frac{1}{TF} \sum_{t=1}^T \sum_{f=1}^F u_h(t, f), \quad h = 1, \dots, H, \quad (113)$$

forming $\mathbf{z} \in \mathbb{R}^H$. Two small fully connected layers and a sigmoid then produce channel gates $\eta_h \in (0, 1)$, which reweight $\mathbf{U}$:

$$\mathbf{F}_{\text{sfg}} = [f'_h(t, f)]_{h=1}^H, \quad f'_h(t, f) = \eta_h u_h(t, f). \quad (114)$$

$\mathbf{F}_{\text{sfg}}$ serves as input to the learnable beamforming block.

MVDR-inspired convex beamforming

PhysMVNet uses an MVDR-inspired, end-to-end learnable convex beamforming parameterization that directly predicts TF-dependent spatial weights from $\mathbf{F}_{\text{sfg}}$, avoiding explicit covariance estimation and matrix inversion [10]. Although classical MVDR typically employs complex weights, we deliberately adopt real, convex TF weights for gain control and stability, while delegating fine complex correction to the residual refinement stage.

We use a 1×1 convolution with parameters $A \in \mathbb{R}^{H \times M}$ and $\mathbf{b} \in \mathbb{R}^M$ to produce per-channel logits:

$$Z_m(t, f) = \sum_{h=1}^H A_{h,m} F_{\text{sfg},h}(t, f) + b_m, \quad m = 1, \dots, M. \quad (115)$$

A softmax across microphones yields nonnegative, normalized convex weights:

$$a_m(t, f) = \frac{\exp(Z_m(t, f))}{\sum_{m'=1}^M \exp(Z_{m'}(t, f))}, \quad \sum_{m=1}^M a_m(t, f) = 1. \quad (116)$$

The beamformed reference spectrum is then computed by filter-and-sum:

$$Y_{\text{beam}}(t, f) = \sum_{m=1}^M a_m(t, f) X_m(t, f). \quad (117)$$

This is again an explicit convex spatial combiner in the sense of Eq. (22). All operations remain differentiable, allowing the final loss to backpropagate into both the SFG and beamformer layers.

Helmholtz-inspired STFT-domain regularization

We introduce a Helmholtz-inspired regularizer in the STFT domain to discourage irregular local TF fluctuations of $Y_{\text{beam}}(t, f)$. We define

$$\mathcal{L}_{\text{Helm}} = \sum_{t,f} \left| \nabla_{t,f}^2 Y_{\text{beam}}(t, f) + k_f^2 Y_{\text{beam}}(t, f) \right|^2, \quad (118)$$

where $|\cdot|$ denotes complex modulus, $k_f = \frac{2\pi\nu_f}{c}$ is a frequency-dependent scaling factor, ν_f (Hz) is the physical frequency of STFT bin f (e.g., $\nu_f = f F_s / N_{\text{FFT}}$), and c is the speed of sound. Although the true Helmholtz equation involves a spatial Laplacian over (x, y, z) , we do not discretize microphone geometry here. Instead, we approximate a TF Laplacian by a 3×3 finite-difference stencil on the (t, f) grid:

$$\nabla_{t,f}^2 Y_{\text{beam}}(t, f) \approx [L * Y_{\text{beam}}](t, f), \quad (119)$$

with

$$L = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

This regularizer is therefore Helmholtz-inspired rather than physically exact: it acts as a simple second-order TF smoothness prior, while k_f serves as a monotone frequency-dependent weighting factor. Empirically, this prior mitigates abrupt TF artifacts without relying on explicit microphone coordinates, which is desirable under array perturbations.

Three-band residual spectral mapping (RSM)

Although the learnable beamformer suppresses most interference, the full beamformed spectrogram $Y_{\text{beam}} \in \mathbb{C}^{T \times F}$ may still contain residual noise and may attenuate certain speech harmonics. We therefore introduce a lightweight RSM module (Fig. 4.2) that predicts an additive complex correction $\Delta \hat{S}(t, f)$.

We partition the F frequency bins into three contiguous sub-bands (low/mid/high) of sizes F_1, F_2, F_3 with $F_1 + F_2 + F_3 = F$. For each band $b \in \{1, 2, 3\}$, we extract $Y_{\text{beam},b} \in \mathbb{C}^{T \times F_b}$ and form a 2-channel real tensor:

$$\mathbf{U}_b = \begin{bmatrix} \Re Y_{\text{beam},b} \\ \Im Y_{\text{beam},b} \end{bmatrix} \in \mathbb{R}^{2 \times T \times F_b}. \quad (120)$$

A small 2D CNN $\mathcal{F}_{\text{RSM},b}$ (three 3×3 convolutional layers) outputs

$$\mathbf{R}_b = \mathcal{F}_{\text{RSM},b}(\mathbf{U}_b) \in \mathbb{R}^{2 \times T \times F_b}, \quad (121)$$

which is converted to a complex residual elementwise:

$$\Delta \hat{S}_b(t, f) = R_b(1, t, f) + j R_b(2, t, f) \in \mathbb{C}, \quad (122)$$

where the index f is local to band b . Concatenating residuals across bands yields $\Delta \hat{S} \in \mathbb{C}^{T \times F}$, and

we refine by addition:

$$\hat{S}(t, f) = Y_{\text{beam}}(t, f) + \Delta \hat{S}(t, f), \quad \hat{S} \in \mathbb{C}^{T \times F}. \quad (123)$$

By focusing on residual correction rather than full re-estimation, RSM often converges faster and preserves speech structure better [20]. The sub-band design is particularly useful for restoring higher-frequency components that beamforming may over-suppress.

Training objective

All components, including SFG, the learnable beamformer, the Helmholtz-inspired regularizer, and RSM, are trained jointly. We use complex-domain MSE as the primary enhancement loss:

$$\mathcal{L}_{\text{enh}} = \sum_{t,f} |\hat{S}(t, f) - S(t, f)|^2, \quad (124)$$

where $S(t, f)$ again denotes the protocol-defined target spectrum under the evaluation protocol of this chapter. The total loss adds Helmholtz regularization from Eq. (118):

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{enh}} + \lambda_{\text{Helm}} \mathcal{L}_{\text{Helm}}. \quad (125)$$

The final enhanced signal is obtained by inverse STFT of $\hat{S}$.

4.3.2 Experimental Setup

Datasets. We again use the simulated CHiME-3/4 multichannel benchmark [41, 42], because this section studies robustness under controlled perturbations such as extremely low SNR and array mismatch. A controlled simulated benchmark is necessary here: both the low-SNR stress test and the microphone-perturbation analysis require clean references and reproducible geometry, which real recordings do not provide.

Implementation details. Signals are sampled at 16 kHz. STFT uses a 512-sample (32 ms) Hamming window and 256-sample (16 ms) hop, yielding 257 nonnegative-frequency bins. For the 6-channel CHiME-3/4 setting, the input feature dimension is $D_{\text{in}} = M + M(M - 1)/2 = 21$. The

SFG block uses a depthwise 3×3 convolution with dilation 2 and padding 2, followed by a 1×1 projection to $H = 128$ channels and squeeze-and-excitation reweighting. The learnable beamforming block predicts $M = 6$ logits at each TF bin, and each RSM sub-band uses a three-layer 3×3 CNN to predict real and imaginary residuals. Training uses 4 s segments with batch size 2, Adam with initial learning rate 10^{-3} , exponential learning-rate decay, and gradient clipping with maximum norm 5. Unless otherwise stated, $\lambda_{\text{Helm}} = 10^{-3}$.

At $F_s=16$ kHz with $N_{\text{FFT}}=512$, the bin resolution is 31.25 Hz, the nonnegative-frequency bins are indexed as $f = 0, \dots, 256$, and the Nyquist bin is $f = 256$ at 8 kHz. Using zero-based half-open indexing, the three contiguous bands are low (0–2.5 kHz), mid (2.5–5 kHz), and high (5–8 kHz), corresponding to $[0, 80)$, $[80, 160)$, and $[160, 257)$. Thus the upper limit 257 is the exclusive endpoint of the 257-bin nonnegative-frequency axis, not an included bin index.

Evaluation metrics. We report NB-PESQ, WB-PESQ, STOI, SI-SDR, and SDR on the main simulated benchmark because the purpose of this section is to evaluate robustness under controlled enhancement conditions. For the additional low-SNR stress test we focus on NB-PESQ, STOI, and SI-SDR, whereas for the microphone-perturbation experiment we also report the same metrics. These choices reflect the main question of this section, namely whether PhysMVNet improves stability under controlled mismatch while remaining lightweight. FLOPs and parameter count are also reported.

4.3.3 Results and Analysis

Performance comparison

Table 4.3 shows that PhysMVNet achieves the highest NB-PESQ, WB-PESQ, and SI-SDR among the compared methods. In particular, relative to SpatialNet [84], PhysMVNet improves NB-PESQ from 3.49 to 3.52 and SI-SDR from 22.1 dB to 22.8 dB, underscoring its ability to suppress interference while retaining speech quality.

Classical mask-based beamforming pipelines (e.g., NBDF and FT-JNF) follow a two-stage process: a neural network first estimates time–frequency masks, and these masks are then used to

Table 4.3: Comparative results of PhysMVNet on 6-channel simulated CHiME-3/4.

Method	NB-PESQ	WB-PESQ	STOI	SI-SDR (dB)	SDR (dB)
Unprocessed	1.82	1.27	0.87	7.5	7.5
MNMF Beamforming [80]	–	–	0.94	–	16.2
FasNet+TAC [63]	2.53	1.91	0.94	16.0	16.6
NBDF [82]	2.74	2.31	0.95	15.7	16.6
FT-JNF [6]	3.20	2.61	0.97	17.6	18.3
McNet [83]	3.38	2.73	0.98	19.2	19.6
Wang et al. [4]	–	–	0.99	22.0	22.4
Oracle MVDR	2.49	1.94	0.97	17.3	17.7
SpatialNet [84]	3.49	2.88	0.98	22.1	22.3
PhysMVNet (Ours)	3.52	2.93	0.99	22.8	22.9

compute beamforming filters. Because mask estimation and filter design are optimized in a separated (or only loosely coupled) manner, mask errors may propagate to the spatial filters, especially in rapidly varying or nonstationary noise.

In contrast, PhysMVNet keeps an explicit filter-and-sum spatial interface but trains the full pipeline end-to-end. Gradients from both the Helmholtz-inspired regularizer and the residual spectral refinement directly update the spatial-weight predictor and downstream refinement, encouraging more stable TF behavior and improving overall enhancement performance under challenging conditions.

Computational complexity and model size

Table 4.4 compares PhysMVNet with representative baselines in terms of runtime complexity (GFLOPs/s) and parameter count under 8 kHz and 16 kHz sampling. PhysMVNet requires approximately 25 GFLOPs/s at 8 kHz, only slightly above SpatialNet and the most compact STFT-based methods, and about 1.6 M parameters, placing it between lightweight narrow-band networks and larger end-to-end architectures. This trade-off is achieved through efficient convolutional design, differentiable convex beamforming, and lightweight spectral refinement.

Table 4.4: Computational complexity and model size of PhysMVNet.

Method	8 kHz		16 kHz	
	FLOPs (G/s)	#Param (M)	FLOPs (G/s)	#Param (M)
NBDF [82]	19.5	1.2	38.9	1.2
FT-JNF [6]	19.5	1.2	38.9	1.2
McNet [83]	29.7	1.9	59.2	1.9
FasNet+TAC [63]	9.7	2.7	10.1	2.8
SpatialNet [84]	23.1	1.2	46.3	1.6
PhysMVNet (Ours)	25.1	1.6	50.2	1.6

Table 4.5: Ablation study on simulated CHiME-3/4.

Variant	NB-PESQ	WB-PESQ	SI-SDR (dB)
PhysMVNet (Ours)	3.52	2.93	22.8
w/o Helmholtz regularizer	3.48	2.90	22.3
w/o RSM	3.37	2.73	20.9
Traditional MVDR + RSM	3.50	2.91	22.2
Single-band RSM	3.49	2.89	22.4

Ablation studies

We conduct ablation experiments on simulated CHiME-3/4 to evaluate the contribution of each PhysMVNet component. The following variants are considered: removing the Helmholtz-inspired regularizer from the loss; removing RSM and retaining only the learnable beamformer; replacing the learnable convex beamformer with a classical MVDR beamformer followed by the same RSM backend; and replacing the three-band RSM with a single full-band 2D convolutional residual mapper.

From Table 4.5, removing the STFT-domain regularizer reduces SI-SDR by about 0.5 dB, indicating the benefit of discouraging irregular local TF behavior. Omitting RSM entails a loss of nearly 2 dB, showing that spatial filtering alone is insufficient to recover attenuated details, especially at higher frequencies. Replacing the learnable convex beamformer with a traditional MVDR front end yields a drop of about 0.6 dB, confirming that end-to-end optimization improves filter accuracy under the same refinement budget. Lastly, switching to a single-band RSM lowers SI-SDR by about 0.4 dB, implying that sub-band partitioning provides more fine-grained spectral restoration.

Table 4.6: Results under extremely low SNR (-10 dB).

Method	NB-PESQ	STOI	SI-SDR (dB)
Unprocessed	1.12	0.68	5.2
Oracle MVDR	1.95	0.76	12.5
McNet [83]	2.35	0.80	14.1
SpatialNet [84]	2.49	0.82	14.7
PhysMVNet (Ours)	2.57	0.82	15.0

Table 4.7: Results under 2 cm microphone-array perturbation.

Method	NB-PESQ	STOI	SDR (dB)
Unprocessed	1.80	0.86	7.1
Oracle MVDR	2.35	0.95	16.0
McNet [83]	2.85	0.96	18.9
SpatialNet [84]	3.19	0.97	20.8
PhysMVNet (Ours)	3.26	0.98	21.5

Performance under extreme low SNR

Table 4.6 shows results at an extreme SNR of -10 dB. Oracle MVDR struggles (12.5 dB SI-SDR), while PhysMVNet reaches 15.0 dB, about 0.3 dB above the strongest baseline (SpatialNet), along with higher NB-PESQ.

This behavior is consistent with the unified end-to-end design: the Helmholtz-inspired STFT-domain regularizer provides a stabilizing prior that helps prevent spatial-weight drift when the noise field diverges from training conditions.

Robustness to microphone-array perturbation

We further examine the impact of microphone-array perturbations by randomly shifting each microphone’s 3D coordinate by up to ± 2 cm at test time. The resulting phase deviations degrade many methods: as shown in Table 4.7, SpatialNet’s SDR drops to 20.8 dB and McNet’s to 18.9 dB, whereas PhysMVNet still achieves 21.5 dB.

Discussion and transition to objective design (MBRCL). PhysMVNet strengthens the reliability of explicit TF-domain spatial filtering under severe mismatch by combining gain-bounded convex mixing with a stabilizing STFT-domain regularizer and frequency-structured residual restoration. This largely addresses the architecture-level failure mode in which spatial suppression becomes unstable or introduces irregular TF artifacts when spatial evidence is weak.

Nevertheless, robust spatial filtering alone does not fully resolve objective mismatch: once distortion-oriented criteria (e.g., complex MSE or SI-SDR) are already strong, further improvements may correlate weakly with perceptual measures such as PESQ and STOI, which are sensitive to subtle band-wise balance and temporal envelope preservation. The next section therefore shifts focus from model structure to optimization and introduces the multi-band relative contrastive loss (MBRCL), a portable analytic loss module that encourages relative improvement over the noisy observation in perceptually motivated sub-bands without relying on auxiliary learned assessment networks.

4.4 A Non-Learned Multi-Band Relative Contrastive Loss

The preceding two sections focus on how explicit spatial filtering is structured and stabilized for deployment. This section addresses how enhancement models are optimized. In practice, a system may achieve strong SI-SDR or complex MSE while yielding limited improvements in PESQ or STOI, especially once suppression is already strong and the remaining perceptual differences depend on subtle spectral balance and temporal envelope preservation. We therefore propose the multi-band relative contrastive loss (MBRCL), a non-learned analytic objective designed to be portable across architectures and datasets without auxiliary quality-assessment networks or pretrained perceptual encoders.

4.4.1 Proposed Method

Enhancement framework

We consider single-channel speech enhancement in the STFT domain to isolate objective design. In this section, $Y(t, f)$ denotes the single-channel noisy input spectrum; this local notation is distinct

from the spatial-reference symbols used earlier in the chapter. Let s , y , and $\hat{s}$ denote the clean, noisy, and enhanced speech waveforms, respectively, and let $S(t, f)$, $Y(t, f)$, and $\hat{S}(t, f)$ denote their complex STFT spectra. The enhancement network $\mathcal{G}_\theta$ takes the noisy magnitude $|Y|$ as input and predicts a real-valued ratio mask $\widehat{M} \in [0, 1]$:

$$\widehat{M} = \mathcal{G}_\theta(|Y|), \quad \widehat{S} = \widehat{M} \odot Y, \quad (126)$$

where $\odot$ denotes element-wise multiplication. The enhanced waveform $\hat{s}$ is obtained by inverse STFT using the phase of Y . The following subsections focus on the loss used to train $\mathcal{G}_\theta$.

Multi-band partitioning and energy extraction

MBRCL operates on mel-spaced sub-bands rather than individual STFT bins. Let the F nonnegative-frequency bins be partitioned into B non-overlapping contiguous bands following the mel scale [87]. Let $\{k_b\}_{b=0}^B$ denote the band-boundary indices, with $0 = k_0 < k_1 < \dots < k_B = F$. These boundaries are obtained by dividing the mel-frequency range into B equal intervals and mapping the resulting boundaries back to linear frequency-bin indices. The b -th band is

$$\mathcal{F}_b = \{f \mid k_{b-1} \leq f < k_b\}, \quad (127)$$

where the boundary convention is half-open so that each bin belongs to exactly one band. For $X \in \{S, Y, \hat{S}\}$, the band-wise energy at time frame t is

$$E_X(b, t) = \sum_{f \in \mathcal{F}_b} |X(t, f)|^2. \quad (128)$$

This aggregation introduces coarse auditory frequency selectivity and level compression through the log-energy distances below, but it does not attempt to model masking or cross-band interactions.

Multi-band relative contrastive loss

Conventional fidelity objectives such as spectral MSE and SI-SDR encourage accurate reconstruction but treat errors relatively uniformly across the TF plane. MBRCL complements these

criteria with two closed-form terms: a relative log-energy contrastive term using the noisy mixture as a dynamic negative anchor, and a temporal-envelope gradient-consistency term.

1) Relative log-energy contrastive loss. For each band, the enhanced signal should be closer to the clean reference than the noisy input is, measured in a level-compressed log-energy domain. We add $\epsilon = 10^{-8}$ inside all logarithms for numerical stability. The frame-averaged log-energy distance between the enhanced and clean speech is

$$d_{\text{enh}}(b) = \frac{1}{T} \sum_{t=1}^T \left| \log (E_{\hat{S}}(b, t) + \epsilon) - \log (E_S(b, t) + \epsilon) \right|, \quad (129)$$

and the corresponding distance between noisy and clean speech is

$$d_{\text{noisy}}(b) = \frac{1}{T} \sum_{t=1}^T \left| \log (E_Y(b, t) + \epsilon) - \log (E_S(b, t) + \epsilon) \right|. \quad (130)$$

The intended relative-margin condition is

$$d_{\text{noisy}}(b) - d_{\text{enh}}(b) \geq m_b, \quad (131)$$

where $m_b \geq 0$ is the prescribed improvement margin for band b . The smooth hinge form used for optimization is

$$\mathcal{L}_{\text{band}} = \sum_{b=1}^B w_b \text{softplus}(d_{\text{enh}}(b) - d_{\text{noisy}}(b) + m_b), \quad (132)$$

where $\text{softplus}(z) = \log(1 + e^z)$ and $w_b \geq 0$ is a band weight. The loss is small when $d_{\text{enh}}(b) \leq d_{\text{noisy}}(b) - m_b$ and increases with margin violation. In the experiments, all bands share the same margin $m_b \equiv m$.

2) Temporal envelope gradient consistency loss. The relative log-energy term improves band-level spectral balance, but it can still allow over-smoothed temporal envelopes. To preserve temporal dynamics, we convert band energies to amplitudes,

$$e_X(b, t) = \sqrt{E_X(b, t) + \epsilon_e}, \quad X \in \{S, \hat{S}\}, \quad (133)$$

with $\epsilon_e = 10^{-8}$, and compute first-order temporal differences for $t = 2, \dots, T$:

$$\begin{aligned}\Delta e_S(b, t) &= e_S(b, t) - e_S(b, t-1), \\ \Delta e_{\hat{S}}(b, t) &= e_{\hat{S}}(b, t) - e_{\hat{S}}(b, t-1).\end{aligned}\tag{134}$$

The temporal gradient consistency loss is

$$\mathcal{L}_{\text{env}} = \sum_{b=1}^B \frac{1}{T-1} \sum_{t=2}^T |\Delta e_{\hat{S}}(b, t) - \Delta e_S(b, t)|.\tag{135}$$

This term penalizes both over-smoothed envelope trajectories and spurious temporal fluctuations in sub-band energy.

3) Overall training objective. MBRCL is used as an auxiliary perceptual-structure objective together with standard fidelity losses. We adopt negative SI-SDR [52] as the waveform-level criterion $\mathcal{L}_{\text{SI}}$ and multi-resolution STFT loss [88] as the spectral criterion $\mathcal{L}_{\text{spec}}$. The total objective is

$$\mathcal{L}_{\text{total}} = \lambda_{\text{SI}} \mathcal{L}_{\text{SI}} + \lambda_{\text{spec}} \mathcal{L}_{\text{spec}} + \lambda_{\text{band}} \mathcal{L}_{\text{band}} + \lambda_{\text{env}} \mathcal{L}_{\text{env}},\tag{136}$$

where λ_{SI} , λ_{spec} , λ_{band} , and λ_{env} are nonnegative scalar weights. Because $\mathcal{L}_{\text{band}}$ and $\mathcal{L}_{\text{env}}$ include absolute values, the objective is differentiable almost everywhere; standard automatic-differentiation packages use subgradients at the measure-zero kink points.

