Department of Computer Science and Engineering

Advanced Machine Learning (AY2025/2026)

Continuous Assessment: Individual Research Paper

Progress Evaluation Report

Project Code: **SP008**

Targeted Enhancements for Single-Channel Speech Enhancement

Student Name:	DEIYAGALA T. D. H.
Student ID:	210112H
Course Conductor:	Dr. Uthayasanker Thayasivam
Work Type:	Individual
Academic Year:	2025/2026
Submission Date:	August 2025

Executive Summary

Single-channel speech means working with audio from just one microphone. The goal is to make speech clearer from that single source, without relying on extra microphones or special hardware. This project focuses on improving the clarity and naturalness of everyday spoken audio in noisy places, while keeping the approach practical and efficient.

The plan is to build on a strong existing model and add a few simple, realistic improvements that fit the course timeline. The changes will focus on how the model learns, how the training data are prepared, and a couple of light touches to make the final audio sound smoother. These are small, low-risk steps designed to help the model handle different types of background noise and mild echo better, without making it slower or heavier.

Progress will be checked using common listening-quality and intelligibility measures, along with basic efficiency checks to confirm that the method remains fast enough for practical use. The main risks are not matching the starting model's results right away and seeing small ups and downs in the scores. To handle this, the work will first recreate the starting point carefully, then add each change one by one and keep what clearly helps.

So far, the reading and setup are in place, the data are ready, and the base system is being connected. Next, the plan is to confirm the starting results and then introduce the planned improvements step by step, reporting early findings in time for the mid-project review.

1. Project Overview & Objectives

Problem. Single-channel speech enhancement under real-world noise and mild reverberation, with the goal of improving perceived quality and intelligibility while maintaining low latency.

Baseline and Benchmark. The reproduced *state-of-the-art baseline* is **DPCN**, and the primary dataset/benchmark is **DEMAND**[7], using the standard train/validation/test splits for fair comparison.

Primary Objective. Achieve clear, measurable gains over the reproduced DPCN baseline on DEMAND *without* materially increasing runtime or model size.

Success Criteria. Relative to the reproduced DPCN on DEMAND:

- Consistent improvements on listening-quality and intelligibility metrics (e.g., PESQ[8]/STOI directionally higher; SI-SDR higher).

- **Efficiency**: Parameter count kept within a small budget increase and real-time factor (RTF) close to or below 1.0 (CPU, single thread).

Literature Review

This review highlights six works relevant to single-channel speech enhancement, beginning with DPCRN as the primary baseline. Each work is organized under the subtopics: *Problem Addressed*, *Innovations*, *Architecture*, *Dataset and Results*, and *Relation to This Project*.

1) DPCRN: Dual-Path Convolution Recurrent Network (Primary Baseline)

[1]
Problem Addressed: DPCRN targets single-channel speech enhancement, improving perceptual quality and intelligibility while maintaining low latency and model size. It handles real-world noise and mild reverberation where long-range temporal context is crucial, delivering strong PESQ/STOI/SI-SDR on standard benchmarks without resorting to heavy architectures.
Innovations: Dual-path temporal modeling segments sequences to efficiently learn short- and long-range dependencies, paired with a convolutional encoder–decoder and recurrent bottleneck for temporal coherence. Stable mask estimation and artifact reduction advance the quality–efficiency trade-off over earlier CRN/RNN baselines.
Architecture: CNN layers encode and decode STFT features with skip connections; the dual-path recurrent module alternates intra-segment and inter-segment modeling. TF-domain masks are estimated and applied to noisy spectra, followed by ISTFT reconstruction, supporting near real-time CPU inference.