4.4.2 Experimental Setup

Datasets. We evaluate on a simulated TIMIT-based corpus and VoiceBank-DEMAND. Following [77], the TIMIT simulation mixes 300 training utterances with 10 non-speech noises at SNRs from -8 to 8 dB in 4 dB steps, uses 100 held-out validation utterances with different noises and SNRs, and tests on 100 core-test utterances mixed with four unseen noises at $-6, 0, 6, 12$, and 18 dB. VoiceBank-DEMAND follows [44] with 11,572 training utterances from 28 speakers and 824 test utterances from 2 unseen speakers and 5 unseen noises.

Model architectures. Unless otherwise stated, the BLSTM and CNN-U-Net models use 257-bin log-magnitude STFTs computed with a 512-point FFT, 32 ms Hamming window, and 16 ms hop.

The BLSTM mask estimator follows [77]: two BLSTM layers with 200 hidden units per direction, a 300-unit fully connected layer with LeakyReLU, and a 257-unit sigmoid mask output. The CNN-U-Net is a compact four-level TF U-Net with encoder widths $16 \rightarrow 32 \rightarrow 64 \rightarrow 128$, mirrored decoder blocks, skip connections, and a final 1×1 sigmoid layer. We also evaluate the recent low-complexity GTCRN backbone [89]; in that comparison, only the training loss is changed.

Baselines and training details. We compare MBRCL with the matched fidelity baseline SI-SDR+MRSTFT (Base), PMSQE fine-tuning [78], Quality-Net (Q-Net) fine-tuning [77], and a frozen WavLM feature-space loss (SSL-Feat). For BLSTM and CNN-U-Net, models are first pretrained with $\mathcal{L}_{\text{base}} = \lambda_{\text{SI}}\mathcal{L}_{\text{SI}} + \lambda_{\text{spec}}\mathcal{L}_{\text{spec}}$ for 60 epochs and then fine-tuned for 20 epochs from the same checkpoint. Base simply continues with $\mathcal{L}_{\text{base}}$, whereas PMSQE, Q-Net, SSL-Feat, and MBRCL use the same initialization and fine-tuning budget and differ only in the added auxiliary term. For GTCRN, we keep the official training recipe and report only a matched Base-versus-MBRCL comparison. We use Adam with initial learning rate 10^{-3} for BLSTM and CNN-U-Net; batch size is 8 on TIMIT and 16 on VoiceBank-DEMAND. For $\mathcal{L}_{\text{spec}}$, the MRSTFT resolutions are (512, 320, 160), (1024, 640, 320), and (2048, 1280, 640). Unless otherwise noted, $\lambda_{\text{SI}} = 0.5$, $\lambda_{\text{spec}} = 1.0$, $\lambda_{\text{band}} = 0.5$, $\lambda_{\text{env}} = 0.2$, $B = 16$, $w_b = 1$, and $m = 0.06$; all hyperparameters are selected on validation data. The margin $m = 0.06$ is in the natural-log energy domain and corresponds to approximately $10 \log_{10}(e^{0.06}) \approx 0.26$ dB in a band-energy ratio.

4.4.3 Results and Analysis

Performance on VoiceBank-DEMAND

Table 4.8 reports results on VoiceBank-DEMAND. For BLSTM, PMSQE mainly improves PESQ and DNSMOS-OVRL but slightly reduces STOI with little change in SI-SDRi, while Q-Net further improves PESQ and SI-SDRi and restores STOI close to the baseline level. The SSL-Feat baseline yields additional gains, but MBRCL achieves the best scores on all four metrics. A similar trend is observed for CNN-U-Net. On GTCRN, MBRCL is compared only with the matched Base objective to verify cross-backbone compatibility, and it again improves all metrics. These results indicate that the proposed loss is compatible with the tested backbones and remains competitive

Table 4.8: Results on VoiceBank-DEMAND.

Architecture	Method	PESQ	STOI	SI-SDRi (dB)	DNSMOS
–	Noisy input	1.97	0.920	0.00	2.45
BLSTM	Base	2.53	0.935	7.02	2.75
	PMSQE	2.61	0.932	6.98	2.81
	Q-Net	2.67	0.934	7.37	2.86
	SSL-Feat	2.73	0.936	7.46	2.92
	MBRCL	2.78	0.938	7.58	2.98
CNN-U-Net	Base	2.46	0.931	6.73	2.71
	PMSQE	2.55	0.929	6.81	2.78
	MBRCL	2.69	0.936	7.27	2.91
GTCRN	Base	2.87	0.945	8.68	3.08
	MBRCL	2.94	0.948	8.87	3.15

Table 4.9: PESQ / STOI scores on the TIMIT test set (BLSTM backbone).

SNR (dB)	Noisy input	BLSTM Variants			
		Base	Q-Net	SSL-Feat	MBRCL
18	2.81/0.971	3.28/0.971	3.37/0.970	3.39/0.972	3.42/0.973
12	2.07/0.924	2.96/0.958	3.03/0.955	3.05/0.957	3.09/0.959
6	1.96/0.832	2.49/0.892	2.58/0.884	2.63/0.887	2.67/0.893
0	1.81/0.716	2.11/0.813	2.48/0.829	2.52/0.834	2.56/0.841
−6	1.24/0.583	1.69/0.724	1.82/0.758	1.86/0.763	1.89/0.772

with learned-loss and frozen-encoder perceptual objectives in the evaluated settings.

Performance on TIMIT

Table 4.9 summarizes PESQ and STOI on the TIMIT test set at different SNRs for the BLSTM model. The column labeled “Noisy input” is the unprocessed noisy mixture before enhancement. MBRCL improves PESQ and matches or slightly improves STOI relative to Q-Net and SSL-Feat, with the largest gains at low SNR where the noisy input is farthest from the clean reference. At high SNR, where the noisy input is already close to clean speech, the room for additional improvement under the relative term is limited, and MBRCL does not degrade intelligibility.

Ablation study

Table 4.10 presents an ablation study on VoiceBank-DEMAND with the BLSTM architecture. Besides PESQ, STOI, and SI-SDRi, it reports two diagnostics computed on the test set under a

Table 4.10: Ablation on MBRCL components (VoiceBank-DEMAND, BLSTM).

No.	Configuration			Metrics				
	B	m	$\mathcal{L}_{\text{env}}$	PESQ	STOI	SI-SDRi	MSR $\uparrow$	nGED $\downarrow$
1	<i>Base</i>			2.53	0.935	7.02	14.8	0.421
2	16	–	$\times$	2.60	0.934	7.14	43.7	0.408
3	8	0.03	$\times$	2.66	0.936	7.23	61.4	0.392
4	16	0.03	$\times$	2.72	0.935	7.38	67.9	0.377
5	16	0.06	$\times$	2.75	0.937	7.49	81.6	0.371
6	16	0.06	$\checkmark$	2.78	0.938	7.58	80.9	0.286
7	24	0.06	$\checkmark$	2.77	0.937	7.51	77.3	0.298

fixed evaluation setting with $B = 16$ and $m = 0.06$. The margin satisfaction rate (MSR) is the percentage of utterance-band instances satisfying $d_{\text{noisy}}(b) - d_{\text{enh}}(b) \geq 0.06$. The normalized gradient-envelope deviation (nGED) is the normalized test-set average of the temporal-envelope gradient mismatch in Eq. (135). Starting from the Base system (No. 1), configuration No. 2 adds a simple multi-band log-energy loss that aligns the enhanced signal with the clean reference while removing the noisy-anchor comparison:

$$\mathcal{L}_{\text{band-clean}} = \sum_{b=1}^B w_b d_{\text{enh}}(b). \quad (137)$$

This yields moderate gains in PESQ and SI-SDRi, suggesting that mel-band aggregation is beneficial even without the relative constraint.

Configurations No. 3–No. 5 progressively introduce the full relative-contrastive mechanism. Compared with the non-relative variant (No. 2), using the noisy input as a per-band negative anchor and enforcing positive margins of $m = 0.03$ or $m = 0.06$ leads to consistent gains in PESQ and SI-SDRi, while the increase in MSR shows that the intended relative ordering is satisfied more frequently. Increasing the number of bands from $B = 8$ to $B = 16$ (No. 3 vs. No. 4) further improves performance, and increasing the margin from $m = 0.03$ to $m = 0.06$ (No. 4 vs. No. 5) brings an additional PESQ gain. Comparing No. 5 and No. 6 shows that enabling $\mathcal{L}_{\text{env}}$ yields the lowest nGED together with the highest STOI and SI-SDRi, supporting its role in temporal-envelope

tracking. Increasing the number of bands to $B = 24$ (No. 7) slightly degrades performance, likely because very narrow bands increase the variance of energy estimates.

Although MBRCL is demonstrated here in a single-channel masking setting, it is architecture-agnostic: it only requires $(\hat{S}, S, Y)$. Therefore, it can also be attached to multichannel systems, such as DSF-RSM or PhysMVNet, once an enhanced output is produced. In the thesis context, this decouples model design, namely explicit deployable spatial filtering, from objective design, namely perceptual-band aligned optimization, enabling systematic reuse across chapters and architectures.

4.5 Conclusion

This chapter focused on a deployment-oriented gap that becomes critical once topology robustness is improved: spatial processing should not only be accurate, but also explicit, stable under mismatch, and efficient, and the training objective should better align with perceptual and intelligibility-oriented evaluation. We therefore centered the design around a modular blueprint: explicit TF-domain spatial filtering in a filter-and-sum form, followed by lightweight spectral restoration, complemented by objective alignment.

Following this blueprint, we established DSF-RSM, demonstrating that a compact learned filter generator can replace heavy covariance-driven operations while retaining an inspectable filtering interface and enabling complex-domain residual correction. We then improved robustness under severe mismatch with PhysMVNet, which preserves the same deployable structure but introduces stabilization through a physics-inspired TF regularizer and a frequency-structured residual restoration strategy to mitigate over-suppression and irregular TF artifacts. Finally, we addressed objective mismatch by proposing MBRCL, a portable analytic multi-band relative contrastive loss that enforces relative improvement over the noisy observation in perceptually motivated bands, without introducing additional learned assessment networks.

Overall, the chapter clarifies what turns “spatial learning” into a deployable multichannel front end: keep the spatial operator explicit and bounded, inject stability-oriented inductive bias for low-SNR and mismatch regimes, and adopt objective modules that better reflect perceptual or utility-driven evaluation. With these components established, the remaining limitation increasingly lies in

long-range temporal dependency modeling and coordinated magnitude–phase refinement in difficult acoustic scenes. Chapter 5 addresses this next step via structured long-context sequence modeling and cross-domain coordination.

Chapter 5

Dual-Path State-Space Modeling with Cross-Domain Interaction

5.1 Introduction

Chapters 3–4 focused on deployment-oriented spatial front ends for multichannel speech enhancement (MCSE): topology-robust inter-channel modeling, explicit and gain-bounded spatial interfaces, and stability- and objective-oriented refinements under mismatch. After the spatial filtering stage has been made more explicit and robust, difficult acoustic scenes expose another bottleneck. Residual interference and reverberation tails often span long time ranges, and magnitude-only enhancement can leave phase-related artifacts that limit perceptual quality and downstream robustness.

The remaining gap is therefore twofold: (i) we need long-context temporal modeling that remains efficient at sequence scale, and (ii) we need coordinated enhancement of magnitude and complex-spectrum structure rather than treating them as loosely coupled post-processing steps. State-space sequence models offer an attractive efficiency profile for long sequences [29], but an effective MCSE design still requires a structured interface that aligns amplitude restoration with phase-consistent spectral refinement.

This chapter addresses these requirements with an end-to-end MCSE system based on dual-path state-space modeling with cross-domain interaction. An S3Conv-based encoder extracts spatial–spectral representations from multichannel complex STFT inputs [90]. Two parallel S5 backbones

then specialize in (i) coarse magnitude estimation (CME) for robust suppression and amplitude restoration and (ii) complex spectral refinement (CSR) for fine spectral detail and phase-related cues [29]. Crucially, an interaction module exchanges intermediate representations between CME and CSR so that amplitude cues can guide complex refinement and phase-consistent structure can regularize magnitude restoration before reconstruction.

The remainder of this chapter is organized as follows. Section 5.2 presents the proposed architecture and interaction mechanism, Section 5.3 summarizes the experimental protocol, and Section 5.4 reports results and analysis. Finally, Section 5.5 concludes this chapter and motivates Chapter 6, which focuses on recognition-centric front-end design under frozen ASR back ends.

5.2 Proposed Method

5.2.1 Overall Architecture

Figure 5.1 illustrates the proposed architecture, which comprises an encoder built on S3Conv layers [90] for initial feature extraction, dual-path S5 modules [29] for long-context sequence modeling, a cross-domain interaction module for structured feature exchange, and a decoder that reconstructs the enhanced complex spectrum.

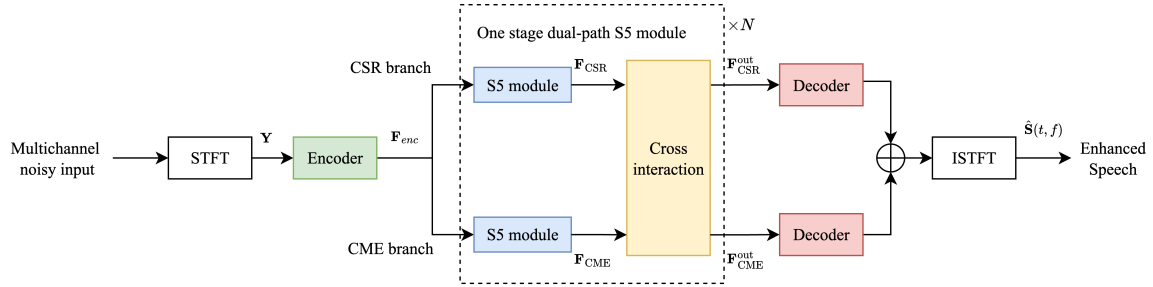


Figure 5.1: Overall architecture of the proposed dual-path model for multichannel speech enhancement. To preserve the original diagram, the schematic input label Y corresponds to the real-valued multichannel input representation $\mathbf{X}_{\text{RI}}$ in the text. The internal branch labels $\mathbf{F}_{\text{CME}}$ and $\mathbf{F}_{\text{CSR}}$ denote the CME and CSR branch features before cross-domain interaction, and $\mathbf{F}_{\text{CME}}^{\text{out}}$ and $\mathbf{F}_{\text{CSR}}^{\text{out}}$ denote the post-interaction features passed to the decoders.

Let $X_m(t, f) \in \mathbb{C}$ denote the noisy mixture STFT coefficient of microphone $m \in \{1, \dots, M\}$ at time frame t and frequency bin f . For network processing, we represent the multichannel complex

STFT in real-valued form by a tensor

$$\mathbf{X}_{\text{RI}} \in \mathbb{R}^{M \times T \times F \times C_{\text{in}}}, \quad C_{\text{in}} = 2,$$

where the last axis corresponds to the real and imaginary components. As in Chapter 2, F denotes the number of nonnegative-frequency bins retained for STFT-domain processing.

The encoder produces a multichannel feature map

$$\mathbf{F}_{\text{enc}} \in \mathbb{R}^{M \times T \times F \times C_{\text{enc}}},$$

where C_{enc} is the number of output channels per microphone. These features are forwarded to the dual-path S5 backbone.

The dual-path S5 backbone is central to our framework: it models long-range temporal dependencies and structured spectral dynamics in challenging acoustic scenes. Rather than fusing only final outputs, we exchange intermediate feature representations between the CME and CSR branches through a cross-domain interaction module, allowing complementary magnitude and complex-spectrum cues to be aligned before subsequent refinement.

Figure 5.1 illustrates a general multi-stage S5 design. As discussed in Section 5.4, ablation studies show that two S5 stages provide an effective trade-off between performance and implementation complexity. Accordingly, we use two S5 stages throughout this chapter.

Finally, the decoder processes the refined branch features and reconstructs the enhanced complex spectrum $\hat{S}(t, f)$, which is converted back to the waveform by iSTFT. Unless otherwise stated, $\hat{S}(t, f)$ is trained against the protocol-defined target spectrum $S(t, f)$ introduced in Chapter 2.

5.2.2 Encoder

The encoder transforms the raw multichannel input into a feature representation suitable for long-context modeling. As depicted in Figure 5.2, it is built around the S3Conv module [90], which captures spatial and spectral structure via parallel convolutional operations.

The input $\mathbf{X}_{\text{RI}}$ is processed by two parallel pathways. The spatial pathway applies convolution

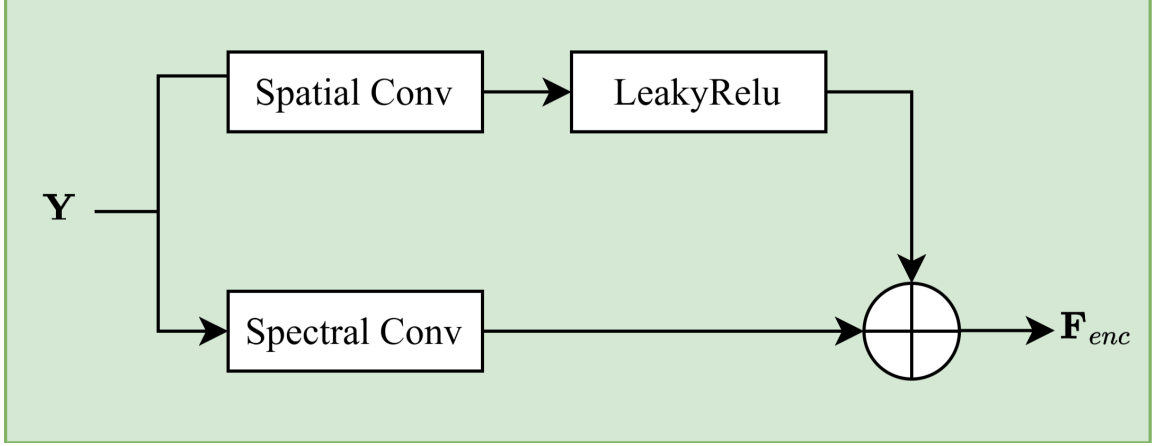


Figure 5.2: Structure of the S3Conv-based encoder. The encoder input corresponds to $\mathbf{X}_{\text{RI}}$, and the encoder output corresponds to $\mathbf{F}_{\text{enc}}$ obtained after the spatial/TF fusion and instance-normalization steps in Eqs. (138)–(143).

along the microphone axis, shared across time–frequency locations, followed by a LeakyReLU nonlinearity:

$$\mathbf{Y}_{\text{spatial}} = \text{LeakyReLU}(\mathbf{W}_{\text{spatial}} * \mathbf{X}_{\text{RI}} + \mathbf{b}_{\text{spatial}}), \quad (138)$$

where $\mathbf{W}_{\text{spatial}} \in \mathbb{R}^{K_s \times C_{\text{in}} \times C_{\text{spatial}}}$ is a spatial kernel of size K_s (we use $K_s=3$), C_{spatial} is the number of output channels, and $*$ denotes convolution.

In parallel, a TF convolution extracts local patterns on the time–frequency grid:

$$\mathbf{Y}_{\text{spectral}} = \mathbf{W}_{\text{spectral}} * \mathbf{X}_{\text{RI}} + \mathbf{b}_{\text{spectral}}, \quad (139)$$

where $\mathbf{W}_{\text{spectral}} \in \mathbb{R}^{K_t \times K_f \times C_{\text{in}} \times C_{\text{spectral}}}$ with kernel sizes $(K_t, K_f) = (5, 1)$ in our implementation, and C_{spectral} is the number of output channels.

For element-wise fusion, we set $C_{\text{spatial}} = C_{\text{spectral}} = C_{\text{enc}}$ so that both pathways output $\mathbb{R}^{M \times T \times F \times C_{\text{enc}}}$ tensors. The outputs of these two paths are fused through element-wise addition:

$$\mathbf{Y}_{\text{combined}} = \mathbf{Y}_{\text{spatial}} + \mathbf{Y}_{\text{spectral}}. \quad (140)$$

To stabilize training, the fused map $\mathbf{Y}_{\text{combined}} \in \mathbb{R}^{M \times T \times F \times C_{\text{enc}}}$ is instance-normalized per

microphone and per feature channel. For each microphone $m \in \{1, \dots, M\}$ and feature channel $c \in \{1, \dots, C_{\text{enc}}\}$, we compute statistics over (t, f) :

$$\mu_{m,c} = \frac{1}{TF} \sum_{t=1}^T \sum_{f=1}^F \mathbf{Y}_{\text{combined}}(m, t, f, c), \quad (141)$$

$$\sigma_{m,c}^2 = \frac{1}{TF} \sum_{t=1}^T \sum_{f=1}^F (\mathbf{Y}_{\text{combined}}(m, t, f, c) - \mu_{m,c})^2. \quad (142)$$

The normalized representation is then

$$\tilde{\mathbf{Y}}(m, t, f, c) = \gamma_c \frac{\mathbf{Y}_{\text{combined}}(m, t, f, c) - \mu_{m,c}}{\sqrt{\sigma_{m,c}^2 + \varepsilon}} + \beta_c, \quad (143)$$

with $\varepsilon = 10^{-5}$. The learnable scale γ_c and bias β_c restore the capacity to adapt mean and variance after normalization. We denote the encoder output as

$$\mathbf{F}_{\text{enc}} = \tilde{\mathbf{Y}} \in \mathbb{R}^{M \times T \times F \times C_{\text{enc}}},$$

and feed it to both the CME and CSR branches.

To reduce dependence on microphone ordering, during training we randomly permute the microphone axis of $\mathbf{X}_{\text{RI}}$ as a data augmentation. Since the model outputs a single enhanced stream and the loss is computed against the same single-channel target, this permutation does not require any special handling in the loss computation. In contrast to the topology-oriented models in Chapter 3, this chapter does not claim strict permutation equivariance; rather, the goal is robustness under a fixed- M deployment setting.

5.2.3 Dual-Path S5 Model

Figure 5.3 provides a structural overview of the S5 module, which implements a simplified state-space layer for efficient sequence modeling [28, 29]. In our setting, S5 is used to capture both short-term and long-term temporal structure while maintaining tractable computation at long sequence lengths.

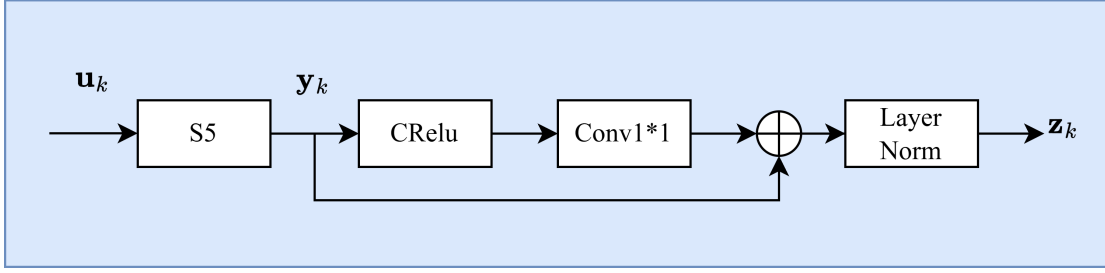


Figure 5.3: Overview of the S5 module, which captures both short-term and long-term dependencies. The labels $\mathbf{u}_k$, $\mathbf{y}_k$, and $\mathbf{z}_k$ denote the flattened per-frame input vector, the S5 output before the lightweight refinement path, and the post-refinement block output, respectively, consistent with Eqs. (148)–(149). The schematic label CRelu corresponds to the generic pointwise nonlinearity $\sigma(\cdot)$ used in the text.

We use four internal dimensions: d_u (input-vector length), d_v (state size), d_y (output-vector length), and d_h (hidden projection size used in the lightweight per-block channel-mixing projections). At each STFT frame t , the encoder provides $\mathbf{F}_{\text{enc}}(:, t, :, :) \in \mathbb{R}^{M \times F \times C_{\text{enc}}}$. We flatten this tensor into a vector $\mathbf{u}_k \in \mathbb{R}^{d_u}$ by stacking the microphone axis (M), frequency bins (F), and feature channels (C_{enc}), with $k = t - 1$ and

$$d_u = M F C_{\text{enc}}.$$

The resulting sequence $\{\mathbf{u}_k\}_{k=0}^{T-1}$ is fed into an S5 block. We use $d_y = d_u$ so that each output $\mathbf{y}_k \in \mathbb{R}^{d_y}$ can be reshaped back to $\mathbb{R}^{M \times F \times C_{\text{enc}}}$ and aligned with the original (m, f) grid per frame.

The current implementation assumes a fixed microphone count within a given training/evaluation configuration: at inference, M matches the value used during training. In our experiments, we use $M=5$ for the simulated set, $M=6$ for CHiME-4, and $M=8$ for ConferencingSpeech 2021. This chapter focuses on long-context modeling and cross-domain magnitude–complex coordination under a fixed array configuration; extending the same design to variable- M inference is left as future work.

For exposition, we start from a standard continuous-time state-space representation:

$$\frac{d\mathbf{v}(\xi)}{d\xi} = \mathbf{A}\mathbf{v}(\xi) + \mathbf{B}\mathbf{u}(\xi), \quad (144)$$

$$\mathbf{y}(\xi) = \mathbf{C}\mathbf{v}(\xi) + \mathbf{D}\mathbf{u}(\xi), \quad (145)$$

where ξ denotes continuous time, $\mathbf{u}(\xi) \in \mathbb{R}^{d_u}$ is the input, $\mathbf{v}(\xi) \in \mathbb{R}^{d_v}$ is the latent state, and $\mathbf{y}(\xi) \in \mathbb{R}^{d_y}$ is the output. For practical implementation, the system is discretized using the bilinear (Tustin) transformation [29]:

$$\mathbf{v}_{k+1} = \mathbf{A}_d \mathbf{v}_k + \mathbf{B}_d \mathbf{u}_k, \quad (146)$$

$$\mathbf{y}_k = \mathbf{C}_d \mathbf{v}_k + \mathbf{D}_d \mathbf{u}_k, \quad (147)$$

where k is the discrete frame index ($0 \leq k < T$).

In our dual-path framework, the extracted TF features are processed by two parallel branches (CME and CSR). Each branch employs a two-stage S5 design: the first S5 block emphasizes coarse temporal structure, and its intermediate representation participates in cross-domain interaction; the second S5 block then performs further refinement conditioned on the fused features.

Following the S5 parameterization used in our implementation, we write the discrete-time update with an explicit feed-through term:

$$\begin{aligned} \mathbf{v}_{k+1} &= \mathbf{A}_{S5} \mathbf{v}_k + \mathbf{B}_{S5} \mathbf{u}_k, \\ \mathbf{y}_k &= \mathbf{C}_{S5} \mathbf{v}_k + \mathbf{D}_{S5} \mathbf{u}_k, \end{aligned} \quad (148)$$

where $\mathbf{v}_k \in \mathbb{R}^{d_v}$ encodes temporal dynamics and $\mathbf{u}_k \in \mathbb{R}^{d_u}$ denotes the stacked multi-microphone TF features. All S5 parameters $\mathbf{A}_{S5}, \mathbf{B}_{S5}, \mathbf{C}_{S5}, \mathbf{D}_{S5}$ are learned during training and fixed during inference. The matrices in (148) denote the effective maps of the implemented S5 layer; the large input and output transformations are realized with the structured MIMO S5 parameterization and lightweight projection layers rather than as unrestricted dense matrices of size $d_u \times d_v$ or $d_y \times d_v$.

The feed-through matrix $\mathbf{D}_{S5}$ enables direct input-to-output interaction, helping retain fine-grained spectral information while modeling long-range temporal structure. Adaptive modulation and residual connections are employed within each S5 block to stabilize training and improve robustness. After each S5 block, we apply a lightweight channel-mixing refinement step. Let $\mathcal{R}(\cdot)$ reshape $\mathbf{y}_k$ to its $M \times F \times C_{\text{enc}}$ form and let $\mathcal{R}^{-1}(\cdot)$ invert that reshape. The refined representation is

$$\mathbf{z}_k = \text{LayerNorm}\left(\mathbf{y}_k + \mathcal{R}^{-1}(\text{Conv}_{1 \times 1}(\sigma(\mathcal{R}(\mathbf{y}_k))))\right), \quad (149)$$

where $\sigma(\cdot)$ is a pointwise nonlinearity and $\text{Conv}_{1 \times 1}$ denotes channel mixing on the reshaped tensor. The resulting vectors are reshaped back to $M \times F \times C_{\text{enc}}$ per frame and forwarded either to (i) the interaction module (after stage 1) or (ii) the decoder (after stage 2), depending on the branch and stage.

5.2.4 Cross-Domain Interaction Module

The cross-domain interaction module enables structured feature exchange between CME and CSR. After the first-stage S5 block in each branch, intermediate feature maps are fed into this module, which fuses complementary magnitude and complex-spectrum cues. The overall structure is illustrated in Figure 5.4.

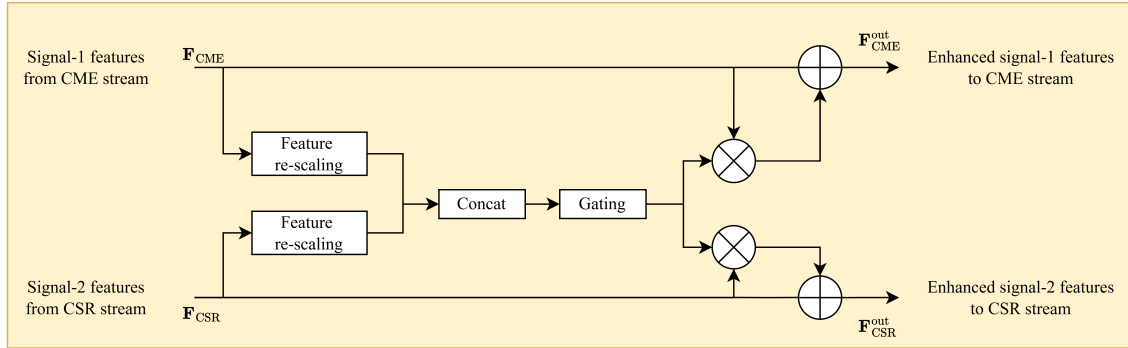


Figure 5.4: Diagram of the cross-domain interaction module. The two inputs correspond to the first-stage CME and CSR branch features. In the text, the projected single-stream interaction features are denoted by $\mathbf{F}_{\text{CME}}$ and $\mathbf{F}_{\text{CSR}}$ in Eq. (150), and the post-interaction outputs are denoted by $\mathbf{F}_{\text{CME}}^{\text{out}}$ and $\mathbf{F}_{\text{CSR}}^{\text{out}}$ in Eq. (154).

Let the first-stage outputs (after reshaping) be

$$\mathbf{O}_{\text{CME}} \in \mathbb{R}^{M \times T \times F \times d_{\text{CME}}}, \quad \mathbf{O}_{\text{CSR}} \in \mathbb{R}^{M \times T \times F \times d_{\text{CSR}}}.$$

To form a compact interaction interface, we first compute a learned microphone projection for each branch:

$$\begin{aligned} \mathbf{F}_{\text{CME}}(t, f, c) &= \sum_{m=1}^M \omega_{m,c}^{\text{CME}} \mathbf{O}_{\text{CME}}(m, t, f, c), \\ \mathbf{F}_{\text{CSR}}(t, f, c) &= \sum_{m=1}^M \omega_{m,c}^{\text{CSR}} \mathbf{O}_{\text{CSR}}(m, t, f, c), \end{aligned} \tag{150}$$

where $\omega_{m,c}^{\text{CME}}$ and $\omega_{m,c}^{\text{CSR}}$ are trainable microphone-combination coefficients. These coefficients define a feature projection for the interaction module and are not intended to be interpreted as beam-forming weights. This yields single-stream feature maps $\mathbf{F}_{\text{CME}} \in \mathbb{R}^{T \times F \times d_{\text{CME}}}$ and $\mathbf{F}_{\text{CSR}} \in \mathbb{R}^{T \times F \times d_{\text{CSR}}}$.

Each map is then channel-wise re-scaled by a learned vector (broadcast over (t, f)):

$$\begin{aligned} \mathbf{F}'_{\text{CME}}(t, f, c) &= g_c^{\text{CME}} \mathbf{F}_{\text{CME}}(t, f, c), \\ \mathbf{F}'_{\text{CSR}}(t, f, c) &= g_c^{\text{CSR}} \mathbf{F}_{\text{CSR}}(t, f, c), \end{aligned} \tag{151}$$

where $\mathbf{g}^{\text{CME}} \in \mathbb{R}^{d_{\text{CME}}}$ and $\mathbf{g}^{\text{CSR}} \in \mathbb{R}^{d_{\text{CSR}}}$.

Next, the re-scaled features are concatenated along the channel axis:

$$\mathbf{F}_{\text{fusion}} = \text{Concat}(\mathbf{F}'_{\text{CME}}, \mathbf{F}'_{\text{CSR}}) \in \mathbb{R}^{T \times F \times (d_{\text{CME}} + d_{\text{CSR}})}. \tag{152}$$

A gating mechanism dynamically selects informative components:

$$\mathbf{F}_{\text{gated}} = \text{Sigmoid}(\mathbf{F}_{\text{fusion}}) \odot \mathbf{F}_{\text{fusion}}, \tag{153}$$

where $\odot$ denotes the Hadamard product. We split $\mathbf{F}_{\text{gated}} = [\mathbf{F}_{\text{gated}}^{\text{CME}}; \mathbf{F}_{\text{gated}}^{\text{CSR}}]$ with $\mathbf{F}_{\text{gated}}^{\text{CME}} \in \mathbb{R}^{T \times F \times d_{\text{CME}}}$ and $\mathbf{F}_{\text{gated}}^{\text{CSR}} \in \mathbb{R}^{T \times F \times d_{\text{CSR}}}$.

Finally, residual updates preserve original information and facilitate gradient flow:

$$\begin{aligned}\mathbf{F}_{\text{CME}}^{\text{out}} &= \mathbf{F}_{\text{gated}}^{\text{CME}} + \mathbf{F}'_{\text{CME}}, \\ \mathbf{F}_{\text{CSR}}^{\text{out}} &= \mathbf{F}_{\text{gated}}^{\text{CSR}} + \mathbf{F}'_{\text{CSR}}.\end{aligned}\tag{154}$$

Broadcast injection for stage-2 consistency. Since the subsequent stage-2 S5 blocks operate on mic-resolved tensors, we inject the interaction outputs back into the mic-resolved representations by broadcasting along the microphone axis. The corresponding branch-specific flattened input sizes are MFd_{CME} and MFd_{CSR} , which reduce to MFC_{enc} in our default configuration where $d_{\text{CME}} = d_{\text{CSR}} = C_{\text{enc}}$:

$$\tilde{\mathbf{O}}_{\text{CME}}(m, t, f, c) = \mathbf{O}_{\text{CME}}(m, t, f, c) + \mathbf{F}_{\text{CME}}^{\text{out}}(t, f, c),\tag{155}$$

$$\tilde{\mathbf{O}}_{\text{CSR}}(m, t, f, c) = \mathbf{O}_{\text{CSR}}(m, t, f, c) + \mathbf{F}_{\text{CSR}}^{\text{out}}(t, f, c).\tag{156}$$

These conditioned tensors satisfy $\tilde{\mathbf{O}}_{\text{CME}} \in \mathbb{R}^{M \times T \times F \times d_{\text{CME}}}$ and $\tilde{\mathbf{O}}_{\text{CSR}} \in \mathbb{R}^{M \times T \times F \times d_{\text{CSR}}}$, and are then forwarded to the second-stage S5 blocks in the corresponding branches.

5.2.5 Decoder

The decoder reconstructs the enhanced complex spectrum from the refined features produced by the CME and CSR streams. To exploit their complementary roles, we use two parallel decoder branches: a CME decoder that emphasizes amplitude restoration and a CSR decoder that refines fine-grained spectral details (including phase-related structure).

Each branch consists of several transposed-convolution layers. For every layer we set stride to 1 and choose padding so that temporal and spectral resolutions are preserved: if the input has T_{in} frames and F_{in} frequency bins, the output satisfies $T_{\text{out}} = T_{\text{in}}$ and $F_{\text{out}} = F_{\text{in}}$. Thus, the transposed kernels act as learnable synthesis filters rather than geometric up-samplers. Kernel shapes such as 5×1 , 1×5 , and 3×3 refine local TF patterns while maintaining full resolution.

Let the decoder outputs of the two branches be $\hat{S}_{\text{CME}}(t, f) \in \mathbb{C}$ and $\hat{S}_{\text{CSR}}(t, f) \in \mathbb{C}$. The final enhanced spectrum is formed by element-wise addition,

$$\hat{S}(t, f) = \hat{S}_{\text{CME}}(t, f) + \hat{S}_{\text{CSR}}(t, f),\tag{157}$$

and iSTFT converts $\hat{S}(t, f)$ back to the time-domain waveform.

5.2.6 Loss Function

To train the dual-path S5 model with cross-domain interaction, we use a weighted sum of three losses: an SI-SDR loss, a magnitude loss, and a complex spectral loss.

Let $\mathbf{s} \in \mathbb{R}^{N_{\text{samp}}}$ denote the clean speech waveform and $\hat{\mathbf{s}} \in \mathbb{R}^{N_{\text{samp}}}$ its enhanced estimate. The scale-invariant signal-to-distortion ratio (SI-SDR) [52] is optimized by minimizing its negative:

$$\mathcal{L}_{\text{SI-SDR}} = -\text{SI-SDR}(\mathbf{s}, \hat{\mathbf{s}}) = -10 \log_{10} \left(\frac{\|\mathbf{s}_{\text{target}}\|_2^2}{\|\hat{\mathbf{s}} - \mathbf{s}_{\text{target}}\|_2^2} \right), \quad (158)$$

where the target projection is

$$\mathbf{s}_{\text{target}} = \frac{\langle \hat{\mathbf{s}}, \mathbf{s} \rangle}{\|\mathbf{s}\|_2^2} \mathbf{s}. \quad (159)$$

In addition to SI-SDR, we incorporate a magnitude loss to explicitly penalize magnitude mismatch between clean and estimated spectra:

$$\mathcal{L}_{\text{Mag}} = \frac{1}{TF} \sum_{t=1}^T \sum_{f=1}^F (|S(t, f)| - |\hat{S}(t, f)|)^2. \quad (160)$$

To further address phase-related discrepancies, we use a complex spectral loss defined on real and imaginary parts. Let $S_{\Re}(t, f) = \Re\{S(t, f)\}$, $S_{\Im}(t, f) = \Im\{S(t, f)\}$, $\hat{S}_{\Re}(t, f) = \Re\{\hat{S}(t, f)\}$, and $\hat{S}_{\Im}(t, f) = \Im\{\hat{S}(t, f)\}$.

$$\mathcal{L}_{\text{CS}} = \frac{1}{TF} \sum_{t=1}^T \sum_{f=1}^F \left((S_{\Re}(t, f) - \hat{S}_{\Re}(t, f))^2 + (S_{\Im}(t, f) - \hat{S}_{\Im}(t, f))^2 \right). \quad (161)$$

The total loss is

$$\mathcal{L}_{\text{total}} = \lambda_{\text{SI-SDR}} \mathcal{L}_{\text{SI-SDR}} + \lambda_{\text{Mag}} \mathcal{L}_{\text{Mag}} + \lambda_{\text{CS}} \mathcal{L}_{\text{CS}}, \quad (162)$$

where $\lambda_{\text{SI-SDR}}$, λ_{Mag} , and λ_{CS} balance contributions of the three terms.

5.3 Experimental Setup

5.3.1 Datasets

We evaluate the proposed model on three publicly available datasets with distinct acoustic characteristics. General dataset introductions are provided in Chapter 2; here we retain only the protocol-specific details that matter for this chapter and explain why each benchmark is used.

Simulated multichannel corpus (TIMIT + DNS Challenge + NOISEX-92). This corpus is used for controlled ablations on long-context modeling, cross-domain interaction, and seen-speaker versus unseen-speaker generalization. Following [91], we employ TIMIT for training, validation, and testing. The TIMIT test set is partitioned into (1) a seen-speaker subset (80 speakers present in training) and (2) an unseen-speaker subset (82 speakers not encountered during training). To simulate realistic noise conditions, 20,000 noise segments are randomly selected from the DNS Challenge dataset. Room impulse responses (RIRs) are generated using the image-source method with a uniform linear array of $M=5$ microphones spaced 4 cm apart. Room length and width are independently sampled from 3 m to 10 m, and room height is sampled from 2.5 m to 3 m. The reverberation time for a 60-dB decay (RT_{60}) is uniformly sampled between 0.16 s and 0.5 s.

The reference-channel SNR in decibels is defined as

$$\text{SNR}_{\text{dB}} = 10 \log_{10} \frac{\mathbb{E}_{t,f} \{|S(t, f)|^2\}}{\mathbb{E}_{t,f} \{|N(t, f) + R_{\text{rev}}(t, f)|^2\}},$$

where $S(t, f)$ is the reference-channel dry target spectrum, and $N(t, f)$ and $R_{\text{rev}}(t, f)$ denote, respectively, the additive-noise component and the excess reverberant component treated as disturbance in the STFT domain. The SNR is uniformly sampled from $[-5, 10]$ dB. For evaluation, we use two highly nonstationary noise signals (`babble` and `factory1`) from NOISEX-92, with tests conducted at SNR levels of -5 , 0 , 5 , and 10 dB. For all reverberant mixtures in this simulated setting, the supervision target $S(t, f)$ is the dry (anechoic) utterance, so reflections introduced by the RIR are treated as disturbance in the sense of the target definition introduced in Chapter 2.

CHiME-4 dataset. This benchmark is used because it stresses simulation-to-real mismatch and allows the chapter to evaluate both intrusive enhancement metrics on simulated data and practical

metrics on real recordings. We conduct experiments on the CHiME-4 Challenge dataset [41] at 16 kHz. Speech utterances are drawn from WSJ0 [92]. For training, we use anechoic clean utterances and generate each M -channel mixture by convolving the clean waveform $s[n]$ with simulated RIRs $h_m[n]$. For evaluation, we use the simulated test subset for intrusive enhancement metrics and the real-recorded subset for non-intrusive quality and ASR evaluation. To increase training diversity, we use the official CHiME scripts to generate simulated mixtures on the fly with SNR sampled from $[-5, 10]$ dB.

ConferencingSpeech 2021 Challenge. This dataset is included because the proposed S5-based architecture is intended for long-context modeling, and ConferencingSpeech provides long-form far-field multi-talker conditions that are difficult to capture with short utterance benchmarks alone. Training mixtures are generated on the fly by segmenting recordings into 8-second chunks and mixing with noise at SNRs from -5 dB to 25 dB, following the official protocol [43]. The simulated recordings are captured using an 8-element linear, non-uniform microphone array.

5.3.2 Parameter Setup and Training Strategy

All input signals are resampled to 16 kHz. We compute the STFT using a 25 ms window, 50% overlap, and a 512-point FFT (257 nonnegative-frequency bins).

We train on fixed-length segments. For the simulated set and the CHiME-4 simulated set, we randomly crop each utterance to 3 s, yielding approximately $T \approx 240$ frames under 50% overlap (the exact value depends on boundary padding and the STFT implementation). For ConferencingSpeech 2021, we use 8 s segments (approximately $T \approx 640$). Utterances shorter than the target length are zero-padded; longer ones are randomly cropped each epoch. At inference, the full utterance is processed without cropping.

The S3Conv encoder uses parallel convolutional operations. Specifically, a spatial convolution with kernel size 3 (along the microphone axis) with LeakyReLU activation captures inter-microphone correlations, while a TF convolution with a 5×1 kernel extracts local TF patterns at the same resolution. On the decoder side, each stream uses a dedicated decoder composed of three transposed-convolution layers with kernel sizes 5×1 , 1×5 , and 3×3 ; stride is set to 1 so that resolution is preserved and the layers act as learnable synthesis filters.

For optimization, we employ Adam with initial learning rate 10^{-3} . The learning rate is reduced by a factor of 2 if the validation loss does not improve for five consecutive epochs, until reaching a predefined minimum. Batch size is 16, and training runs for 50 epochs. These hyperparameters were selected on validation data from the simulated set and then reused for the CHiME-4 and ConferencingSpeech experiments.

We apply data augmentation during training: noise levels are randomized within $[-5, 10]$ dB, and RIRs are generated using the image-source method with random room dimensions and RT_{60} ranging from 0.16 s to 0.7 s. This wider RT_{60} range is used for training augmentation, while the controlled simulated evaluation described above uses $RT_{60} \in [0.16, 0.5]$ s. For reimplemented baselines used in the ablation studies, the same augmentation pipeline is applied when applicable. For RIR generation in the simulated setting, we use the same linear array geometry as evaluation (five microphones with 4 cm spacing).