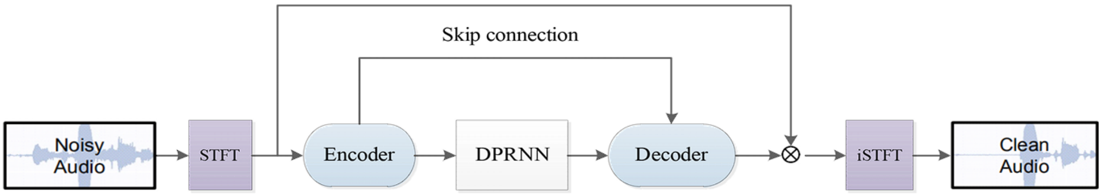

Figure 1: Proposed DPCRN model

Dataset and Results: Evaluated on VoiceBank+DEMAND, reporting PESQ, STOI/ESTOI, SI-SDR, parameters, and RTF. DPCRN consistently achieves competitive quality with moderate compute, showing clear gains over classical baselines while remaining CPU-friendly.
Relation to This Project: DPCRN is adopted as the SOTA baseline (B0) on DEMAND for fair ablations. Planned enhancements—multi-term losses, gentle SNR curriculum, light RIRs, EMA, and a tiny post-filter—complement DPCRN without increasing runtime, aiming for statistically sound PESQ/STOI/SI-SDR gains.

2) RNNoise: Lightweight RNN Denoiser (Efficiency Reference)[2]

Problem Addressed: RNNoise focuses on real-time noise suppression under tight CPU and memory constraints, prioritizing ultra-low latency over peak enhancement scores. It targets conferencing and streaming scenarios where consistent performance on general-purpose hardware outweighs leaderboard metrics.

Innovations: Compact GRU networks with DSP-guided features achieve practical suppression at minimal cost, establishing evaluation norms that include runtime, memory, and stability alongside perceptual metrics.

Architecture: A small recurrent network operates on frame-wise features to estimate suppression gains or masks, avoiding heavy convolutions or attention, enabling predictable latency and resource usage.

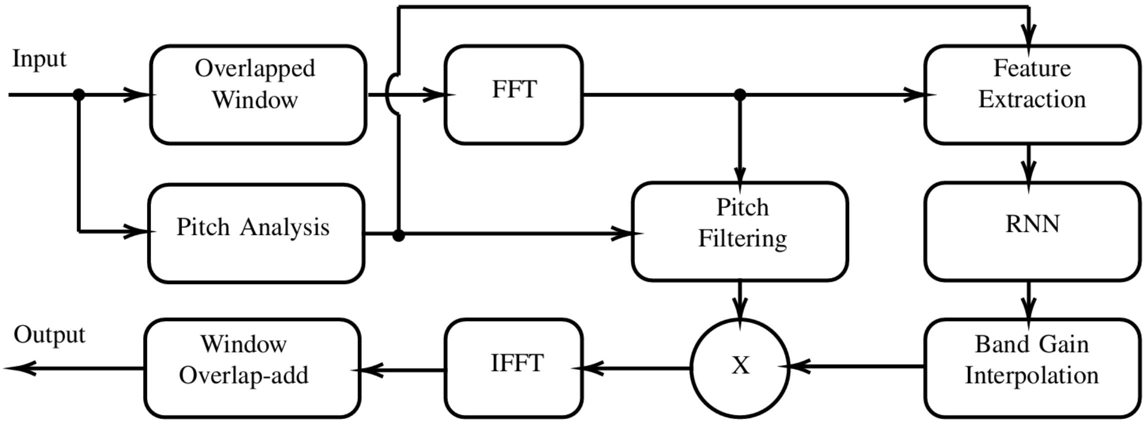

Figure 2: RNNoise system architecture overview[2]

Dataset and Results: While not SOTA on VoiceBank+DEMAND, RNNoise delivers smooth, real-time suppression on diverse streams with extremely low resource usage. It lags on PESQ/STOI/SI-SDR but excels in RTF and memory efficiency.

Relation to This Project: RNNoise serves as an efficiency yardstick, ensuring DPCRN improvements remain practical. Comparisons help guide trade-offs between quality and runtime or memory.

3) CRN Variants: Convolutional Recurrent Networks [3]

Problem Addressed: CRNs combine local spectral detail and temporal coherence to improve intelligibility and perceptual quality while avoiding large attention overhead. They balance gains with parameter count, training stability, and acceptable latency.

Innovations: U-Net-like encoder–decoders with recurrent bottlenecks, mask-based estimates, phase-aware options, and refined losses reduce musical noise and artifacts. Modular designs allow incremental improvements.

Architecture: CNN encoders extract TF features; LSTM/GRU bottlenecks model temporal dependencies; decoders reconstruct magnitude or complex spectra. Skip connections preserve structure. Optional additions (SE gates, depthwise convolutions) improve efficiency without major redesigns.

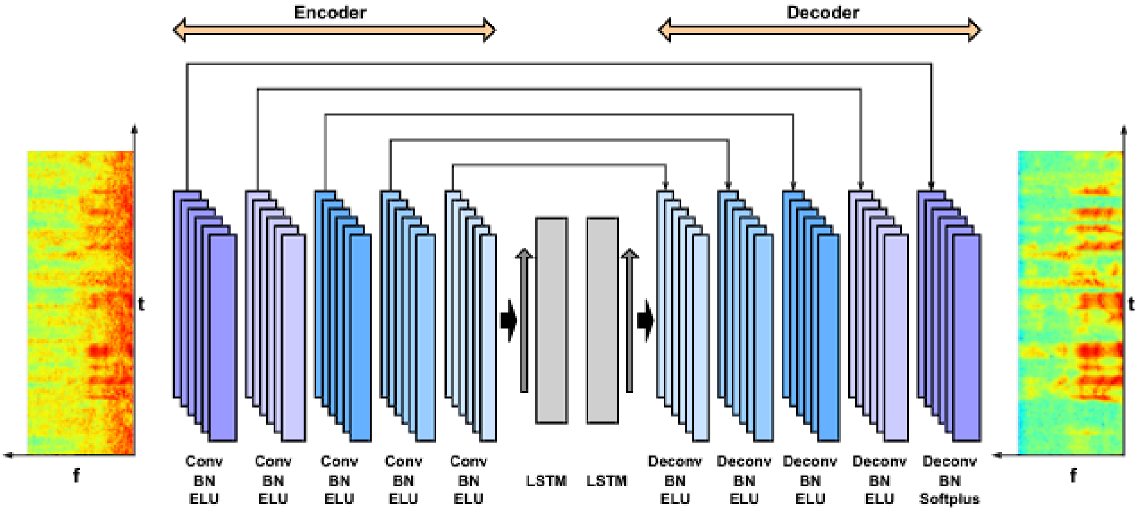

Figure 3: Network architecture of the proposed CRN[3]

Dataset and Results: Evaluated on VoiceBank+DEMAND, CRNs deliver solid PESQ/STOI/SI-SDR gains with a strong quality–efficiency balance.
Relation to This Project: CRN literature supports CNN+RNN baselines like DPCRN and motivates low-risk enhancements—multi-term losses, gentle curricula, minimal gating—preserving compactness while allowing ablation studies.

4) Attention and Transformer-Based Enhancement [4]

Problem Addressed: Attention models capture long-range dependencies and nonstationary noise patterns, aiming to improve quality in difficult SNRs or acoustic scenes without excessive compute or latency.
Innovations: Self-attention in TF space, block-sparse/sub-band attention, and CNN–Transformer hybrids preserve local structure while modeling global context efficiently.
Architecture: CNN front-ends extract features; attention layers replace or augment RNNs; streaming/sub-band attention retains real-time viability. These systems are heavier than CRNs, often requiring compression or pruning for low-latency CPU inference.