Among the compared baselines, some models natively support multichannel input, whereas others are originally designed for single-channel enhancement. We therefore follow each baseline’s native input configuration and evaluation protocol whenever possible, and any additional wrapper or post-processing used for a particular baseline is stated explicitly in the corresponding experiment.

Table 5.1 lists key hyperparameter configurations:

Table 5.1: Symbol definitions and values (Chapter 5).

Symbol	Description	Value
C_{enc}	Number of output channels per microphone in the encoder	128
C_{spatial}	Channels in the spatial convolution pathway	128
C_{spectral}	Channels in the TF convolution pathway	128
d_u	Input dimension to each S5 block, $M \times F \times C_{\text{enc}}$	164,480
d_v	Hidden-state size in the S5 module	256
d_y	Output dimension of the S5 block (equal to d_u)	164,480
d_h	Hidden projection size used in the lightweight per-block channel-mixing layers	128
d_{CME}	Feature dimension in the CME branch	128
d_{CSR}	Feature dimension in the CSR branch	128
$\lambda_{\text{SI-SDR}}$	Weight of the SI-SDR loss	1.0
λ_{Mag}	Weight of the magnitude loss	0.5
λ_{CS}	Weight of the complex-spectrum loss	0.5

Note that $d_u = MFC_{\text{enc}}$; the numeric value above corresponds to the simulated configuration with $M=5$ and $F=257$. This value is the flattened feature dimension presented to the structured

S5 block, not the dimension of a fully dense state matrix. For CHiME-4 and ConferencingSpeech 2021, the same formula applies with the corresponding microphone count.

5.3.3 Evaluation Metrics

The metrics reported in this chapter depend on whether clean references are available and on whether the benchmark is enhancement-centric or downstream-task-centric. On the simulated corpora we report PESQ, STOI or ESTOI, and SDR or SI-SNR because clean targets are available and the main question is enhancement quality under long-context modeling. On the CHiME-4 real set we report DNSMOS and WER because intrusive references are unavailable and practical utility matters more than purely signal-level fidelity. On ConferencingSpeech 2021 we report PESQ, STOI, ESTOI, and SI-SNR to reflect both perceptual quality and long-context suppression performance.

5.4 Results and Analysis

5.4.1 Effect of Dual-Path Models

We conduct ablation experiments on the simulated multichannel dataset (TIMIT with simulated RIRs and NOISEX-92 noise). Noisy signals are generated using babble and factory noise at SNRs from -5 dB to 10 dB. Table 5.2 compares the unprocessed mixture, CME-only, CSR-only, and the full dual-path model.

Table 5.2: Effect of dual-path configurations.

Model	PESQ	ESTOI	SDR (dB)	SI-SNR (dB)
Unprocessed	1.49	0.251	-4.79	-0.25
CME Only	2.05	0.341	0.75	5.20
CSR Only	2.12	0.354	1.10	5.60
Proposed	3.06	0.599	7.90	10.30

Both single-branch variants (CME-only and CSR-only) provide notable improvement over the unprocessed noisy signal. However, combining both branches yields a substantial additional gain, achieving the best scores across all metrics. The proposed model reaches PESQ 3.06, ESTOI 0.599, SDR 7.90 dB, and SI-SNR 10.30 dB, supporting the use of coordinated coarse magnitude estimation and complex spectral refinement.

5.4.2 Effect of Temporal Sequence Models

Table 5.3 compares temporal sequence backbones adopted as alternatives to S5. We implement custom variants of Dual-S4 and Dual-S4D following [28, 29], a Dual-gLSTM model [93], a Dual-TCM model [94], a Dual-gGRU variant [95], and WaveNet [96]. We also report parameter counts and GMAC/s under a consistent setup (5 microphones, 8-second utterances, 512-point STFT with 50% overlap), computed using `ptflops` (v0.7.1, PyTorch 2.1.0) [61] from a single forward pass (batch=1) and normalized by utterance duration; STFT and iSTFT are excluded.

Compared with S4D-style stacks [28], our S5 layer [29] reduces computation by (i) using a diagonal $\mathbf{A}_d$, reducing the per-step update from $O(d_v^2)$ to $O(d_v)$; (ii) sharing one MIMO SSM per branch rather than replicating SISO SSMs across channels; (iii) evaluating the SSM recurrently rather than materializing long convolution kernels often used in S4D implementations; and (iv) keeping a modest state width d_v with lightweight 1×1 projections and a direct feed-through $\mathbf{D}_{S5}$.

We also evaluate different numbers of S5 stages: Single-S5 (one block), Double-S5 (two blocks; our default setting), and Triple-S5 (three blocks).

Table 5.3: Comparison of temporal sequence models in performance and computational complexity.

Model	PESQ	ESTOI	SDR (dB)	SI-SNR (dB)	Params (M)	GMAC/s
Dual-S4	2.68	0.449	5.41	8.20	7.80	140.9
Dual-S4D	2.74	0.453	5.59	8.45	9.25	162.7
Dual-gLSTM	2.36	0.423	4.40	7.19	14.60	195.4
Dual-TCM	2.49	0.434	4.79	7.83	26.10	226.8
Dual-gGRU	2.45	0.431	4.60	7.52	12.30	158.3
WaveNet	2.61	0.442	5.05	7.96	31.50	256.1
Single-S5	2.78	0.491	6.02	8.85	5.80	118.3
Double-S5 (default)	3.06	0.599	7.90	10.30	6.65	131.2
Triple-S5	3.08	0.604	8.13	10.58	8.32	155.7

All temporal models improve over the unprocessed noisy speech. Importantly, the Double-S5 configuration achieves a strong performance–efficiency trade-off. Although Triple-S5 yields slightly higher PESQ and ESTOI, it incurs higher cost (8.32 M parameters and 155.7 GMAC/s). Double-S5 achieves PESQ 3.06, ESTOI 0.599, SDR 7.90 dB, and SI-SNR 10.30 dB with 6.65 M parameters and 131.2 GMAC/s, making it preferable for deployment-oriented settings.

5.4.3 Effect of Cross-Domain Interaction Modules

Table 5.4 compares fusion methods for integrating CME and CSR information: feature concatenation, weighted fusion, self-attention, and the proposed cross-domain interaction module. The proposed module explicitly re-scales branch features and applies gating to control each stream’s contribution.

Table 5.4: Effect of cross-domain interaction modules.

Model	PESQ	ESTOI	SDR (dB)	SI-SNR (dB)	GMAC/s
Feature Concatenation	2.12	0.353	3.48	6.33	120.4
Weighted Feature Fusion	2.22	0.360	3.59	6.68	124.7
Self-Attention	2.49	0.453	4.95	8.08	138.2
Proposed	3.06	0.599	7.90	10.30	131.9

While concatenation and weighted fusion yield limited improvement, they lack structured inter-stream interaction. Self-attention improves fusion but remains below the proposed interaction module. Our method achieves the highest scores across all evaluation metrics. In addition, it requires 131.9 GMAC/s, approximately 5% lower than the self-attention variant (138.2 GMAC/s) while providing noticeably stronger enhancement.

5.4.4 Effect of Feature Source for Cross-Domain Interaction

The interaction module can operate on different feature stages. We compare two variants: removing interaction entirely (*No interaction*) and applying interaction to intermediate features produced before the final channel-mixing refinement in each stage-1 S5 block (*Intermediate features*).

Table 5.5: Ablation on feature source for the interaction module.

Feature Source	PESQ	ESTOI	SDR (dB)	SI-SNR (dB)
No cross-domain interaction	2.48	0.431	4.75	7.35
Intermediate features (ours)	3.06	0.598	7.90	10.30

Feeding intermediate features substantially improves all metrics, indicating that this representation retains richer magnitude–phase correlations that the interaction module can exploit for better enhancement.

5.4.5 Robustness to Microphone Permutation and Geometry

To verify that the model is not overly tied to a particular microphone order or array spacing, we conduct an additional ablation on the simulated multichannel dataset. We compare three test configurations: (i) the canonical 5-microphone linear array used during training, (ii) a single random microphone permutation per utterance, and (iii) a perturbed array geometry using RIRs generated with 6 cm spacing (instead of 4 cm).

Table 5.6: Robustness to channel permutation and geometry.

Configuration	PESQ	ESTOI	SDR (dB)	SI-SNR (dB)
Canonical	3.06	0.599	7.90	10.30
Random permutation	3.04	0.595	7.85	10.25
6 cm spacing	3.02	0.592	7.80	10.18

Performance differences are marginal across all metrics, indicating robustness to microphone re-ordering and moderate geometry perturbations within the fixed- M setting considered in this chapter. This behavior is consistent with the random microphone-permutation augmentation applied during training, even though the model is not explicitly permutation-equivariant by design.

5.4.6 Results on Seen and Unseen Speaker Test Sets

We evaluate the proposed model on seen and unseen speaker test sets using the simulated multichannel dataset. Tables 5.7 and 5.8 report PESQ, ESTOI, and SDR separately at each SNR. Table 5.7 reports seen-speaker results separately for Babble and Factory1, while Table 5.8 reports the aggregated unseen-speaker results over both noises.

As shown in Table 5.7, the proposed model achieves the strongest PESQ and ESTOI under both Babble and Factory1 at all evaluated SNRs. For SDR, the proposed model is strongest from -5 to 5 dB, while FaSNet is slightly higher at the easiest 10 dB point. This pattern indicates that the proposed model is particularly effective in the more difficult low-to-medium SNR conditions.

The clear PESQ margin over Conv-TasNet and MIMO-UNet suggests that the dual-path decomposition (CME + CSR) better preserves speech naturalness while suppressing noise and reverberation. Compared with heavier baselines, the proposed model also offers a favorable performance–efficiency trade-off (6.65 M parameters and 131.2 GMAC/s).

Table 5.7: Results on the seen-speaker test set (TIMIT + NOISEX-92 with simulated RIRs).

Noise	Method	PESQ				ESTOI				SDR (dB)				Params	GMAC/s
		-5	0	5	10	-5	0	5	10	-5	0	5	10	(M)	(/s)
<i>Babble</i>	Unprocessed	1.49	1.82	2.16	2.52	0.251	0.378	0.515	0.650	-4.79	0.10	5.07	10.07	–	–
	Conv-TasNet	1.66	2.16	2.59	2.97	0.371	0.544	0.685	0.798	1.89	6.85	10.89	14.66	5.12	178.4
	GCCTN	2.01	2.39	2.76	3.11	0.434	0.573	0.709	0.808	4.40	8.51	12.12	15.43	3.87	142.7
	MIMO-UNet	1.72	2.26	2.62	3.05	0.395	0.554	0.672	0.782	2.28	7.43	10.95	14.69	8.25	197.3
	FaSNet	2.06	2.45	2.82	3.18	0.454	0.597	0.729	0.828	5.40	9.45	12.85	16.09	2.74	79.8
	Proposed	2.35	2.89	3.33	3.66	0.599	0.733	0.826	0.892	7.90	10.30	13.20	15.97	6.65	131.2
<i>Factory1</i>	Unprocessed	1.32	1.58	1.93	2.27	0.193	0.340	0.467	0.636	-4.80	0.10	5.07	10.06	–	–
	Conv-TasNet	1.68	2.01	2.51	2.88	0.309	0.484	0.660	0.769	3.36	7.65	11.37	14.98	5.12	178.4
	GCCTN	2.00	2.29	2.61	2.92	0.398	0.557	0.684	0.802	5.20	9.13	12.46	15.67	3.87	142.7
	MIMO-UNet	1.91	2.37	2.78	3.15	0.392	0.547	0.682	0.790	5.83	9.38	12.31	15.33	8.25	197.3
	FaSNet	1.87	2.37	2.65	2.96	0.408	0.556	0.710	0.798	5.72	10.06	13.27	16.32	2.74	79.8
	Proposed	2.38	2.93	3.38	3.71	0.571	0.692	0.796	0.877	8.75	10.55	13.35	16.19	6.65	131.2

Table 5.8: Results on the unseen-speaker test set (TIMIT + NOISEX-92).

Metrics	PESQ				ESTOI				SDR (dB)			
SNR (dB)	-5	0	5	10	-5	0	5	10	-5	0	5	10
Unprocessed	1.30	1.62	1.99	2.36	0.216	0.347	0.498	0.644	-4.80	0.10	5.08	10.05
Conv-TasNet	1.68	2.02	2.57	2.69	0.326	0.519	0.644	0.765	3.89	6.85	10.89	14.66
GCCTN	1.93	2.40	2.73	2.71	0.371	0.544	0.685	0.798	5.42	7.64	11.70	15.53
MIMO-UNet	1.85	2.31	2.75	3.01	0.387	0.560	0.697	0.806	5.82	9.31	12.22	15.48
FaSNet	2.01	2.41	2.79	3.13	0.430	0.601	0.737	0.836	6.46	10.27	13.36	16.41
Proposed	2.50	2.95	3.32	3.65	0.580	0.746	0.828	0.890	8.75	12.35	15.12	17.95

On the unseen-speaker set (Table 5.8), the proposed model maintains the strongest performance across all SNRs and metrics, demonstrating robust generalization.

DNSMOS scores in Table 5.9 further confirm perceptual improvements: the proposed model gives the highest DNSMOS at every SNR.

Table 5.9: DNSMOS on the unseen-speaker test set (TIMIT + NOISEX-92).

Metrics	DNSMOS			
SNR (dB)	-5	0	5	10
Unprocessed	2.04	2.12	2.31	2.62
Conv-TasNet	2.51	2.66	2.93	3.17
GCCTN	2.51	2.69	2.94	3.25
MIMO-UNet	2.60	2.81	3.02	3.23
FaSNet	2.52	2.77	3.03	3.31
Proposed	3.18	3.61	3.75	3.85

Table 5.10: Comparative results on 6-channel simulated CHiME-4.

Method	NB-PESQ	WB-PESQ	STOI	SI-SDR (dB)	SDR (dB)
Noisy (unprocessed)	1.82	1.27	0.87	7.5	7.5
MNMF Beamforming [80]	–	–	0.94	–	16.2
FasNet+TAC [63]	2.53	1.91	0.94	16.0	16.6
NBDF [82]	2.74	2.31	0.95	15.7	16.6
FT-JNF [6]	3.20	2.61	0.97	17.6	18.3
McNet [83]	3.38	2.73	0.98	19.2	19.6
Wang et al. [4]	–	–	0.99	22.0	22.4
Oracle MVDR	2.49	1.94	0.97	17.3	17.7
SpatialNet [84]	3.49	2.88	0.98	22.1	22.3
TADRNet [97]	2.68	2.13	0.99	19.7	21.2
SEMamba [98]	3.50	2.90	0.99	20.2	21.2
USES [99]	–	3.16	0.98	–	20.6
Proposed	3.83	3.25	0.99	25.4	25.6

5.4.7 Results on CHiME-4 Dataset

Table 5.10 reports performance on the 6-channel simulated CHiME-4 set, including classical beamformers (MNMF, Oracle MVDR), mask-based neural beamformers (NBDF, FasNet+TAC, FT-JNF), recent CNN/Transformer baselines (McNet, TADRNet, SpatialNet), and state-space models (SEMamba, USES).

The proposed dual-path S5 achieves the best results across every metric. It improves NB-PESQ to 3.83 (vs. 3.49 for SpatialNet) and WB-PESQ to 3.25 (vs. 2.88). The SI-SDR improvement is substantial at 25.4 dB, which is 3.3 dB above SpatialNet and 5.2 dB above SEMamba. These gains support the effectiveness of separating coarse magnitude estimation from fine complex-spectrum refinement and enabling intermediate cross-domain interaction.

Table 5.11 reports results on the CHiME-4 real test set. Following common practice, we evaluate ASR WER with the official CHiME-4 recipe and report non-intrusive perceptual quality using DNSMOS, since clean references are unavailable for intrusive evaluation on the real subset.

Our model attains the highest DNSMOS (3.20), substantially higher than Noisy (1.46) and USES (2.94). It improves perceptual quality without compromising ASR robustness: WER decreases from 11.0% (USES) to 6.3% (ours), an absolute reduction of 4.7 percentage points, while remaining close to the best WER achieved by Oracle MVDR (6.1%).

Table 5.11: DNSMOS and ASR results on the CHiME-4 real test set.

Method	WER (%)	DNSMOS
Noisy (unprocessed)	6.7	1.46
Oracle MVDR	6.1	2.10
SpatialNet [84]	7.2	3.05
USES [99]	11.0	2.94
Proposed	6.3	3.20

5.4.8 Results on ConferencingSpeech 2021 Development Set

Table 5.12 presents performance on the ConferencingSpeech 2021 development set using PESQ, STOI, ESTOI, and SI-SNR.

Table 5.12: Results on the ConferencingSpeech 2021 development set.

Model	PESQ	STOI	ESTOI	SI-SNR (dB)
Noisy	1.52	0.82	0.69	4.47
Baseline	2.00	0.89	0.78	9.16
MIMO-Unet	2.24	0.91	0.82	9.29
Spatial-DCCRN [70]	2.52	0.92	0.85	10.17
Proposed	2.68	0.93	0.86	10.66

The proposed model outperforms all baselines across all metrics. In particular, it improves PESQ to 2.68 (vs. 2.52 for Spatial-DCCRN) and achieves the highest STOI, ESTOI, and SI-SNR, indicating stronger perceptual quality, intelligibility, and noise suppression in multi-talker scenarios.

5.5 Conclusion

This chapter addressed a bottleneck that persists even when spatial filtering is explicit and stable: residual interference and reverberation tails often span long contexts, and phase-insensitive designs can leave artifacts that are difficult to remove with short-context refinement. To tackle this gap, we proposed an end-to-end multichannel framework based on dual-path state-space modeling with cross-domain interaction, targeting both efficient long-context modeling and coordinated magnitude–phase (complex-spectrum) enhancement.

The proposed system couples an S3Conv spatial–spectral encoder [90] with two parallel S5-based sequence modeling streams [29]: a coarse magnitude estimation (CME) stream for robust

suppression and amplitude restoration, and a complex spectral refinement (CSR) stream for fine spectral detail and phase-related cues. A key design choice is to exchange intermediate representations between the two streams, so complementary amplitude and complex-spectrum cues are aligned before reconstruction rather than fused only at the end. Evaluations on simulated and real multichannel benchmarks show consistent gains, and ablations attribute the improvements to the dual-path decomposition, the efficiency of S5 for long-context modeling, and the effectiveness of intermediate cross-domain interaction compared with late or loosely coupled fusion.

At the same time, the CHiME-4 real-set results highlight a deployment-facing issue that motivates the next chapter: improvements in enhancement and perceptual metrics do not necessarily translate into the best downstream recognition performance when the recognizer is frozen. Chapter 6 therefore shifts from enhancement-centric modeling to recognition-centric front-end design, emphasizing artifact control and observation fusion under frozen-ASR pipelines, with the goal of obtaining reliable WER gains without requiring back-end adaptation.

Chapter 6

Efficient Front-End Speech Enhancement for Robust ASR via Parallel Time–Band Mixing and Learned Observation Fusion

6.1 Introduction

Chapters 3–5 developed enhancement front ends that progressively improve spatial robustness, deployability under mismatch, and long-context modeling for difficult acoustic scenes. This chapter turns to a recognition-centric deployment setting: a speech-enhancement front end placed before a frozen automatic speech recognition (ASR) back end. In this setting, improvements in enhancement metrics or perceptual quality do not necessarily translate into lower word error rate (WER), because the recognizer can be disproportionately sensitive to enhancement artifacts and processing distortion.

Unlike Chapters 3–5, which focus primarily on multichannel speech enhancement, this chapter studies a single-channel enhancement front end placed before a frozen ASR back end. Robust ASR in real acoustic environments remains challenging due to nonstationary noise, reverberation, and

domain mismatch. Although large-scale pre-trained recognizers such as Whisper improve generalization, error rates can still increase sharply under adverse conditions [9]. In many deployments the recognizer is treated as a fixed component because it is proprietary, difficult to retrain, or too costly to fine-tune, so the speech front end becomes the primary means for improving robustness without modifying the ASR back end.

A straightforward strategy is to place a speech enhancement (SE) module before ASR, but this often exposes a critical mismatch: enhancement that improves perceptual or signal-level metrics does not necessarily reduce WER. Under a frozen recognizer, nonlinear enhancement may introduce artifacts or processing distortion that is disproportionately harmful to recognition [30, 100, 101]. At the same time, efficiency is a primary deployment constraint: modern ASR back ends are already compute-heavy, so additional front-end latency or limited front-end parallelism can become a deployment bottleneck [33, 34].

This chapter addresses the gap with a frozen-recognizer front end designed around two deployment requirements: parallel-friendly enhancement and recognition-stable synthesis without development-set coefficient tuning. We propose a Parallel Time–Band Mixer (PTBM) backbone that avoids within-block recurrent or sequential scans by combining parallel temporal mixing and per-frame cross-band interaction on a compact band-split grid. On top of the enhanced estimate, we introduce ASR-oriented learned observation fusion (LOF), which predicts band–frame fusion weights and performs structured complex-spectrum fusion between the enhanced signal and the original observation, allowing adaptive interpolation toward the original observation in artifact-prone regions. During training, we optionally include a lightweight auxiliary connectionist temporal classification (CTC)-based consistency objective (Aux-CTC) to discourage recognition-unstable distortions while keeping downstream ASR back ends frozen; this term is used only during training and does not change inference complexity.

The remainder of this chapter is organized as follows. Section 6.2 presents the proposed PTBM–LOF front end and learning objectives. Section 6.3 describes datasets, training, and decoding protocols under frozen recognizers. Section 6.4 reports recognition results, ablations, and analysis. Finally, Section 6.5 concludes this chapter and transitions to the thesis-level conclusions.

6.2 Proposed Method

6.2.1 System Overview

Figure 6.1 illustrates the proposed PTBM–LOF front end under a frozen ASR back end. Let $x[n]$ denote a single-channel noisy waveform and let $X(t, f) \in \mathbb{C}$ denote its complex STFT coefficient at frame t and nonnegative-frequency bin $f \in \mathcal{F}_+$. Stacking over all time frames and processed frequency bins gives $X \in \mathbb{C}^{T \times F}$, where T and F are the numbers of time frames and retained nonnegative-frequency bins, respectively.

We convert X into a band-split representation $Z^{(0)} \in \mathbb{R}^{T \times B \times D}$ by grouping the F frequency bins into B non-overlapping sub-bands and linearly embedding the real and imaginary parts within each sub-band, where D is the sub-band embedding dimension. We process $Z^{(0)}$ using N_{blk} stacked Parallel Time–Band Mixer (PTBM) blocks to obtain the final backbone representation $Z = Z^{(N_{\text{blk}})}$.

We then employ three prediction heads. Two reconstruction heads map Z to a predicted complex mask $\widehat{M} \in \mathbb{C}^{T \times F}$ and a complex residual $R \in \mathbb{C}^{T \times F}$ on the STFT grid, respectively, which are combined to form an enhanced spectrum $\widehat{S}$. We further introduce a LOF head that predicts a band–frame fusion map to synthesize a fused spectrum Y by combining $\widehat{S}$ and X within each sub-band. Applying inverse STFT to Y yields the waveform $y[n]$ fed to the frozen ASR back end.

During training, we keep the ASR back end frozen and optionally add an Aux-CTC loss computed with a frozen CTC teacher [102]. The teacher parameters are fixed; gradients from this loss are propagated only to the front end through the synthesized waveform, and the auxiliary branch is removed at inference.

6.2.2 Band-Split Tokenizer

We compute the N -point complex STFT using an N -sample analysis window with hop size H , yielding $X \in \mathbb{C}^{T \times F}$ with $F = N/2 + 1$ nonnegative-frequency bins per frame. We partition the F frequency bins into B contiguous non-overlapping sub-bands with sizes $\{F_b\}_{b=1}^B$ satisfying $\sum_{b=1}^B F_b = F$. Let $X_b \in \mathbb{C}^{T \times F_b}$ denote the sub-band spectrogram extracted from X by selecting the bins assigned to sub-band b .

For frame t and sub-band b , we concatenate the real and imaginary parts of $X_b(t, :)$ to form a

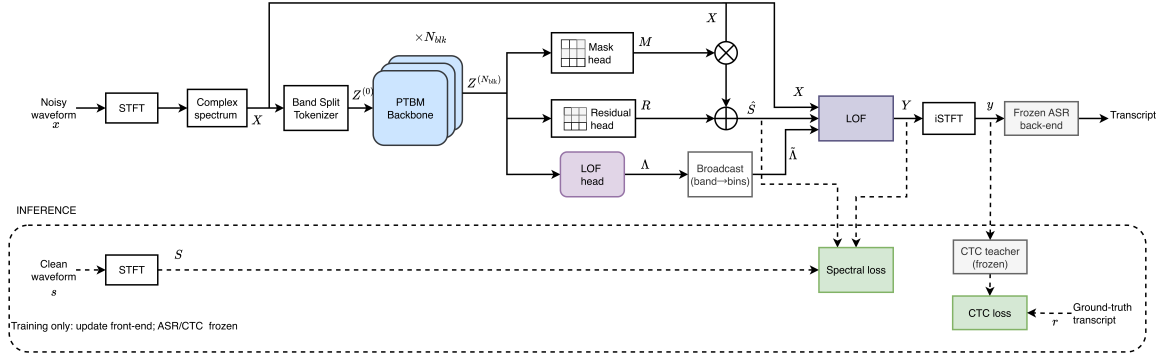


Figure 6.1: Overall architecture of the proposed PTBM-LOF front end under a frozen ASR back end. The schematic mask label M in the source figure corresponds to the predicted complex mask $\hat{M}$ in the text. The enhanced spectrum before learned observation fusion is denoted by $\hat{S}$, and the fused spectrum passed to the recognizer is denoted by Y , consistent with Eqs. (174) and (176). Solid boxes and arrows denote the inference path from the noisy waveform to the fused waveform passed to the recognizer. Dashed boxes or arrows denote training-only components, including spectral supervision on $\hat{S}$ and Y and an auxiliary CTC consistency loss using a frozen CTC teacher. Only the front-end parameters are updated during training; the ASR back end and the CTC teacher remain frozen.

real-valued sub-band token,

$$\mathbf{x}_b(t) = [\Re\{X_b(t, :)\}^\top, \Im\{X_b(t, :)\}^\top]^\top \in \mathbb{R}^{2F_b}. \quad (163)$$

Each token is mapped to a fixed embedding dimension D using a sub-band-specific linear projection,

$$\mathbf{z}_{t,b}^{(0)} = \left(W_b^{\text{tok}}\right)^\top \mathbf{x}_b(t) + \mathbf{e}_b \in \mathbb{R}^D, \quad (164)$$

where $W_b^{\text{tok}} \in \mathbb{R}^{2F_b \times D}$ is the learnable projection matrix for sub-band b and $\mathbf{e}_b \in \mathbb{R}^D$ is a learnable sub-band embedding that encodes sub-band order. Stacking $\mathbf{z}_{t,b}^{(0)}$ over all frames and sub-bands produces $Z^{(0)} \in \mathbb{R}^{T \times B \times D}$, with $Z^{(0)}(t, b, :)$ equal to the corresponding embedded token. This tensor serves as the input to the PTBM backbone.

6.2.3 Parallel Time-Band Mixer Backbone

The enhancement backbone is constructed by stacking N_{blk} Parallel Time-Band Mixer (PTBM) blocks (Figure 6.2). Each block maps $Z^{(\ell)}$ to $Z^{(\ell+1)}$, where $\ell \in \{0, \dots, N_{\text{blk}} - 1\}$ indexes the

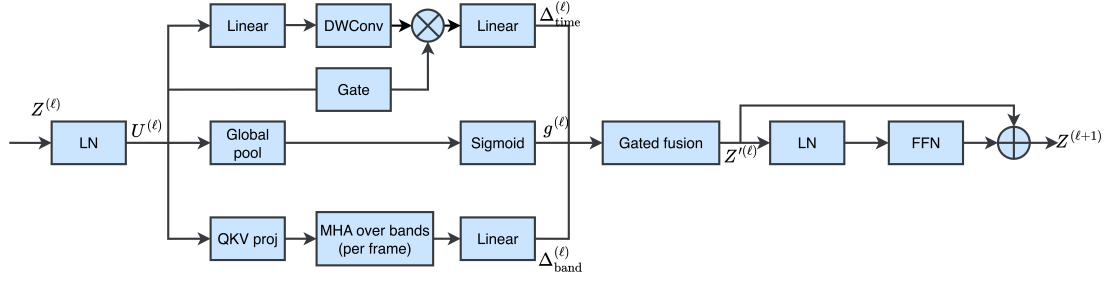


Figure 6.2: One PTBM block. After pre-normalization, temporal mixing and per-frame cross-band mixing run in parallel, corresponding to Eqs. (166) and (169), respectively. The scalar mixing gate is $g^{(\ell)}$ in Eq. (171), followed by the pre-normalized residual FFN in Eq. (173).

block. We set $Z^{(0)}$ as the tokenizer output and use $Z^{(N_{\text{blk}})}$ as the final backbone representation. A PTBM block contains two sequence-parallel branches: (i) a Temporal ConvMixer (TCM) branch that performs intra-band temporal mixing within each sub-band, and (ii) a Cross-Band Attention Mixer (CBA) branch that models per-frame cross-band interaction via self-attention over the B sub-band tokens. An interaction module fuses the two branch updates, followed by a lightweight channel-mixing feed-forward module (FFN).

We adopt a pre-normalized formulation:

$$U^{(\ell)} = \text{LN}\left(Z^{(\ell)}\right), \quad (165)$$

where $\text{LN}(\cdot)$ denotes layer normalization [103] applied along the channel dimension.

Temporal ConvMixer (TCM)

The TCM branch processes each sub-band independently. Let $\text{DWConv}_{d_\ell}(\cdot)$ denote a depth-wise 1-D convolution along the time axis with kernel size k and dilation d_ℓ , applied independently for each sub-band and bottleneck channel, with padding chosen to preserve the frame length T . We project $U^{(\ell)}$ to a bottleneck dimension D_b and form a gated temporal update:

$$\Delta_{\text{time}}^{(\ell)} = \left(\text{DWConv}_{d_\ell} \left(U^{(\ell)} W_i^{(\ell)} \right) \odot \sigma \left(U^{(\ell)} W_g^{(\ell)} \right) \right) W_{o,t}^{(\ell)}, \quad (166)$$

where $W_i^{(\ell)}, W_g^{(\ell)} \in \mathbb{R}^{D \times D_b}$ and $W_{o,t}^{(\ell)} \in \mathbb{R}^{D_b \times D}$ are learned pointwise linear projections, $\sigma(\cdot)$ is the sigmoid, and $\odot$ denotes element-wise product. Thus $\Delta_{\text{time}}^{(\ell)} \in \mathbb{R}^{T \times B \times D}$.

Cross-Band Attention Mixer (CBA)

The CBA branch models interactions across the B sub-bands independently at each time frame. For a fixed frame t , let $U_t^{(\ell)} = U^{(\ell)}(t, :, :) \in \mathbb{R}^{B \times D}$ denote the sub-band-token matrix. We compute multi-head self-attention (MHA) [104] over the sub-band axis using bottleneck projections. For frame t , we form query, key, and value matrices $Q_t^{(\ell)}, K_t^{(\ell)}, V_t^{(\ell)} \in \mathbb{R}^{B \times D_b}$ via

$$Q_t^{(\ell)} = U_t^{(\ell)} W_Q^{(\ell)}, \quad K_t^{(\ell)} = U_t^{(\ell)} W_K^{(\ell)}, \quad V_t^{(\ell)} = U_t^{(\ell)} W_V^{(\ell)}, \quad (167)$$

where $W_Q^{(\ell)}, W_K^{(\ell)}, W_V^{(\ell)} \in \mathbb{R}^{D \times D_b}$ are learned projections. We use h attention heads and split the bottleneck dimension as $d_h = D_b/h$ (with D_b divisible by h). Accordingly, we reshape $Q_t^{(\ell)}, K_t^{(\ell)}, V_t^{(\ell)} \in \mathbb{R}^{B \times D_b}$ into h head slices $Q_t^{(\ell,i)}, K_t^{(\ell,i)}, V_t^{(\ell,i)} \in \mathbb{R}^{B \times d_h}$ for $i = 1, \dots, h$.

The per-head scaled dot-product attention is

$$\text{Attn}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_h}}\right) V, \quad (168)$$

where the softmax is applied row-wise over the sub-band dimension. We concatenate the h head outputs and project back to D to obtain the cross-band update at frame t ,

$$\begin{aligned} \Delta_{\text{band}}^{(\ell)}(t, :, :) = & \text{Concat}(\text{Attn}(Q_t^{(\ell,1)}, K_t^{(\ell,1)}, V_t^{(\ell,1)}), \\ & \dots, \text{Attn}(Q_t^{(\ell,h)}, K_t^{(\ell,h)}, V_t^{(\ell,h)})) W_{o,b}^{(\ell)}, \end{aligned} \quad (169)$$

where $W_{o,b}^{(\ell)} \in \mathbb{R}^{D_b \times D}$ projects the concatenated head outputs back to dimension D . Stacking over all frames yields $\Delta_{\text{band}}^{(\ell)} \in \mathbb{R}^{T \times B \times D}$.

Interaction (Gate-and-Fusion)

We fuse the two branch updates using a block-level scalar mixing gate shared across all frames and sub-bands:

$$p^{(\ell)} = \text{Pool}\left(U^{(\ell)}\right) \in \mathbb{R}^D, \quad (170)$$

$$\text{Pool}(U) = \frac{1}{TB} \sum_{t=1}^T \sum_{b=1}^B U(t, b, :),$$

$$g^{(\ell)} = \sigma\left((w^{(\ell)})^\top p^{(\ell)} + c^{(\ell)}\right), \quad (171)$$

where $w^{(\ell)} \in \mathbb{R}^D$ and $c^{(\ell)} \in \mathbb{R}$ are learned parameters. The fused residual update is

$$Z'^{(\ell)} = Z^{(\ell)} + g^{(\ell)} \Delta_{\text{time}}^{(\ell)} + (1 - g^{(\ell)}) \Delta_{\text{band}}^{(\ell)}. \quad (172)$$

Finally, we apply a channel-mixing FFN with pre-normalization and a residual connection:

$$Z^{(\ell+1)} = Z'^{(\ell)} + \text{FFN}\left(\text{LN}\left(Z'^{(\ell)}\right)\right). \quad (173)$$

Reconstruction Interface

Following the final PTBM block, we reconstruct a full-resolution complex spectrum on the STFT grid using a mask-plus-residual interface with two parallel prediction heads: a mask head and a residual head. We assemble $\widehat{M}, R \in \mathbb{C}^{T \times F}$ on the STFT grid and reconstruct

$$\widehat{S} = \widehat{M} \odot X + R, \quad (174)$$

where $\odot$ denotes element-wise multiplication on the STFT grid. Applying inverse STFT to $\widehat{S}$ yields the enhanced-only waveform $\widehat{s}[n]$. Throughout this chapter, $\widehat{S}(t, f)$ denotes the enhanced spectrum before observation fusion, while $Y(t, f)$ denotes the fused spectrum actually passed to the frozen recognizer in LOF variants.

6.2.4 Learned Observation Fusion for Recognition Stability

To mitigate ASR-sensitive enhancement artifacts under a frozen recognizer, we fuse the enhanced spectrum $\hat{S}$ with the noisy spectrum X on the STFT grid. LOF predicts a structured band-frame fusion map on the band-split grid and applies it on the STFT grid by expanding band-level weights to frequency-bin weights, enabling time-varying interpolation toward the original observation without development-set coefficient tuning.

Let $Z \in \mathbb{R}^{T \times B \times D}$ denote the final band-split representation produced by the PTBM backbone. A lightweight prediction head $f_{\text{lof}} : \mathbb{R}^D \rightarrow \mathbb{R}$ is applied pointwise to each token $Z(t, b, :)$ to produce a scalar logit, yielding

$$\Lambda = \sigma(f_{\text{lof}}(Z)) \in [0, 1]^{T \times B}. \quad (175)$$

A value of $\Lambda(t, b)$ close to one indicates greater reliance on the enhanced estimate in sub-band b at frame t , whereas a value close to zero indicates stronger interpolation toward the original observation.

To apply fusion on the STFT grid, we expand each sub-band weight to all frequency bins belonging to that sub-band, yielding $\tilde{\Lambda} \in [0, 1]^{T \times F}$. Let $b(f) \in \{1, \dots, B\}$ denote the sub-band index of frequency bin f ; we define $\tilde{\Lambda}(t, f) = \Lambda(t, b(f))$, which is piecewise-constant along frequency within each sub-band.

The synthesized complex spectrum is then computed by

$$Y = \tilde{\Lambda} \odot \hat{S} + (\mathbf{1} - \tilde{\Lambda}) \odot X, \quad (176)$$

where $\mathbf{1}$ is an all-ones tensor with the same dimensions as $\tilde{\Lambda}$. Applying inverse STFT to Y yields the synthesized waveform $y[n]$, which is fed to the frozen recognizer. Since $\tilde{\Lambda}(t, f) \in [0, 1]$, Eq. (176) forms a pointwise convex combination of $\hat{S}$ and X on the complex STFT grid. A global scalar fusion is a special case obtained by setting $\Lambda(t, b) = \omega$ for all (t, b) .

6.2.5 Frozen Recognizer Interface and Learning Objective

The proposed front end produces two waveform-level outputs: the enhanced estimate $\hat{s}[n]$ obtained from $\hat{S}$, and the LOF-synthesized waveform $y[n]$ obtained from Y . In ASR evaluations, A0 feeds $\hat{s}[n]$ to the downstream recognizer, whereas LOF variants feed $y[n]$. The recognizer remains frozen and is not fine-tuned.

For training, we optimize the front end with paired clean and noisy data while keeping all recognizer parameters fixed. To keep notation compact, we omit explicit crop-level and utterance-level superscripts in the objective below. In the two spectral terms, $\hat{S}$, Y , and S denote the enhanced spectrum, LOF output spectrum, and clean reference spectrum for the training segment used by the spectral losses. In the auxiliary CTC term, $y[n]$ and $\mathbf{r}$ denote the LOF-synthesized waveform and transcript or token sequence used by the CTC teacher. We minimize

$$\mathcal{L} = \mathcal{L}_{\text{spec}}(\hat{S}, S) + \alpha \mathcal{L}_{\text{spec}}(Y, S) + \beta \mathcal{L}_{\text{CTC}}(\mathcal{R}_{\text{CTC}}(y), \mathbf{r}), \quad (177)$$

where α and β weight the fused-spectrum supervision and the auxiliary recognition-consistency term, respectively. The first two terms train the enhancement and fused spectra toward the clean reference. When random spectral crops are used, these terms are evaluated on the crop spectra; when Aux-CTC is enabled, the CTC term is evaluated on the corresponding transcript-bearing waveform segment according to the training protocol, so the transcript span is matched to the waveform passed to the teacher.

The mapping $\mathcal{R}_{\text{CTC}}(\cdot)$ denotes a frozen CTC teacher used only during training, and $\mathcal{L}_{\text{CTC}}(\cdot, \cdot)$ is the standard CTC loss [102]. The teacher parameters are not updated. Gradients from the CTC loss are back-propagated through the teacher input to the front end, so only the front-end parameters are optimized. We supervise both $\hat{S}$ and Y to avoid a degenerate solution in which LOF relies excessively on the observation and leaves the enhancement branch under-trained.

The spectral reconstruction loss is

$$\mathcal{L}_{\text{spec}}(A, S) = \|\Re(A - S)\|_1 + \|\Im(A - S)\|_1 + \gamma \||A| - |S|\|_1, \quad (178)$$

where $\Re(\cdot)$ and $\Im(\cdot)$ denote real and imaginary parts, $|\cdot|$ denotes complex magnitude, and each ℓ_1 term denotes the mean absolute value over the corresponding time–frequency grid. The scalar weight γ controls the contribution of the magnitude term.

6.2.6 Complexity Analysis

We analyze the computational complexity of the proposed PTBM–LOF front end in terms of multiply–accumulate operations (MACs) per second of audio. Let T_{fps} denote the number of STFT frames per second. Table 6.1 summarizes the leading-order neural front-end terms, up to constant factors; the PTBM rows are reported per layer, and the overall cost scales linearly with N_{blk} . The table excludes the frozen ASR back end and omits the FFT/iSTFT implementation details that are shared across STFT-domain front ends.

Table 6.1: Leading-order MAC complexity per second (front end only).

Module	Asymptotic Complexity
Tokenization	$\mathcal{O}(T_{\text{fps}}FD)$
PTBM Block (per layer)	
Temporal Mixing	$\mathcal{O}(T_{\text{fps}}BDD_b + T_{\text{fps}}BD_bk)$
Cross-Band Attention	$\mathcal{O}(T_{\text{fps}}BDD_b + T_{\text{fps}}B^2D_b)$
Channel FFN	$\mathcal{O}(T_{\text{fps}}BrD^2)$
Reconstruction and Fusion	
LOF Predictor (f_{lof})	$\mathcal{O}(T_{\text{fps}}BD)$
Mask and Residual Heads	$\mathcal{O}(T_{\text{fps}}FD)$

6.3 Experimental Setup

6.3.1 Datasets and Evaluation Protocols

Datasets

To enable direct comparison with recent robust-ASR front-end studies, we adopt the same dataset splits and mixture-generation procedures as the corresponding reference protocols whenever applicable, and avoid introducing incompatible data pipelines. Table 6.2 summarizes all evaluation tracks and the key protocol constraints. The benchmarks are chosen to answer three distinct

questions: (i) can the front end improve WER under standard frozen-ASR benchmarks, (ii) does the benefit persist under SNR-stratified and out-of-domain stress tests, and (iii) does the approach transfer across language and back-end. Unless explicitly stated otherwise, we train a separate front-end model for each primary benchmark track using its corresponding training pipeline. For evaluation-only stress tests (Aurora-4 and VoiceBank-DEMAND), we directly evaluate the LibriSpeech+DNS-trained front end without additional tuning.

First, we use the Deep Noise Suppression (DNS) Challenge test sets, with and without reverberation, [40] and CHiME-4 real recordings [42] as primary benchmarks to quantify recognition robustness under controlled acoustic conditions and real-world noise. For CHiME-4, we intentionally adopt two reporting conventions: the 1-channel real-data protocol in [34] and the `et_simu/et_real` Channel 5 protocol in [105], enabling direct comparison against both lightweight front-end designs and recent bridging-module systems.

Second, we include analysis and generalization stress tests following [106]. The LibriSpeech+DNS track [40, 48] provides controlled SNR stratification to diagnose over-suppression and artifact sensitivity at higher SNR. Specifically, we construct five SNR bins at $\{-12, -6, 0, 6, 12\}$ dB and report WER for each bin, matching the breakdown in [106]. Aurora-4 [49] and VoiceBank-DEMAND [44, 107] are used as evaluation-only mismatched-condition tests: unless explicitly stated, we do not tune front-end hyperparameters on these sets, and use them to assess out-of-domain robustness under fixed training settings.

Third, we assess cross-lingual portability using the Mandarin protocol in [34]. Noisy Mandarin train and dev sets are generated by mixing AISHELL-1 speech [50] with DNS noises over a wide SNR range, and evaluation is performed on real in-house mobile recordings in two acoustic environments (Kitchen and Mall), with stronger reverberation in the Mall condition. For the in-house Mandarin evaluation, each environment (Kitchen and Mall) consists of three predefined subsets (t1–t3) corresponding to different recording sessions or devices, following the protocol of [34]. We report per-subset character error rate (CER) for t1–t3 in each environment. The in-house Mandarin evaluation is used as a portability case study on real mobile recordings; our main conclusions are supported by the public benchmarks above and do not rely on the in-house dataset.

Table 6.2: Summary of datasets and evaluation protocols.