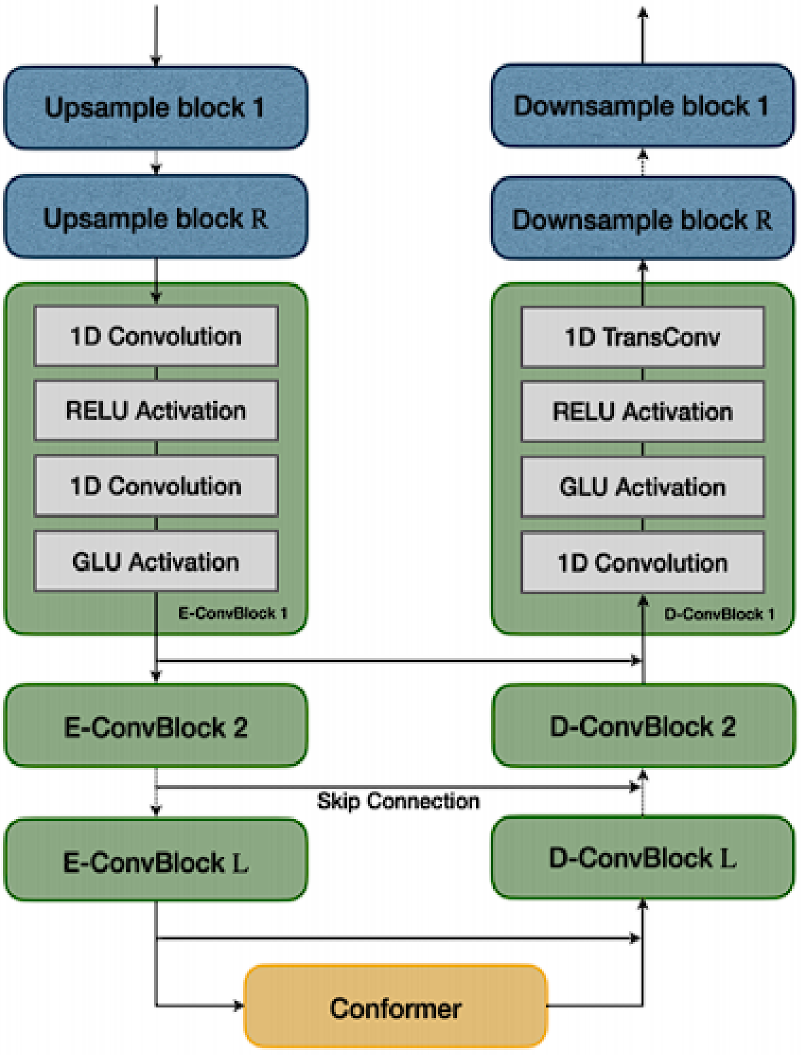

Figure 4: Overview of proposed Architecture[4]

Dataset and Results: VoiceBank+DEMAND and DNS datasets show strong metrics but higher compute and memory footprints. Real-time factor can be impacted unless optimized.
Relation to This Project: Highlights potential quality gains but risk of violating efficiency constraints. This project favors light modifications (small gating, depthwise convolutions) over full attention stacks.

5) Phase-Aware Learning: Complex Spectral Mapping and Losses[5]

Problem Addressed: Magnitude-only enhancement leaves phase errors. Phase-aware learning models complex spectra or masks for perceptual improvements without destabilizing training.
Innovations: Predict complex ratio masks or real/imaginary spectra with mixed objectives (time-domain SI-SDR + multi-resolution STFT losses). Stabilizes learning, reduces

artifacts, improves detail, and keeps overhead low.
Architecture: Backbones mirror CRN/DPCRN; outputs and loss functions are complex-domain. Skip connections and compact temporal modeling maintain efficiency.
Dataset and Results: On VoiceBank+DEMAND, phase-aware losses improve PESQ/STOI/SI-SDR with small overhead.
Relation to This Project: Implements a simple multi-term objective with time-domain fidelity and MR-STFT components, optionally phase-aware, achieving steady DEMAND gains with minimal runtime impact.

6) Curriculum, Augmentation, and Training Stability[6]

Problem Addressed: Generalization across noises, SNRs, and mild reverberation is challenging; training can be unstable. Low-overhead, robust practices improve convergence and reliability without heavy architectures.
Innovations: SNR curricula start with easier mixtures, progressing to harder; light RIR convolution adds realism; expanded noise banks diversify training. EMA, careful optimizers, schedulers, and gradient clipping enhance stability.
Architecture: Methods are architecture-agnostic, integrating with CRN/DPCRN or attention-based backbones. Modular design allows controlled ablations to quantify contributions.
Dataset and Results: VoiceBank+DEMAND-style setups show small but reliable PESQ/STOI/SI-SDR gains and stable metrics across runs, with minimal impact on inference speed.
Relation to This Project: Plans include gentle SNR curriculum, light RIR augmentation, EMA, and minimal post-filtering. Each element is validated via ablations on DPCRN baseline to achieve measurable, reproducible gains while respecting real-time and model-size constraints.

2. Project Plan

The project plan is structured to first establish a reliable baseline by reproducing existing state-of-the-art results. This ensures a clear reference point against which all subsequent enhancements can be evaluated. Emphasis will be placed on reproducibility, modular design, and rigorous experimentation, enabling systematic tracking of improvements while maintaining consistency in metrics, data handling, and overall implementation practices.