Track / Condition	Lang.	Training and Dev Protocol (Front End)	Evaluation Sets / Metric
<i>I. Primary Robust ASR Benchmarks</i>			
DNS (Non-reverb / Reverb)	EN	DNS recipe and two-condition protocol in [34]; SNR from -5 to 20 dB.	DNS synthetic test sets (Non-reverb & Reverb); WER.
CHiME-4 (Real, 1-ch)	EN	Online simulation [34]: <code>tr05_simu_1ch</code> + DNS noises; SNR -5 to 20 dB.	<code>dt05_real</code> (dev) and <code>et05_real</code> (eval); WER.
CHiME-4 (Bridge)	EN	Channel 5 protocol for bridging-module comparison in [105].	Channel 5 of <code>et_simu</code> and <code>et_real</code> ; WER.
<i>II. Analysis and Generalization Stress Tests</i>			
LibriSpeech + DNS (SNR breakdown)	EN	Trained on LibriSpeech <code>train-clean-360</code> + DNS noise (train split) following [106].	<code>test-clean</code> + held-out DNS noises at predefined SNR bins; WER.
Aurora-4 (Unseen)	EN	Evaluation-only: directly evaluate the LibriSpeech+DNS-trained model.	Official Aurora-4 noisy test sets; WER.
VoiceBank-DEMAND	EN	Evaluation-only: directly evaluate the LibriSpeech+DNS-trained model.	Official test set; WER and enhancement metrics.
<i>III. Cross-lingual Evaluation</i>			
Mandarin (AISHELL-1)	ZH	AISHELL-1 train/dev mixed with DNS noises at SNR -5 to 20 dB [34].	In-house mobile recordings (Kitchen/Mall); <code>t1-t3</code> subsets. CER.

Evaluation Metrics

This chapter is recognition-centric, so the primary evaluation metric is WER for English benchmarks and CER for Mandarin benchmarks. Signal-level or perceptual metrics are reported only when they help interpret recognition behavior, for example in the SNR-stratified and out-of-domain stress-test analyses. Concretely, DNS and CHiME-4 results are primarily reported in WER, Mandarin results in CER, and selected LibriSpeech+DNS, Aurora-4, and VoiceBank-DEMAND tables additionally include STOI or PESQ as diagnostic reference to reveal over-suppression or artifact behavior. This metric choice is consistent with the main purpose of the chapter: the question is not whether the front end looks better under enhancement metrics alone, but whether it leads to more recognition-stable synthesis under frozen ASR back ends.

6.3.2 Parameter Setup and Training Strategy

We summarize the signal-processing configuration, model hyperparameters, and training strategy used in all experiments. Unless otherwise stated, we keep the front-end architecture and optimizer schedule fixed across tracks, and only change the training data according to the dataset-specific protocols in Section 6.3.1.

Signal processing and band split. All signals are sampled at 16 kHz. We compute a complex STFT [108] using an N -sample Hann analysis window, hop size $H = 256$ samples (16 ms), and an N -point FFT with $N = 512$, resulting in $F = N/2 + 1 = 257$ nonnegative-frequency bins per frame. This 50% overlap configuration satisfies the constant overlap-add (COLA) condition for the Hann window and enables perfect reconstruction with the standard iSTFT overlap-add normalization.

We partition the frequency axis into $B = 32$ contiguous non-overlapping sub-bands: the first 31 bands contain 8 bins each and the final band contains 9 bins. Each band token is projected to an embedding dimension of $D = 64$ (Section 6.2.2). For reverberant conditions, we follow the referenced protocols and generate mixtures by convolving the clean speech with a randomly sampled room impulse response (RIR) prior to noise mixing.

Front-end architecture. The PTBM backbone stacks $N_{\text{blk}} = 6$ blocks. In each block, both the temporal and cross-band mixing paths use a bottleneck dimension $D_{\text{b}} = 32$. The temporal path employs a gated dilated depthwise convolution with kernel size $k = 5$ and a fixed dilation schedule $\{d_\ell\}_{\ell=0}^5 = \{1, 2, 4, 8, 16, 1\}$. The cross-band mixer uses multi-head self-attention over the B band tokens with $h = 4$ heads. Each PTBM block includes a channel-mixing feed-forward module with expansion ratio $r = 4$. We apply dropout with probability $p = 0.1$ in the mixing paths and in the feed-forward module.

For the reconstruction heads (Section 6.2.3), we predict a complex mask and a complex residual for each band and assemble them into full-resolution $\widehat{M}, R \in \mathbb{C}^{T \times F}$. We use $\tanh(\cdot)$ to bound the real and imaginary components of the complex mask, while leaving the residual unconstrained.

The LOF predictor $f_{\text{lof}}(\cdot)$ is a two-layer MLP applied to each (t, b) token: $\text{Linear}(D \rightarrow 32) \rightarrow$

GELU $\rightarrow$ Linear($32 \rightarrow 1$), followed by a sigmoid to obtain Λ in Eq. (175).

Frozen back ends and decoding. All ASR back ends remain frozen and are not fine-tuned on any benchmark. For DNS and CHiME-4 English benchmarks, we report WER with frozen Whisper [9] Tiny, Medium, and Large back ends, without external language-model fusion, and apply the same text normalization used in the reference protocol [34]. For the LibriSpeech+DNS SNR breakdown and out-of-domain stress tests following [106], we additionally report the results with frozen Whisper Base and Large-v3 back ends under the same fixed decoding and scoring configuration. For Mandarin, we report CER using a frozen, officially released Paraformer-large back end [34, 109].

For all Whisper back ends, we use deterministic beam-search decoding with a fixed configuration to avoid protocol ambiguity. All decoding settings are fixed and applied consistently across experiments unless otherwise noted. We force the decoding task to `transcribe` and fix the language token to English (`language=en`) for all English benchmarks, that is, we do not use automatic language detection. We set `temperature=0.0` and `beam_size=5`, disable `best-of` and `temperature fallback`, and disable cross-window prompting by setting `condition_on_previous_text=False`. We do not provide any external prompt (`initial_prompt=None`, `carry_initial_prompt=False`) and do not apply an external VAD filter; each utterance is decoded as provided by the benchmark protocol, without additional segmentation heuristics. Unless otherwise noted, we decode without word-level timestamps and apply the same text normalization used in the reference protocols.

Optimization and objectives. We train the front end on random crops of length $T_{\text{seg}} = 4.0$ s; shorter utterances are zero-padded. We optimize Eq. (177) with $\alpha = 0.5$ and $\gamma = 0.5$ throughout. When Aux-CTC is enabled, we set $\beta = 0.1$; otherwise, we set $\beta = 0$. We use an effective batch size of 32 segments per update, implemented with gradient accumulation when needed. We train for 54k update steps per track, following the training-budget convention used in [34]. We use AdamW [110] with $\beta_1 = 0.9$, $\beta_2 = 0.98$, weight decay $w_d = 10^{-2}$, and a peak learning rate of $\eta = 2 \times 10^{-4}$. Training uses 3k-step linear warmup and cosine decay [111] thereafter. We clip the global gradient norm to 5.0 and use automatic mixed precision. All experiments use a fixed random seed of 1234.

For all tracks where Aux-CTC is enabled, the training mixtures are generated from clean corpora with transcripts, for example LibriSpeech [48], WSJ [73], and AISHELL-1 [50] depending on the track, so paired text supervision is available for the auxiliary CTC term. For the auxiliary recognition-consistency term, $\mathcal{R}_{\text{CTC}}(\cdot)$ is a frozen CTC recognizer used only during training. For English, we use a wav2vec2-base CTC model [67] fine-tuned on LibriSpeech-960h, which is the default teacher unless otherwise stated. For Mandarin, we use a wav2vec2 CTC model trained on AISHELL-1. The CTC teacher and the evaluation back ends remain frozen; only the front-end parameters are updated when the auxiliary CTC term is enabled.

We report efficiency in giga multiply–accumulate operations per second (GMAC/s) as a hardware-agnostic measure of front-end complexity [34]. Unless otherwise stated, all reported GMAC/s refer to the enhancement front end only, excluding the ASR back end, matching the conventions in recent front-end studies.

Key hyperparameters are summarized in Table 6.3.

Front-end variants. We use the following fixed variant definitions throughout this chapter:

- A0: PTBM enhancement only, feeding $\hat{s}$ to the ASR back end.
- A1: fixed scalar fusion with $\omega = 0.5$, that is, $\tilde{\Lambda}(t, f) = \omega$ in Eq. (176).
- A2: utterance-level LOF, that is, a single weight $\omega \in [0, 1]$ per utterance, expanded to $\tilde{\Lambda}(t, f) \equiv \omega$ in Eq. (176).
- A3: band–frame LOF, predicting $\Lambda \in [0, 1]^{T \times B}$.
- A4: A3 with Aux-CTC enabled during training ($\beta = 0.1$; wav2vec2-base CTC teacher).

6.3.3 Baselines and Back-End Recognizers

We compare the proposed front end against recent robust-ASR enhancement front ends and fusion-based bridging systems under frozen recognizers. For baselines, we follow the original protocols of the cited references. When official code or models are available, we re-run baselines under our fixed decoding and scoring configuration; otherwise, we report the numbers as published and

Table 6.3: Hyperparameter settings for the proposed system.

Symbol	Description	Value
<i>Signal Processing</i>		
f_s	Sampling rate	16 kHz
N	FFT size	512
H	STFT hop length	256 (16 ms)
F	Frequency bins per frame	257
B	Number of sub-bands	32
<i>Model Architecture (PTBM-LOF)</i>		
D	Band embedding dimension	64
N_{blk}	Number of PTBM blocks	6
D_b	Bottleneck dimension (mixers)	32
k	Depthwise-convolution kernel size	5
$\{d_\ell\}$	Temporal dilation schedule	$\{1, 2, 4, 8, 16, 1\}$
h	Attention heads (cross-band)	4
r / p	FFN expansion ratio / dropout	4 / 0.1
<i>Optimization and Training</i>		
T_{seg}	Training crop length	4.0 s
η	Peak learning rate	2×10^{-4}
w_d	Weight decay	10^{-2}
Steps	Warmup / total update steps	3k / 54k
<i>Loss Functions</i>		
α	Spec. loss weight ($\mathcal{L}_{\text{spec}}$)	0.5
γ	Magnitude-term weight in $\mathcal{L}_{\text{spec}}$	0.5
β	Aux-CTC weight ($\mathcal{L}_{\text{CTC}}$)	0.1

cite the corresponding source. We explicitly indicate dev-set tuned fusion-coefficient settings for prior methods when applicable to avoid ambiguity in comparisons.

Efficiency-oriented BSRNN family. Following the lightweight robust-ASR front-end benchmark of Zhao et al. [34], we include the band-split RNN backbone (BSRNN) and representative efficiency variants, including grouped RNN (GR), pre-downsampling and post-upsampling (PPS), layer-wise resampling (LWR, including SYNC and ASYNC variants reported under the corresponding protocols), and progressive sub-band pruning (SBP-P, reported in selected tables when applicable). For these baselines, we follow the observation-fusion post-processing protocol used in [34], where fusion coefficients, including scalar observation-alignment coefficients, are tuned on the dev set.

Bridging-module baselines on CHiME-4 (Channel-5 protocol). To directly compare with bridging-module systems that predict observation-fusion or observation-alignment coefficients as a separate

module, we include FRCRN+BM and DCCRN+BM reported under the ICASSP 2025 CHiME-4 Channel-5 protocol [105]. Here FRCRN [112] and DCCRN [113] denote the underlying SE front ends used in that protocol. These systems represent two-stage pipelines where an SE model produces an enhanced signal and a bridging module selects fusion coefficients, in contrast to our end-to-end learned fusion.

Pre-trained SE models with bridge-style integration (SNR breakdown and stress tests). For controlled SNR analysis and cross-corpus stress tests, we follow the EUSIPCO 2024 bridging-gap benchmark [106] and include representative pre-trained SE models and their ASR-oriented integration variants as reported therein: CMGAN [114], MP-SENet [115], DEMUCS [116], and SEMamba [98]. We primarily report the bridge-style integration variants, and include selected Enhanced rows only when needed to diagnose over-suppression artifacts at high SNR.

Frozen back-end recognizers and scoring. For English benchmarks, we report WER using frozen Whisper back ends with the same model sizes as the corresponding reference protocols, namely Tiny, Medium, and Large for DNS and CHiME-4 in [34], and Base and Large-v3 for the bridging-gap protocols in [106]. For Mandarin in-house evaluation, we report CER under the frozen recognizer used in [34]. We apply standard text normalization consistent with each protocol. For cross-back-end transfer on CHiME-4 (Section 6.4.4), we additionally report results with a frozen English Paraformer back end (Paraformer-EN) using a fixed decoding configuration.

Whisper decoding configuration. All Whisper decoding hyperparameters are fixed and reported in Section 6.3.2 to avoid protocol ambiguity. We apply the same configuration when re-running baselines whenever possible.

6.4 Results and Analysis

We evaluate the proposed PTBM-LOF front end under frozen recognizers across primary robust-ASR benchmarks, real-recording tests, and out-of-domain stress conditions. Unless otherwise

noted, all comparisons are made in terms of ASR error rates (WER or CER), which are the primary objective of this chapter. For brevity, most benchmark tables report A0, A3, and A4 as the enhancement-only baseline, the learned-fusion variant, and the final system. When needed to illustrate the limitations of fixed or coarse-grained fusion on real recordings, we additionally include A1 and A2, for example in Table 6.5.

6.4.1 DNS Challenge: Efficiency Under Reverberation

Table 6.4: DNS Challenge test results with frozen Whisper back ends. MACs (GMAC/s) are for the front end only.

ID	Method	MACs (GMAC/s)	without Reverb: WER (%)			with Reverb: WER (%)		
			Tiny	Medium	Large	Tiny	Medium	Large
1	Noisy	–	15.02	8.85	4.80	39.51	21.13	10.94
2	BSRNN [34]	1.84	11.36	5.59	4.23	36.60	15.34	10.29
3	+GR [34]	1.09	11.75	5.72	4.40	36.82	15.60	10.87
4a	+PPS [34]	0.55	13.63	5.84	4.95	38.19	20.93	10.91
4c	+LWR-SYNC [34]	1.19	11.93	5.77	4.38	36.72	15.56	10.89
6	+++GR [34]	0.62	11.76	5.72	4.40	36.87	15.61	10.90
A0	Ours (PTBM, no LOF)	0.58	11.83	5.81	4.47	36.95	15.65	10.82
A3	Ours (PTBM + LOF)	0.60	11.13	5.46	4.19	35.78	15.04	10.14
A4	Ours (A3 + Aux-CTC)	0.60	10.97	5.41	4.13	35.52	14.96	10.09

Table 6.4 highlights the intended accuracy–efficiency trade-off. Compared to the recurrent BSRNN front end (ID 2), our final system (A4) reduces the front-end compute from 1.84 to 0.60 GMAC/s, that is, by more than a factor of three, while achieving similar or better WER under both non-reverberant and reverberant conditions. For instance, under the reverberant condition with Whisper Lrg, A4 attains 10.09% WER versus 10.29% for BSRNN.

The ablation trend clarifies the role of each component. A0 is consistently weaker than A3 and A4, indicating that feeding the enhanced estimate alone is insufficient to control ASR-sensitive artifacts. LOF (A3) provides consistent gains over A0 across model sizes and acoustic conditions, showing that LOF eliminates the need for dev-set fusion-coefficient tuning while achieving competitive recognition accuracy. Aux-CTC (A4) yields a smaller but consistent additional gain, which becomes more pronounced on real or mismatched recordings (Sections 6.4.2–6.4.8).

6.4.2 CHiME-4 Real 1-Channel: Generalization to Real Recordings

Table 6.5: CHiME-4 1-channel real-recording results with frozen Whisper back ends (WER, %).

Method	MACs (GMAC/s)	Dev (<code>dt05_real</code>)			Eval (<code>et05_real</code>)		
		Tiny	Medium	Large	Tiny	Medium	Large
Noisy	–	21.76	5.55	4.41	35.03	8.63	6.69
DPARN [34]	1.22	21.13	5.52	4.67	32.97	8.84	7.56
BSRNN [34]	1.84	16.50	5.19	4.25	26.24	8.17	6.36
+++GR [34]	0.62	16.72	5.31	4.35	27.12	8.30	6.40
A0 (PTBM, no LOF)	0.58	17.23	5.41	4.43	27.54	8.47	6.53
A1 (Fixed fusion, $\omega = 0.50$)	0.58	18.04	5.67	4.62	28.13	8.74	6.78
A2 (Utterance-LOF)	0.59	16.79	5.24	4.31	26.83	8.26	6.39
A3 (Band-Frame LOF)	0.60	16.18	5.12	4.17	26.07	8.04	6.27
A4 (A3 + Aux-CTC)	0.60	15.93	5.06	4.13	25.87	7.97	6.16

Real CHiME-4 recordings provide a stringent test for front-end deployability because enhancement artifacts and dataset mismatch are more likely to be amplified by a frozen ASR back end. Table 6.5 shows two consistent patterns.

First, learned fusion is critical on this real-recording track. A1, which uses a fixed scalar fusion coefficient, is consistently worse than A0, indicating that a single fixed fusion coefficient cannot reliably balance suppression and distortion across utterances under real mismatch. Both A2 and A3 improve over A0, and A3 is consistently stronger than A2, supporting the importance of predicting fusion weights at a finer granularity.

Second, Aux-CTC is most beneficial on the real eval sets. On `et05_real`, A4 improves over A3 across all Whisper sizes, and achieves WER comparable to BSRNN while operating at roughly one-third of the front-end compute. This is consistent with the intended role of Aux-CTC: it discourages recognition-unstable distortions during training while keeping the evaluation back end frozen. We further examine whether this benefit transfers to a non-Whisper back end in Section 6.4.4.

6.4.3 Ablations on Band Interaction and Fusion Granularity

To isolate the impact of cross-band interaction in the PTBM backbone, fusion granularity in LOF, and the auxiliary CTC regularizer, we conduct controlled ablations on the CHiME-4 1-channel real track with a fixed frozen Whisper-Medium back end. All variants use the same training data,

optimization schedule, and decoding configuration unless stated otherwise. While Table 6.5 reports A0–A4 across multiple Whisper sizes for overall comparison, here we fix the back end to provide a cleaner attribution study and to include additional variants not covered by A0–A4, for example frame-only and band-only fusion, and alternative cross-band mixers.

Table 6.6: Additional ablations on CHiME-4 real recordings (Whisper-Medium).

Variant	MACs	Dev (%)	Eval (%)
<i>A. PTBM cross-band interaction (keep LOF, no Aux-CTC)</i>			
w/o cross-band mixing	0.45	5.52	8.62
band-MLP mixing	0.50	5.28	8.29
band-Conv1D mixing	0.52	5.30	8.35
PTBM (Proposed)	0.60	5.12	8.04
<i>B. LOF granularity (fix PTBM, no Aux-CTC)</i>			
Enhanced only (A0)	0.58	5.41	8.47
Fixed scalar fusion ($\omega = 0.50$)	0.58	5.67	8.74
Utterance-level LOF	0.59	5.24	8.26
Frame-only LOF	0.59	5.19	8.16
Band-only LOF	0.60	5.21	8.21
Band–Frame LOF	0.60	5.12	8.04
<i>C. Aux-CTC sensitivity (β is weight)</i>			
$\beta = 0.00$ (no Aux-CTC)	0.60	5.12	8.04
$\beta = 0.05$	0.60	5.09	8.01
$\beta = 0.10$ (Proposed A4)	0.60	5.06	7.97
wav2vec2-large CTC teacher	0.60	5.01	7.91

Table 6.6 leads to three conclusions. First, cross-band interaction is necessary: removing cross-band mixing degrades WER substantially (8.62 vs. 8.04), and lightweight alternatives (band-MLP and band-Conv1D) recover part of the gap but remain inferior to band attention, indicating that structured cross-band dependency modeling is beneficial under real mismatch. Second, fusion granularity matters: moving from global or utterance-level fusion to time- or band-dependent fusion consistently improves WER, and the full band–frame map achieves the best result, suggesting that both temporal dynamics and frequency-dependent artifact patterns are relevant for recognition-stable synthesis. Third, Aux-CTC behaves as a stable training-time regularizer: increasing β improves WER monotonically in this range, and a stronger teacher provides a small additional gain, while keeping inference complexity unchanged.

Table 6.7: Cross-back-end evaluation on the CHiME-4 real 1-channel track.

Method	MACs (GMAC/s)	Whisper-Medium		Paraformer-EN	
		dev	eval	dev	eval
Noisy	–	5.55	8.63	8.14	12.37
DPARN [34]	1.22	5.52	8.84	7.89	11.91
BSRNN [34]	1.84	5.19	8.17	7.16	10.83
+++GR [34]	0.62	5.31	8.30	7.42	11.26
A0 (PTBM, no LOF)	0.58	5.41	8.47	7.53	11.49
A3 (PTBM + LOF)	0.60	5.12	8.04	7.09	10.67
A4 (A3 + Aux-CTC)	0.60	5.06	7.97	6.93	10.41

6.4.4 CHiME-4 Real 1-Channel: Cross-Back-End Transfer

Table 6.7 shows that the proposed front end improves recognition accuracy consistently across different frozen back ends. Although Paraformer-EN exhibits higher absolute WER than Whisper-Medium on this track, the relative trends remain consistent: LOF (A3) improves over the enhancement-only baseline (A0), and Aux-CTC (A4) provides an additional gain. In particular, A4 reduces Paraformer-EN WER from 12.37% to 10.41% on `et05_real`, while remaining substantially more efficient than BSRNN (0.60 vs. 1.84 GMAC/s). This cross-back-end consistency suggests that, under our fixed decoding configuration, the benefit of LOF is not tied to a specific recognizer family, but rather to suppressing enhancement artifacts that are broadly detrimental to ASR.

6.4.5 Cross-Lingual Real-World Mandarin: Portability Beyond English

Table 6.8 evaluates cross-lingual portability under real-world Mandarin mobile recordings. We emphasize that this in-house evaluation serves as a portability case study, while our main claims are established on public benchmarks.

Two observations are consistent across subsets. First, the Mall condition is substantially more challenging than Kitchen for all methods, making it a stronger test bed for artifact control under real reverberation and mismatch. Second, LOF is critical for stabilizing recognition under this regime. The enhancement-only variant A0 degrades some subsets relative to BSRNN, whereas introducing LOF (A3) improves all Mall subsets and all Kitchen subsets relative to A0. Aux-CTC (A4) yields further gains and achieves the best CER across all six subsets.

Compared to BSRNN, the final system (A4) attains lower CER on all six subsets with roughly

Table 6.8: In-house Mandarin per-set CER (%) breakdown following the protocol in [34]. Each environment contains three test sets (t1–t3), corresponding to different predefined recording sessions or subsets.