2.1 Baseline Reproduction Plan

Codebase & Tooling. The implementation will be based on **PyTorch**, organized into a modular structure (`/data`, `/configs`, `/src`, `/scripts`, `/eval`). Deterministic seeds will be enforced to ensure repeatability, and mixed-precision training will be employed where beneficial for efficiency in both memory and speed.

Dataset.

- The **DEMAND (Diverse Environments Multi-channel Acoustic Noise Database)** dataset will serve as the primary benchmark. DEMAND is widely used in speech enhancement research as it provides recordings of real-world noise captured in diverse acoustic environments (e.g., domestic, office, public, and transportation settings). Each recording contains spatially and temporally varying background noise collected with high-quality microphones, making it more realistic than artificially generated noise.
- For this project, DEMAND will be used in a single-channel setting by downmixing to mono. Standard training–validation–test splits will be adopted, with strict enforcement of non-overlapping noise scenes across splits to prevent data leakage. All preprocessing steps (resampling, loudness normalization, and SNR-based mixing) will follow established community protocols, ensuring fair comparability with prior work.

Preprocessing. Audio will be converted to mono with consistent time–frequency transforms (STFT parameters fixed across all experiments). Loudness normalization and SNR-based mixing (covering easy-to-hard noise levels) will be applied. Validation and test noise sources will be disjoint from training to maintain unbiased evaluation.

Evaluation Metrics. Performance will be measured with both perceptual and objective criteria:

- **Perceptual:** PESQ, STOI
- **Signal-based:** SI-SDR (or SDR)
- **Efficiency:** Real-time factor (RTF), parameter count, and peak memory usage

Reproduction Target. The goal is to reproduce the baseline **DPCN** performance on DEMAND within a narrow tolerance. This configuration will be frozen and designated as *Baseline (B0)*, forming the foundation for subsequent ablation studies.

2.2 Milestones & Deliverables

- **B0 reproduction:** Baseline reproduced, verified, and locked.
- **Ablation studies:** Iterative experiments with structured reports per modification.
- **Mid-term report:** Consolidated summary of results (tables, plots, initial analysis).
- **Final deliverables:** Comprehensive evaluation, polished code repository, reproducibility documentation, and consolidated results.

2.3 Reporting Protocol

Results will be reported with emphasis on three dimensions:

- **Quality and intelligibility:** PESQ, STOI, SI-SDR.
- **Efficiency:** RTF, parameter count, multiply–accumulate operations (MACs), peak memory.
- **Reliability:** Statistical robustness via 95% bootstrap confidence intervals and paired significance tests against B0.

2.4 Reproducibility & Repository Structure

- **/configs:** YAMLs defining B0 and ablation settings (including seeds & hyperparameters).
- **/scripts:** Key utilities (`prepare_data.py`, `train.py`, `eval.py`, `profile.py`).
- **/src:** Modular implementation for models, loss functions, datasets, and augmentations.
- **Experiment Tracking:** Results logged in CSV/JSONL formats; tensorboard curves; commit hash stored with outputs for traceability.

2.5 Risks & Mitigations

- **Risk:** Baseline mismatch or high metric variance. **Mitigation:** Fixed seeds/configs and multiple short repeat runs for validation.
- **Risk:** Limited compute or time. **Mitigation:** Prioritize low-cost ablations, enable early stopping, and adopt staged evaluation.
- **Risk:** Data leakage from overlapping conditions. **Mitigation:** Strict enforcement of disjoint splits; automated verification scripts.

2.6 Resources & Dependencies

- **Compute:** GPU/CPU allocation (with defined budget of hours), adequate storage, and checkpoint frequency policy.
- **Dependencies:** Public access to DEMAND dataset; pinned Python and library versions for reproducibility; containerized environment (Docker/Conda) for cross-system consistency.

3. Methodology Outline

The methodology is designed to systematically improve single-channel speech enhancement while maintaining efficiency. It begins by defining core objectives and loss formulations, then moves to structured data preparation and training strategies. Lightweight inference techniques are integrated to reduce artifacts, and a carefully planned ablation study allows incremental evaluation, ensuring each modification contributes measurable benefits without compromising model stability or runtime performance.