Method	MACs (GMAC/s)	Kitchen CER (%)			Mall CER (%)		
		t1	t2	t3	t1	t2	t3
Noisy	–	6.15	6.62	6.12	45.63	58.20	62.05
BSRNN [34]	1.84	5.84	6.15	5.78	39.11	50.07	55.22
+GR [34]	1.09	5.92	6.24	5.90	40.11	53.10	58.04
+LWR-ASYNC(4) [34]	1.19	5.85	6.16	5.79	39.38	50.26	55.70
+LWR-ASYNC(16) [34]	1.03	5.87	6.18	5.79	39.58	51.15	56.86
++SBP-P [34]	0.99	5.88	6.19	5.83	39.74	52.08	56.90
+++GR [34]	0.62	5.92	6.26	5.92	40.12	53.44	58.06
A0 (PTBM, no LOF)	0.58	5.95	6.31	5.95	39.85	52.34	57.09
A3 (PTBM + LOF)	0.60	5.82	6.05	5.72	38.90	49.85	55.04
A4 (A3 + Aux-CTC)	0.60	5.77	6.02	5.65	38.54	49.42	54.66

one-third of the front-end compute (0.60 vs. 1.84 GMAC/s), indicating that the proposed learned synthesis and auxiliary constraint generalize beyond English-only acoustics.

6.4.6 Direct Comparison to Bridging Modules on CHiME-4 (Channel-5 Protocol)

Table 6.9: CHiME-4 Channel-5 protocol comparison against bridging-module systems (WER, %). Lower is better.

Method	Whisper Base		Whisper Large-v3	
	et_simu	et_real	et_simu	et_real
No front end	12.40	18.24	4.33	4.99
FRCRN + BM [105]	10.95	15.52	4.14	4.36
DCCRN + BM [105]	11.49	16.28	4.18	4.76
A0 (PTBM, no LOF)	11.83	16.54	4.19	4.87
A3 (PTBM + LOF)	10.53	15.17	4.06	4.22
A4 (A3 + Aux-CTC)	10.28	14.93	4.03	4.11

Table 6.9 benchmarks our approach against bridging-module competitors under the Channel-5 reporting convention. Across both `et_simu` and `et_real`, LOF variants (A3 and A4) consistently improve over the enhancement-only baseline (A0), indicating that learned observation fusion is beneficial when evaluated with a frozen ASR back end. Compared with two-stage bridging-module

systems, A4 achieves lower WER under both Whisper Base and Whisper Large-v3, and the advantage is consistent on real recordings, suggesting improved robustness to simulation-to-real mismatch. For example, with Whisper Large-v3 on `et_real`, A4 achieves 4.11% WER compared to 4.36% for FRCRN+BM, while simplifying the pipeline by avoiding a separately trained coefficient-selection module. Finally, Aux-CTC provides a consistent additional gain from A3 to A4 across back ends, supporting its role as a training-time regularizer that discourages recognition-unstable distortions without modifying the evaluation recognizer.

6.4.7 SNR-Stratified Analysis on LibriSpeech + DNS

Table 6.10: LibriSpeech `test-clean` mixed with DNS noises at predefined SNR bins (WER, %). This table emphasizes high-SNR behavior to diagnose over-suppression. Rows for bridge-style integration follow [106]. STOI and PESQ are computed on the front-end output waveform, that is, the fused signal for bridge or LOF variants.

SE model / Method	Whisper Base: WER (%)					Whisper Large-v3: WER (%)					STOI	PESQ
	-12	-6	0	6	12	-12	-6	0	6	12		
Noisy	73.8	46.5	21.5	11.0	7.7	44.3	18.7	6.8	3.8	3.2	0.752	1.272
CMGAN Enh. [106]	80.3	57.4	29.4	14.0	8.5	69.1	42.0	15.5	5.9	3.7	0.780	1.642
CMGAN Bridge [106]	70.3	42.5	19.1	10.4	7.5	43.7	18.0	6.4	3.7	3.1	0.770	1.298
DEMUCS Bridge [106]	68.4	40.5	18.7	10.5	7.6	42.3	16.7	6.2	3.7	3.1	0.777	1.296
SEMamba Bridge [106]	70.4	42.1	18.6	10.0	7.3	43.7	17.9	6.3	3.7	3.1	0.777	1.370
MP-SENet Bridge [106]	72.3	45.6	20.5	10.7	7.5	44.9	18.9	6.9	3.8	3.2	0.762	1.305
A0 (PTBM, no LOF)	71.5	44.1	20.6	11.4	8.2	45.2	20.7	7.6	4.3	3.5	0.764	1.453
A3 (PTBM + LOF)	68.6	41.4	18.4	10.3	7.63	42.1	17.4	6.3	3.6	3.14	0.791	1.384
A4 (A3 + Aux-CTC)	68.2	40.7	18.2	10.1	7.61	41.5	17.1	6.2	3.6	3.12	0.793	1.391

Table 6.10 provides a diagnostic view of the high-SNR regime, where enhancement-only base-lines can suffer from over-suppression. A representative example is CMGAN Enhanced: at 12 dB, its Base WER (8.5%) is worse than Noisy (7.7%), suggesting that perceptually oriented enhancement can introduce distortions that are amplified by the recognizer. In contrast, our LOF variants remain close to the Noisy performance at 12 dB (A3: 7.63%, A4: 7.61%), while improving substantially at low SNR. At -12 dB, A4 reduces Base WER from 73.8% (Noisy) to 68.2% and reduces Large-v3 WER from 44.3% to 41.5%. Compared with the enhancement-only PTBM variant A0, A4 improves WER at every SNR bin for both frozen back ends, indicating that LOF and Aux-CTC

stabilize recognition without relying on an arithmetic average across SNR conditions.

6.4.8 Out-of-Domain Stress Tests: Aurora-4 and VoiceBank-DEMAND

Table 6.11: Performance comparison on out-of-domain stress tests. We report WER (%) for frozen Whisper Base and Large-v3 back ends, along with STOI and PESQ for reference. Bridge-style baselines follow [106]. STOI and PESQ are computed on the front-end output waveform, that is, the fused signal for bridge or LOF variants.

Method	Aurora-4 (Unseen)				VoiceBank-DEMAND (Mismatched)			
	Base	Large-v3	STOI	PESQ	Base	Large-v3	STOI	PESQ
Noisy	22.1	17.3	0.812	1.394	9.0	3.1	0.921	1.973
CMGAN Bridge [106]	22.0	17.2	0.859	1.940	7.8	2.7	0.933	2.183
MP-SENet Bridge [106]	21.4	17.2	0.842	1.618	8.2	2.6	0.934	2.238
DEMUCS Bridge [106]	21.7	17.1	0.845	1.613	8.5	2.9	0.931	2.223
SEMamba Bridge [106]	21.4	17.2	0.845	1.640	7.6	2.7	0.936	2.278
A0 (PTBM, no LOF)	22.4	17.6	0.831	1.547	8.4	2.9	0.924	1.854
A3 (PTBM + LOF)	21.3	16.9	0.852	1.523	7.4	2.6	0.934	2.051
A4 (A3 + Aux-CTC)	20.8	16.7	0.856	1.538	7.2	2.4	0.937	2.083

Aurora-4 and VoiceBank-DEMAND are used as out-of-domain stress tests to assess robustness under unseen noise and channel conditions and mismatched noise distributions, respectively. Table 6.11 summarizes WER with frozen Whisper back ends, together with STOI and PESQ for reference.

On Aurora-4, improvements are expected to be modest due to domain shift. Nevertheless, the proposed method yields consistent gains over both Noisy and bridge-style baselines. In particular, A4 reduces WER from 22.1% to 20.8% with Whisper Base and from 17.3% to 16.7% with Large-v3, indicating improved recognition stability under unseen conditions.

VoiceBank-DEMAND further highlights the misalignment between perceptual metrics and ASR robustness. Several strong bridge-style baselines achieve higher PESQ, for example CMGAN and SEMamba, yet they do not necessarily yield the lowest WER. By contrast, A4 attains the best WER among the reported methods (7.2% vs. 7.6% on Base; 2.4% vs. 2.7% on Large-v3), supporting the motivation of LOF: recognition-oriented synthesis prioritizes artifact suppression that benefits frozen recognizers even when perceptual metrics are not maximized.

6.4.9 Mechanism Analysis of LOF

We analyze the role of learned observation fusion using quantitative evidence from controlled ablations, SNR-stratified evaluation, and mismatch tests.

LOF as a structured fusion constraint. LOF forms a pointwise convex combination on the complex STFT grid using a fusion map that is piecewise-constant along frequency within each sub-band. Consequently, the synthesized spectrum Y at each TF bin lies on the line segment connecting $\hat{S}$ and X . This structurally constrains the front end to interpolate between enhancement and observation, rather than generating unconstrained TF patterns, making the mechanism interpretable as recognition-oriented adaptive interpolation between the enhanced estimate and the original observation.

Fusion granularity is non-trivial. Table 6.6(B) shows that progressively finer fusion granularity improves WER under the same backbone and training setup. In particular, moving from scalar or utterance-level fusion to frame-dependent or band-dependent fusion yields consistent gains, and the full band-frame LOF achieves the best performance. If the learned fusion collapsed to a near-constant coefficient, these variants would behave similarly; the observed ordering therefore supports that band-frame (time $\times$ sub-band) structure in the fusion weights is beneficial for recognition stability.

SNR-stratified behavior is consistent with adaptive interpolation. The SNR breakdown in Table 6.10 provides a diagnostic view of the suppression-distortion trade-off. At high SNR, for example 12 dB, enhancement-only outputs can degrade recognition due to over-suppression or processing distortion, whereas LOF variants remain close to the noisy baseline. At low SNR, for example -12 dB, LOF yields substantial WER reductions, indicating that the model can still leverage enhancement when suppression is critical. Together, these trends are consistent with LOF acting as an adaptive interpolation mechanism that assigns less weight to the enhanced estimate when processing distortion is risky, while relying more on enhancement when noise dominates.

Consistent gains under real and mismatched conditions. On real CHiME-4 recordings (Table 6.5) and out-of-domain stress tests (Table 6.11), LOF consistently improves over enhancement-only outputs. Such settings amplify artifact sensitivity for frozen recognizers, so the consistent gains support the hypothesis that learned fusion mitigates ASR-sensitive distortions without validation-time coefficient tuning.

Overall, the structured fusion form, the granularity-driven ablations, and the SNR-stratified and mismatch evaluations together provide a quantitative justification of LOF.

6.4.10 Summary of Findings

Across synthetic benchmarks, real recordings, and out-of-domain stress tests, the results support three conclusions. First, PTBM provides a competitive front end under stringent front-end compute constraints, enabling strong recognition performance without heavy recurrent designs. Second, LOF is the main contributor to deployability and robustness, consistently improving over enhancement-only and fixed-fusion variants by eliminating dev-set fusion-coefficient tuning through end-to-end learned fusion. Third, the gains are consistent across recognizers and languages, transferring from Whisper to Paraformer and from English to Mandarin, indicating strong cross-model and cross-lingual generalization.

6.5 Conclusion

This chapter studied a practical but under-addressed deployment setting: improving robust ASR when the recognizer is frozen. In this regime, gains in enhancement or perceptual metrics do not reliably translate into lower WER, because recognizers can be disproportionately sensitive to enhancement artifacts and processing distortion.

To address this mismatch under strict efficiency constraints, we proposed a frozen-recognizer front end that combines (i) a parallel-friendly band-split backbone, the Parallel Time–Band Mixer (PTBM), and (ii) ASR-oriented learned observation fusion (LOF). PTBM performs sequence-parallel

temporal mixing and per-frame cross-band interaction on a compact band-token grid, and reconstructs an enhanced spectrum using a mask-plus-residual interface. On top of the enhanced estimate, LOF predicts band-frame fusion weights and performs structured complex-spectrum fusion between the enhanced spectrum and the original observation. This yields an interpretable, convex interpolation mechanism that assigns less weight to the enhanced estimate in artifact-prone regions, while eliminating development-set tuning of scalar fusion coefficients. Optionally, Aux-CTC further regularizes the front end during training to discourage recognition-unstable distortions while keeping the evaluation ASR back end unchanged; it introduces no additional inference-time module or operation.

Across DNS Challenge-derived benchmarks (with and without reverberation), CHiME-4 real recordings, SNR-stratified analysis, out-of-domain stress tests (Aurora-4 and VoiceBank-DEMAND), and cross-lingual Mandarin mobile recordings, the proposed PTBM-LOF front end achieves a favorable accuracy-efficiency trade-off. The improvements are consistent across different frozen recognizers (Whisper and a non-Whisper Paraformer back end), supporting the key conclusion of this chapter: under frozen-ASR pipelines, recognition-stable synthesis and artifact-aware fusion can be more decisive for WER than enhancement metrics alone.

As the final algorithmic chapter, these results complete the thesis progression from topology-robust multichannel modeling and deployable spatial processing (Chapters 3–5) to recognition-stable front-end design for frozen-ASR deployments. Chapter 7 summarizes the thesis-level contributions and outlines future directions.

Chapter 7

Conclusion and Future Work

7.1 Concluding Remarks

Deep learning has become central to modern speech front ends, enabling substantial progress in speech enhancement and robust automatic speech recognition (ASR). However, high benchmark scores alone are not sufficient for deployment. A deployable front end must remain stable under noise, reverberation, array variation, and domain mismatch; expose reliable and inspectable processing interfaces; preserve information needed by downstream tasks; and operate within practical computational budgets, especially when the downstream recognizer is fixed.

The central theme of this thesis is to transform high-performing enhancement models into robust and deployable speech front ends by combining data-driven learning with explicit, constrained, and task-aware processing interfaces. This theme is developed through four tightly connected requirements: (i) topology-robust multichannel spatial processing; (ii) explicit and lightweight spatial filtering with stability-oriented regularization and perceptually motivated objective design; (iii) scalable long-context temporal modeling with coordinated magnitude restoration and complex-spectrum refinement; and (iv) recognition-compatible synthesis and training objectives for frozen-ASR pipelines. The four main algorithmic chapters, Chapters 3–6, address these requirements in turn.

Chapter 3 investigated graph-based spatial topology learning as a way to reduce sensitivity

to microphone ordering, fixed array assumptions, and covariance-estimation brittleness in multi-channel processing. The chapter first introduced complex-domain graph convolution for topology-aware spatial fusion while preserving phase-sensitive cues. It then redesigned the spatial stage as an explicit per-time–frequency convex weighting interface, producing a gain-bounded filter-and-sum style reference that is inspectable and compatible with downstream neural refinement. Finally, stability and generalization were strengthened by keeping inference covariance-free and adding lightweight stabilizers, including learnable temperature control and low-band cross-frequency coupling, together with training-only teacher alignment. Across simulated and real recordings, these results show that topology-aware spatial learning becomes more deployment-friendly when graph reasoning is coupled with an explicit and bounded spatial interface.

Chapter 4 moved from topology learning to a deployable spatial-filtering blueprint and refined it along three complementary dimensions: architectural structure, robustness priors, and objective alignment. The chapter first established Dynamic Spatial Filtering with Residual Spectral Mapping (DSF-RSM), where the spatial operation remains an explicit and efficient filter-and-sum front end, and a lightweight residual mapping module restores spectral details that spatial suppression may attenuate. It then developed PhysMVNet, which preserves an explicit spatial-filtering structure while improving robustness under acoustic mismatch and array perturbation through a physics-inspired STFT-domain regularizer and a multi-band residual restoration strategy. Finally, the chapter addressed optimization mismatch by proposing an analytic, non-learned multi-band relative contrastive loss (MBRCL) that can be attached to different architectures without auxiliary assessment networks. Together, these studies show that deployment-quality enhancement benefits not only from efficient model design, but also from stability-oriented inductive bias and portable objectives that better align training with perceptual and task-oriented evaluation.

Chapter 5 addressed another bottleneck in difficult acoustic scenes: long-context temporal modeling and coordinated magnitude and complex-spectrum enhancement. It proposed an end-to-end dual-path state-space framework with cross-domain interaction. A spatial–spectral encoder provides array-aware time–frequency features, while a two-stage S5 backbone enables efficient long-sequence modeling. By separating coarse magnitude estimation from complex spectral refinement

and exchanging intermediate representations between the two streams, the model aligns complementary cues before reconstruction rather than relying only on final-stage fusion. Comprehensive evaluations on simulated and real multichannel benchmarks show that this structured dual-path design improves enhancement quality and robustness, and that intermediate cross-domain interaction provides clearer gains than loosely coupled branch designs in the reported ablations.

Chapter 6 shifted the emphasis from enhancement-centric modeling to recognition-compatible front-end design under frozen recognizers. This setting highlights a key deployment challenge: an enhanced signal that improves perceptual or reconstruction metrics can still harm recognition if it introduces ASR-sensitive artifacts or removes weak recognizer-relevant cues. To address both robustness and efficiency, the chapter proposed a parallel-friendly band-split backbone, the Parallel Time–Band Mixer (PTBM), which avoids within-block sequential scans. It also introduced ASR-oriented learned observation fusion (LOF), which synthesizes a recognition-stable signal through structured fusion between the enhanced estimate and the original observation on the complex STFT grid. A lightweight training-only auxiliary consistency objective further improves recognition stability without changing inference-time complexity. Across noisy and reverberant benchmarks, real recordings, and cross-lingual evaluations, the results show that recognition-stable synthesis is crucial in frozen-ASR pipelines, and that structured learned fusion can replace brittle development-set coefficient tuning while maintaining a favorable accuracy–efficiency trade-off.

Overall, this thesis demonstrates that robust speech pipelines benefit most when front ends are designed jointly for signal fidelity, stability under mismatch, computational efficiency, and downstream recognition compatibility. The chapter-by-chapter contributions establish a deployable methodology: make spatial processing explicit and bounded, inject stability-oriented inductive bias under acoustic and array mismatch, model long context efficiently with structured sequence modules, and control recognition-sensitive artifacts through task-aware synthesis and frozen-recognizer-aligned objectives.

7.2 Future Work

The results of this thesis suggest several directions for further advancing deployable speech front ends. The following topics are intentionally tied to the four main algorithmic chapters so that the future work follows directly from the limitations and observations identified in this dissertation.

- **Extending spatial front ends to variable arrays and heterogeneous devices.** Building on the topology-robust graph-based spatial processing in Chapter 3, an important next step is to support variable microphone sets at inference. In practical deployments, microphones may be missing, swapped, added, or distributed across heterogeneous devices. Different devices may also introduce channel-dependent frequency responses, gain differences, and calibration mismatch. Future work can target variable- M spatial processing and reliability-aware spatial weighting, so that the model becomes conservative when spatial evidence is ambiguous, especially at low frequencies or under low signal-to-noise ratio (SNR) conditions. Lightweight device-response compensation and channel-mismatch normalization could also be integrated into the spatial front end to improve cross-device generalization without retraining for each hardware configuration.
- **Developing more principled physics-guided stabilization and portable objectives under real mismatch.** Chapter 4 established a deployable filtering blueprint and explored physics-inspired regularization and analytic objective design. A remaining limitation is that several deployment failure modes, such as over-suppression at high SNR, locally irregular time–frequency fluctuations under severe noise, and instability under distribution shift, are not always directly coupled to condition-adaptive training mechanisms. Future work can develop regularizers and objective terms that respond to measurable indicators such as spatial coherence, low-band ambiguity, residual distortion, and residual noise. Another promising direction is to improve objective portability using unlabeled real recordings by combining analytic band-structured objectives, such as MBRCL-style constraints, with teacher–student consistency or augmentation consistency. Such approaches could reduce simulation-to-real gaps while keeping inference-time computation unchanged.

- **Building streaming-capable dual-path long-context models with structured interaction.**

Chapter 5 demonstrated that dual-path state-space modeling with cross-domain interaction improves coordinated magnitude and complex-spectrum refinement, but practical systems often impose strict streaming constraints. Future work can develop streaming-friendly state-space implementations with explicit state caching, bounded look-ahead, and chunk-consistent normalization and interaction policies. Cross-domain interaction can also be made reliability-aware: when phase cues are unreliable, the interaction module should avoid forcing aggressive phase refinement and should instead prioritize stable magnitude restoration. A further direction is to redesign the state-space interface to operate directly on structured tensors, such as time–band–channel representations, rather than relying on full flattening. This can improve memory and computation scaling as microphone count and frequency resolution increase.

- **Advancing multichannel recognition-stable synthesis and stronger frozen-ASR alignment.**

Chapter 6 showed that recognition-stable synthesis through learned observation fusion (LOF) is critical under frozen recognizers, especially on real recordings and under mismatch. A direct extension is to build a multichannel robust-ASR front end that integrates spatial reliability with band–frame fusion. In such a system, observation fusion would become spatially aware: it could back off from enhancement in artifact-prone regions while still exploiting array gains when spatial evidence is reliable. Frozen-ASR alignment can also be strengthened beyond a single auxiliary constraint by using multiple teachers, such as connectionist temporal classification (CTC), sequence-to-sequence, or recurrent neural network transducer (RNN-T) models; confidence-aware consistency; and cross-back-end robustness tests as a standard evaluation axis. Finally, closing the gap to deployment requires system-level validation, including wall-clock latency on target hardware, quantization-aware training for complex-spectrum estimation and fusion predictors, and operator choices that translate low reported giga multiply–accumulate operations per second (GMAC/s) into real throughput gains when paired with large frozen recognizers.

Appendix A

Publications from the Thesis Research

Throughout my doctoral research work, the following peer-reviewed conference and journal papers have been published or submitted.

1. **X. Shen**, R. Wang, W.-P. Zhu, and B. Champagne, “Dual-path state-space modeling with cross-domain interaction for multichannel speech enhancement,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 33, pp. 4239–4252, 2025, doi: 10.1109/TASL-PRO.2025.3618543.
2. **X. Shen** and W.-P. Zhu, “Multichannel speech enhancement using complex-valued graph convolutional networks and triple-path attentive recurrent networks,” in Proc. Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2024, pp. 1–6, doi: 10.1109/APSIPAASC63619.2025.10849253.
3. **X. Shen**, R. Wang, and W.-P. Zhu, “Dynamic spatial filtering with residual spectral mapping for lightweight multichannel speech enhancement,” in Proc. IEEE International Symposium on Circuits and Systems (ISCAS), 2025, pp. 1–5, doi: 10.1109/ISCAS56072.2025.11043352.
4. **X. Shen**, W.-P. Zhu, and B. Champagne, “PhysMVNet: Physics-informed end-to-end MVDR beamformer with residual spectral mapping for multichannel speech enhancement,” in Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), 2025.
5. **X. Shen**, W.-P. Zhu, and B. Champagne, “MVDR-consistent graph-attention beamforming with

local Conformer for multichannel speech enhancement,” accepted by the IEEE Workshop on Sensor Array and Multichannel Signal Processing (SAM), 2026.