3.1 Loss Design (Core)

- Use a simple multi-term objective that balances time-domain fidelity with spectral consistency, avoiding heavy or exotic components.
- Keep weighting stable and easy to tune (fixed or gently scheduled), favoring predictable training.

3.2 Data & Training

- Gentle SNR curriculum (start easier, then include harder mixes) and light reverberation to improve robustness on DEMAND-like conditions.
- Stable training setup with well-tested optimizer/scheduler choices, exponential moving average of weights, and gradient clipping.

3.3 Inference Smoothing (Lightweight)

- Apply a small energy floor and mild band-limited smoothing on masks/outputs to reduce musical noise without modifying DPCN's core.

3.4 Ablation Plan

- Start from *B0 (DPCN on DEMAND)*, add the loss change, then the data/training tweaks, then light smoothing; measure after each step and keep only changes that show consistent gains with minimal cost.

3.4.1 Experimental Design & Ablations

Baselines.

- **B0: Reproduced DPCN (frozen config)** (*frozen config* = settings are fixed for fair comparisons).
- **B1: RNNoise off-the-shelf (efficiency reference)** (*off-the-shelf* = used as-is without modifications; *efficiency reference* = a speed/latency and small-model comparison point).

Planned ablations (added cumulatively) (*ablations* = add one change at a time to see its effect; *cumulatively* = each step keeps prior changes).

1. **A1: B0 + SI-SNR + MR-STFT (fixed equal weights)** (*SI-SNR* = a time-domain quality measure; *MR-STFT* = multi-resolution spectral view; *fixed equal weights* = simple, constant balance).
2. **A2: A1 + uncertainty-based dynamic loss weighting** (*dynamic weighting* = training auto-adjusts importance of each loss term).
3. **A3: A2 + TF-SE gate at bottleneck** (*TF-SE gate* = light attention/gating over time–frequency features; *bottleneck* = central layer where features are most compressed).
4. **A4: A3 + depthwise separable convs** (*depthwise separable* = lighter convolutions that can reduce compute while keeping quality).
5. **A5: A4 + EMA/SWA and curriculum SNR** (*EMA/SWA* = averaging weights for stabler models; *curriculum SNR* = start with easier noise levels, then harder).

4. Timeline & Milestones (Weeks 5–12)

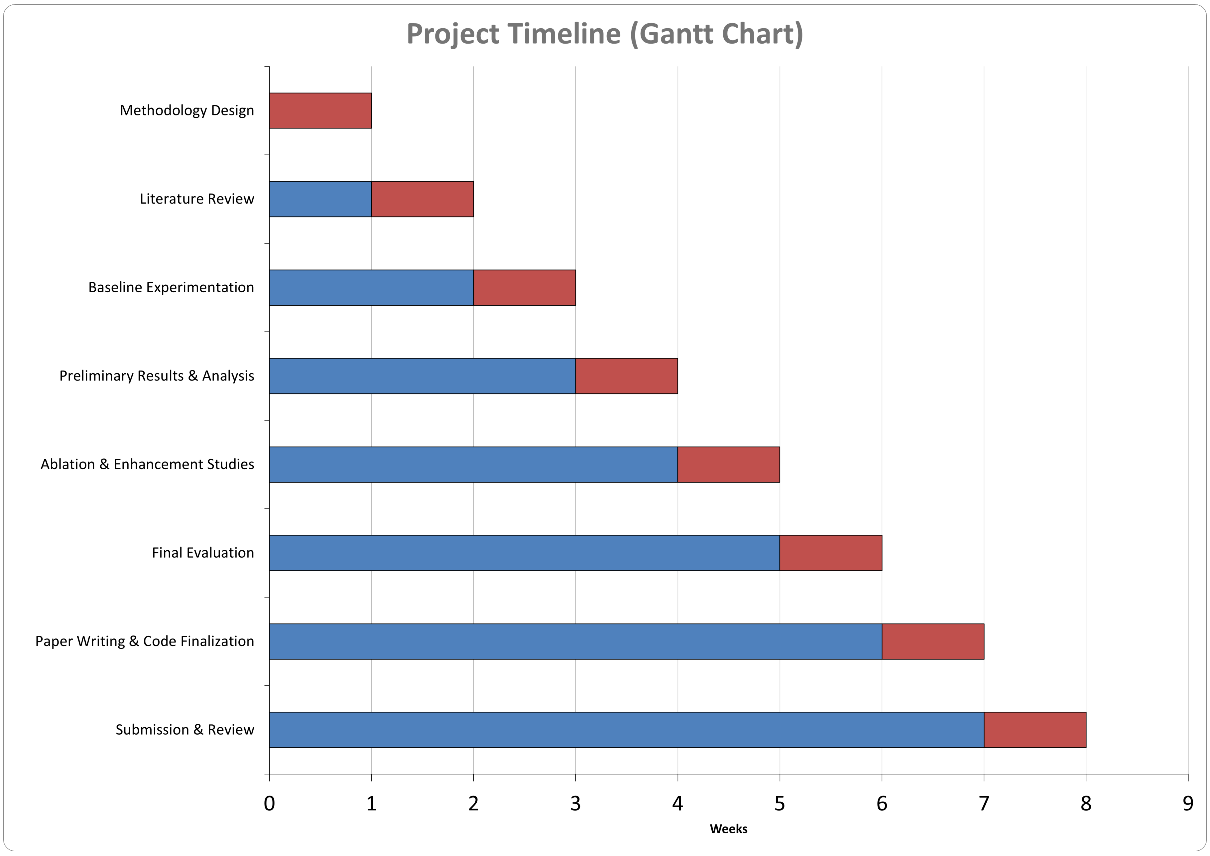

Figure 5: Gantt Chart

5. Risk Register & Mitigations

Baseline mismatch. *Mitigation:* Maintain strict configuration parity; ablate STFT/masking details; report performance deltas relative to *our* B0 baseline only.

Small or noisy gains. *Mitigation:* Use confidence intervals and paired statistical tests; retain only effects that are significant and stable across different noise types and SNR slices.

Compute/latency creep. *Mitigation:* Employ depthwise convolutions; enforce a parameter cap; profile every pull request; reserve test-time augmentation (TTA) for *offline tables only.*

Data leakage. *Mitigation:* Enforce noise/utterance disjointness across splits; implement automated split checksum validation.

6. Current Status (as of Progress Evaluation)

- Literature review notes completed (DPCN variants, complex masking, MR-STFT).
- **To be continued:**
  - Data pipeline
  - B0 training
  - A1 loss implementation

7. Expected Contributions

1. A simple, reproducible multi-objective loss (SI-SNR + MR-STFT + complex CRM) with uncertainty weighting for DPCRN.
2. A tiny TF-SE gate that yields quality gains.
3. A rigorously reported ablation suite with efficiency metrics (RTF, params, MACs) and statistical testing.
4. Clean, open repo enabling course replication and conference submission.

References

[1] X. Le, H. Chen, K. Chen, and J. Lu, "DPCRN: Dual-Path Convolution Recurrent Network for Single Channel Speech Enhancement," Interspeech, pp. 2811–2815, 2021. doi:10.21437/Interspeech.2021-296.

[2] J.-M. Valin, “RNNoise: Learning Noise Suppression,” project documentation and technical report, 2018. (Lightweight RNN-based real-time noise suppression; official project materials.)

[3] K. Tan and D. L. Wang, “A Convolutional Recurrent Neural Network for Real-Time Speech Enhancement,” Interspeech, pp. 3229–3233, 2018.

[4] S. Kim, S. Lee, K. Kim, S. Yoon, and S. Beack, “Time-Domain Speech Enhancement Using Conformer,” Interspeech, pp. 2127–2131, 2021.

[5] D. S. Williamson, Y. Wang, and D. L. 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. doi:10.1109/TASLP.2015.2512042.

[6] B. Gao, M. Sun, and C. Xiong, “SNR-Based Progressive Learning of Deep Neural Networks for Speech Enhancement,” Interspeech, pp. 3713–3717, 2016.

[7] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, “Speech Enhancement for a Noise-Robust Text-to-Speech Synthesis System Using DNNs,” Interspeech, pp. 352–356, 2016. (Common anchor for VoiceBank+DEMAND.)

[8] 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,” Proc. IEEE ICASSP, pp. 749–752, 2001.