6. **X. Shen**, R. Wang, W.-P. Zhu, and B. Champagne, “A non-learned multi-band relative contrastive loss for speech enhancement,” submitted to *IEEE Signal Processing Letters*, under second-round review.
7. **X. Shen**, R. Wang, W.-P. Zhu, and B. Champagne, “Parallel time–band mixing with learned observation-adding for robust ASR front ends,” submitted to Interspeech, 2026.
8. **X. Shen**, R. Wang, W.-P. Zhu, and B. Champagne, “Efficient front-end speech enhancement for robust ASR via parallel time–band mixing and learned observation fusion,” submitted to *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.
9. **X. Shen**, R. Wang, W.-P. Zhu, and B. Champagne, “Covariance-free inference microphone-graph attention spatial front end with learnable temperature for multichannel speech enhancement,” manuscript in preparation to be submitted.

Bibliography

- [1] S. Gannot, E. Vincent, S. Markovich-Golan, and A. Ozerov, “A consolidated perspective on multimicrophone speech enhancement and source separation,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 4, pp. 692–730, 2017.
- [2] J. Benesty, J. Chen, and Y. Huang, *Microphone Array Signal Processing*. Springer, 2008.
- [3] E. Warsitz and R. Häb-Umbach, “Blind acoustic beamforming based on generalized eigenvalue decomposition,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1529–1539, 2007.
- [4] Z.-Q. Wang, P. Wang, and D. Wang, “Complex spectral mapping for single- and multi-channel speech enhancement and robust ASR,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1778–1787, 2020.
- [5] H. Kim, K. Kang, and J. W. Shin, “Factorized MVDR deep beamforming for multi-channel speech enhancement,” *IEEE Signal Processing Letters*, vol. 29, pp. 1898–1902, 2022.
- [6] K. Tesch and T. Gerkmann, “Insights into deep non-linear filters for improved multi-channel speech enhancement,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 563–575, 2023.
- [7] D. Lee and J.-W. Choi, “DeFTAN-AA: Array geometry agnostic multichannel speech enhancement,” in *Proc. Interspeech*, 2024, pp. 3360–3364.

- [8] C. Bøddeker, H. Erdogan, T. Yoshioka, and R. Häb-Umbach, “Exploring practical aspects of neural mask-based beamforming for far-field speech recognition,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 6697–6701.
- [9] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust speech recognition via large-scale weak supervision,” in *Proc. International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 202. PMLR, 2023, pp. 28 492–28 518.
- [10] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. Wiley-Interscience, 2002.
- [11] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, “Speech dereverberation based on variance-normalized delayed linear prediction,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1717–1731, 2010.
- [12] 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.
- [13] J. Heymann, L. Drude, and R. Häb-Umbach, “Neural network based spectral mask estimation for acoustic beamforming,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 196–200.
- [14] H. Erdogan, J. R. Hershey, S. Watanabe, M. I. Mandel, and J. Le Roux, “Improved MVDR beamforming using single-channel mask prediction networks,” in *Proc. Interspeech*, 2016, pp. 1981–1985.
- [15] C. Bøddeker, P. Hanebrink, L. Drude, J. Heymann, and R. Häb-Umbach, “Optimizing neural-network supported acoustic beamforming by algorithmic differentiation,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 171–175.

- [16] K. Qian, Y. Zhang, S. Chang, X. Yang, D. Florencio, and M. Hasegawa-Johnson, “Deep learning based speech beamforming,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5389–5393.
- [17] J. Heymann, L. Drude, C. Bøddeker, P. Hanebrink, and R. Häb-Umbach, “BeamNet: End-to-end training of a beamformer-supported multi-channel ASR system,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5325–5329.
- [18] Y. Luo, C. Han, N. Mesgarani, E. Ceolini, and S.-C. Liu, “FaSNet: Low-latency adaptive beamforming for multi-microphone audio processing,” in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 260–267.
- [19] Z. Zhang, Y. Xu, M. Yu, S.-X. Zhang, L. Chen, and D. Yu, “ADL-MVDR: All deep learning MVDR beamformer for target speech separation,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6089–6093.
- [20] A. Li, W. Liu, C. Zheng, and X. Li, “Embedding and beamforming: All-neural causal beamformer for multichannel speech enhancement,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6487–6491.
- [21] P. Tzirakis, A. Kumar, and J. Donley, “Multi-channel speech enhancement using graph neural networks,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 3415–3419.
- [22] H. Guo, Y. Chen, X.-L. Zhang, and X. Li, “Graph attention based multi-channel U-Net for speech dereverberation with ad-hoc microphone arrays,” in *Proc. Interspeech*, 2024, pp. 617–621.
- [23] S. Brody, U. Alon, and E. Yahav, “How attentive are graph attention networks?” in *International Conference on Learning Representations (ICLR)*, 2022.
- [24] D. Lee and J.-W. Choi, “DeFT-AN: Dense frequency-time attentive network for multichannel speech enhancement,” *IEEE Signal Processing Letters*, vol. 30, pp. 155–159, 2023.

- [25] D. Lee and J.-W. Choi, “DeFTAN-II: Efficient multichannel speech enhancement with sub-group processing,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 4850–4866, 2024.
- [26] Y. Luo, Z. Chen, and T. Yoshioka, “Dual-path RNN: Efficient long sequence modeling for time-domain single-channel speech separation,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 46–50.
- [27] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, “Conformer: Convolution-augmented transformer for speech recognition,” in *Proc. Interspeech*, 2020, pp. 5036–5040.
- [28] A. Gu, K. Goel, A. Gupta, and C. Ré, “On the parameterization and initialization of diagonal state space models,” in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 35 971–35 983.
- [29] J. T. H. Smith, A. Warrington, and S. W. Linderman, “Simplified state space layers for sequence modeling,” in *International Conference on Learning Representations (ICLR)*, 2023.
- [30] K. Iwamoto, T. Ochiai, M. Delcroix, R. Ikeshita, H. Sato, S. Araki, and S. Katagiri, “How bad are artifacts?: Analyzing the impact of speech enhancement errors on ASR,” in *Proc. Interspeech*, 2022, pp. 5418–5422.
- [31] Y. Yang, A. Pandey, and D. Wang, “Time-domain speech enhancement for robust automatic speech recognition,” in *Proc. Interspeech*, 2023, pp. 4913–4917.
- [32] Y. Dissen, S. Yonash, I. Cohen, and J. Keshet, “Enhanced ASR robustness to packet loss with a front-end adaptation network,” in *Proc. Interspeech*, 2024, pp. 5008–5012.
- [33] J. Yu, H. Chen, Y. Luo, R. Gu, and C. Weng, “High fidelity speech enhancement with band-split RNN,” in *Proc. Interspeech*, 2023, pp. 2483–2487.
- [34] S. Zhao, W. Wang, and Y. Qian, “Lightweight front-end enhancement for robust ASR via frame resampling and sub-band pruning,” in *Proc. Interspeech*, 2025, pp. 3409–3413.

- [35] R. Scheibler, E. Bezzam, and I. Dokmanić, “Pyroomacoustics: A Python package for audio room simulation and array processing algorithms,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 351–355.
- [36] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, “Neural message passing for quantum chemistry,” in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, vol. 70. PMLR, 2017, pp. 1263–1272.
- [37] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *International Conference on Learning Representations (ICLR)*, 2017.
- [38] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph attention networks,” in *International Conference on Learning Representations (ICLR)*, 2018.
- [39] A. Gu, K. Goel, and C. Ré, “Efficiently modeling long sequences with structured state spaces,” in *International Conference on Learning Representations (ICLR)*, 2022.
- [40] C. K. A. Reddy, V. Gopal, R. Cutler, E. Beyrami, R. Cheng, H. Dubey, S. Matuskevych, R. Aichner, A. Aazami, S. Braun, P. Rana, S. Srinivasan, and J. Gehrke, “The INTER-SPEECH 2020 deep noise suppression challenge: Datasets, subjective testing framework, and challenge results,” in *Proc. Interspeech*, 2020, pp. 2492–2496.
- [41] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, “The third CHiME speech separation and recognition challenge: Dataset, task and baselines,” in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2015, pp. 504–511.
- [42] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, “An analysis of environment, microphone and data simulation mismatches in robust speech recognition,” *Computer Speech & Language*, vol. 46, pp. 535–557, 2017.
- [43] W. Rao, Y. Fu, Y. Hu, X. Xu, Y. Jv, J. Han, Z. Jiang, L. Xie, Y. Wang, S. Watanabe, Z.-H. Tan, H. Bu, T. Yu, and S. Shang, “ConferencingSpeech challenge: Towards far-field multi-channel speech enhancement for video conferencing,” in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2021, pp. 679–686.

- [44] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, “Speech enhancement for a noise-robust text-to-speech synthesis system using deep recurrent neural networks,” in *Proc. Interspeech*, 2016, pp. 352–356.
- [45] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, “DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM,” National Institute of Standards and Technology (NIST), Tech. Rep. NISTIR 4930, 1993.
- [46] A. Varga and H. J. M. Steeneken, “Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems,” *Speech Communication*, vol. 12, no. 3, pp. 247–251, 1993.
- [47] E. Guizzo, C. Marinoni, M. Pennese, X. Ren, X. Zheng, C. Zhang, B. Masiero, A. Uncini, and D. Comminiello, “L3DAS22 challenge: Learning 3D audio sources in a real office environment,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 9186–9190.
- [48] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “LibriSpeech: An ASR corpus based on public domain audio books,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.
- [49] D. Pearce and H.-G. Hirsch, “The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions,” in *Proc. 6th International Conference on Spoken Language Processing (ICSLP)*, vol. 4, 2000, pp. 29–32.
- [50] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, “AISHELL-1: An open-source mandarin speech corpus and a speech recognition baseline,” in *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*, 2017, pp. 1–5.
- [51] Y. Shi, H. Bu, X. Xu, S. Zhang, and M. Li, “AISHELL-3: A multi-speaker mandarin TTS corpus,” in *Proc. Interspeech*, 2021, pp. 2756–2760.

- [52] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, “SDR—half-baked or well done?” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 626–630.
- [53] International Telecommunication Union, “ITU-T Recommendation P.862: Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs,” 2001, withdrawn on 2024-01-05; see the P.863 series for current standards.
- [54] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, “Perceptual evaluation of speech quality (PESQ)—a new method for speech quality assessment of telephone networks and codecs,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, 2001, pp. 749–752.
- [55] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, “An algorithm for intelligibility prediction of time-frequency weighted noisy speech,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.
- [56] J. Jensen and C. H. Taal, “An algorithm for predicting the intelligibility of speech masked by modulated noise maskers,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 2009–2022, 2016.
- [57] C. K. A. Reddy, V. Gopal, and R. Cutler, “DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6493–6497.
- [58] International Telecommunication Union, “ITU-T Recommendation P.808: Subjective evaluation of speech quality with a crowdsourcing approach,” 2021, recommendation P.808 (06/2021).
- [59] C. K. A. Reddy, V. Gopal, and R. Cutler, “DNSMOS P.835: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 886–890.

- [60] International Telecommunication Union, “ITU-T Recommendation P.835: Subjective test methodology for evaluating speech communication systems that include noise suppression algorithm,” 2003, approved 2003-11-13.
- [61] V. Sovrasov, “ptflops: A FLOPs counting tool for neural networks in PyTorch framework,” 2018–2024, software repository. [Online]. Available: <https://github.com/sovrasov/flops-counter.pytorch>
- [62] J. Benesty, J. Chen, and Y. Huang, “A generalized MVDR spectrum,” *IEEE Signal Processing Letters*, vol. 12, no. 12, pp. 827–830, 2005.
- [63] Y. Luo, Z. Chen, N. Mesgarani, and T. Yoshioka, “End-to-end microphone permutation and number invariant multi-channel speech separation,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6394–6398.
- [64] A. Pandey, B. Xu, A. Kumar, J. Donley, P. Calamia, and D. Wang, “Multichannel speech enhancement without beamforming,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6502–6506.
- [65] A. Pandey, B. Xu, A. Kumar, J. Donley, P. Calamia, and D. Wang, “TPARN: Triple-path attentive recurrent network for time-domain multichannel speech enhancement,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6497–6501.
- [66] S. Jia, S. Jiang, S. Zhang, M. Xu, and X. Jia, “Graph-in-graph convolutional network for hyperspectral image classification,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 1, pp. 1157–1171, 2024.
- [67] A. Baeovski, Y. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 12 449–12 460.
- [68] K. Tan and D. Wang, “A convolutional recurrent neural network for real-time speech enhancement,” in *Proc. Interspeech*, 2018, pp. 3229–3233.

- [69] M. Hao, J. Yu, and L. Zhang, “Spatial-temporal graph convolution network for multichannel speech enhancement,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6512–6516.
- [70] S. Lv, Y. Fu, Y. Jv, L. Xie, W. Zhu, W. Rao, and Y. Wang, “Spatial-DCCRN: DCCRN equipped with frame-level angle feature and hybrid filtering for multi-channel speech enhancement,” in *Proc. IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 436–443.
- [71] C. Marinoni, R. F. Gramaccioni, C. Chen, A. Uncini, and D. Comminiello, “Overview of the L3DAS23 challenge on audio-visual extended reality,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–2.
- [72] D. S. Williamson, Y. Wang, and D. Wang, “Complex ratio masking for monaural speech separation,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 3, pp. 483–492, 2016.
- [73] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, “WSJCAM0: A british english speech corpus for large vocabulary continuous speech recognition,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, 1995, pp. 81–84.
- [74] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. A. P. Habets, R. Häb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, R. Maas, S. Gannot, and B. Raj, “The REVERB challenge: A common evaluation framework for dereverberation and recognition of reverberant speech,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2013, pp. 1–4.
- [75] Z. Chen, T. Yoshioka, L. Lu, T. Zhou, Z. Meng, Y. Luo, J. Wu, X. Xiao, and J. Li, “Continuous speech separation: Dataset and analysis,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7284–7288.

- [76] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, “A short-time objective intelligibility measure for time-frequency weighted noisy speech,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2010, pp. 4214–4217.
- [77] S.-W. Fu, C.-F. Liao, and Y. Tsao, “Learning with learned loss function: Speech enhancement with Quality-Net to improve perceptual evaluation of speech quality,” *IEEE Signal Processing Letters*, vol. 27, pp. 26–30, 2020.
- [78] J. M. Martín-Doñas, A. M. Gómez, J. A. González, and A. M. Peinado, “A deep learning loss function based on the perceptual evaluation of the speech quality,” *IEEE Signal Processing Letters*, vol. 25, no. 11, pp. 1680–1684, 2018.
- [79] E. Vincent, R. Gribonval, and C. Févotte, “Performance measurement in blind audio source separation,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.
- [80] K. Shimada, Y. Bando, M. Mimura, K. Itoyama, K. Yoshii, and T. Kawahara, “Unsupervised speech enhancement based on multichannel NMF-informed beamforming for noise-robust automatic speech recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 5, pp. 960–971, 2019.
- [81] B. Tolooshams, R. Giri, A. H. Song, U. Isik, and A. Krishnaswamy, “Channel-attention dense U-Net for multichannel speech enhancement,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 836–840.
- [82] X. Li and R. Horaud, “Narrow-band deep filtering for multichannel speech enhancement,” *arXiv preprint arXiv:1911.10791*, 2019.
- [83] Y. Yang, C. Quan, and X. Li, “MCNet: Fuse multiple cues for multichannel speech enhancement,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

- [84] C. Quan and X. Li, “SpatialNet: Extensively learning spatial information for multichannel joint speech separation, denoising and dereverberation,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 1310–1323, 2024.
- [85] Z.-Q. Wang, S. Cornell, S. Choi, Y. Lee, B.-Y. Kim, and S. Watanabe, “FNeural speech enhancement with very low algorithmic latency and complexity via integrated full- and sub-band modeling,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.
- [86] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7132–7141.
- [87] S. S. Stevens, J. Volkman, and E. B. Newman, “A scale for the measurement of the psychological magnitude pitch,” *The Journal of the Acoustical Society of America*, vol. 8, no. 3, pp. 185–190, 1937.
- [88] R. Yamamoto, E. Song, and J.-M. Kim, “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6199–6203.
- [89] X. Rong, T. Sun, X. Zhang, Y. Hu, C. Zhu, and J. Lu, “GTCRN: A speech enhancement model requiring ultralow computational resources,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 971–975.
- [90] Z. Lai, C. Yan, and Y. Fu, “Hybrid spectral denoising transformer with guided attention,” in *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 13 065–13 075.
- [91] W. Liu, A. Li, C. Zheng, and X. Li, “A separation and interaction framework for causal multi-channel speech enhancement,” *Digital Signal Processing*, vol. 126, p. 103519, 2022.
- [92] Z.-Q. Wang, J. Le Roux, and J. R. Hershey, “Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation,” in *Proc. IEEE*

- International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 1–5.
- [93] N. Bisagno, B. Zhang, and N. Conci, “Group LSTM: Group trajectory prediction in crowded scenarios,” in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 213–225.
- [94] J. Zhang, J. Zou, Z. Su, P. Lian, and S. Fan, “DTCNA-Net: Dual temporal convolutional neural network with noise estimation and attention mechanism for intelligent fault diagnosis under the complex noise environment,” in *Proceedings of the 6th International Conference on Pattern Recognition and Intelligent Systems (PRIS)*, 2024, pp. 41–48.
- [95] L. Sun, S. Yuan, A. Gong, L. Ye, and E. S. Chng, “Dual-branch modeling based on state-space model for speech enhancement,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 1457–1467, 2024.
- [96] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “WaveNet: A generative model for raw audio,” *arXiv preprint arXiv:1609.03499*, 2016.
- [97] A. Pandey, B. Xu, A. Kumar, J. Donley, P. Calamia, and D. Wang, “Time-domain ad-hoc array speech enhancement using a triple-path network,” in *Proc. Interspeech*, 2022, pp. 729–733.
- [98] R. Chao, W.-H. Cheng, M. La Quatra, S. M. Siniscalchi, C.-H. H. Yang, S.-W. Fu, and Y. Tsao, “An investigation of incorporating Mamba for speech enhancement,” in *Proc. IEEE Spoken Language Technology Workshop (SLT)*, 2024, pp. 302–308.
- [99] W. Zhang, K. Saijo, Z.-Q. Wang, S. Watanabe, and Y. Qian, “Toward universal speech enhancement for diverse input conditions,” in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2023, pp. 1–6.

- [100] S. Araki, A. Yamamoto, T. Ochiai, K. Arai, A. Ogawa, T. Nakatani, and T. Irino, “Impact of residual noise and artifacts in speech enhancement errors on intelligibility of human and machine,” in *Proc. Interspeech*, 2023, pp. 2503–2507.
- [101] T. Ochiai, K. Iwamoto, M. Delcroix, R. Ikeshita, H. Sato, S. Araki, and S. Katagiri, “Re-thinking processing distortions: Disentangling the impact of speech enhancement errors on speech recognition performance,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 3589–3602, 2024.
- [102] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks,” in *Proc. International Conference on Machine Learning (ICML)*, 2006, pp. 369–376.
- [103] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” *arXiv preprint arXiv:1607.06450*, 2016.
- [104] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.
- [105] Z. Cui, C. Cui, T. Wang, M. He, H. Shi, M. Ge, C. Gong, L. Wang, and J. Dang, “Reducing the gap between pretrained speech enhancement and recognition models using a real speech-trained bridging module,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025, pp. 1–5.
- [106] K.-C. Wang, Y.-J. Li, W.-L. Chen, Y.-W. Chen, Y.-C. Wang, P.-C. Yeh, C. Zhang, and Y. Tsao, “Bridging the gap: Integrating pre-trained speech enhancement and recognition models for robust speech recognition,” in *Proc. European Signal Processing Conference (EUSIPCO)*, 2024, pp. 426–430.
- [107] J. Thiemann, N. Ito, and E. Vincent, “The diverse environments multi-channel acoustic noise database (DEMAND): A database of multichannel environmental noise recordings,” *Proceedings of Meetings on Acoustics*, vol. 19, no. 1, p. 035081, 2013.

- [108] D. Griffin and J. S. Lim, “Signal estimation from modified short-time Fourier transform,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.
- [109] Z. Gao, S. Zhang, I. McLoughlin, and Z. Yan, “Paraformer: Fast and accurate parallel transformer for non-autoregressive end-to-end speech recognition,” in *Proc. Interspeech*, 2022, pp. 2063–2067.
- [110] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *International Conference on Learning Representations (ICLR)*, 2019.
- [111] I. Loshchilov and F. Hutter, “SGDR: Stochastic gradient descent with warm restarts,” in *International Conference on Learning Representations (ICLR)*, 2017.
- [112] S. Zhao, B. Ma, K. N. Watcharasupat, and W.-S. Gan, “FRCRN: Boosting feature representation using frequency recurrence for monaural speech enhancement,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 9281–9285.
- [113] Y. Hu, Y. Liu, S. Lv, M. Xing, S. Zhang, Y. Fu, J. Wu, B. Zhang, and L. Xie, “DCCRN: Deep complex convolution recurrent network for phase-aware speech enhancement,” in *Proc. Interspeech*, 2020, pp. 2472–2476.
- [114] R. Cao, S. Abdulatif, and B. Yang, “CMGAN: Conformer-based metric GAN for speech enhancement,” in *Proc. Interspeech*, 2022, pp. 936–940.
- [115] Y.-X. Lu, Y. Ai, and Z.-H. Ling, “MP-SENet: A speech enhancement model with parallel denoising of magnitude and phase spectra,” in *Proc. Interspeech*, 2023, pp. 3834–3838.
- [116] A. Défossez, G. Synnaeve, and Y. Adi, “Real time speech enhancement in the waveform domain,” in *Proc. Interspeech*, 2020, pp. 3291–3295.