A Comparative Study of Noise Effects, VAD, and ASR Methods in Speech Recognition

Anastasiia Sapeha, Ibrahim Kovan, Subashkumar Rajanayagam, Kirill Karpov, Maksim Gering, Dmitry Kachan and Eduard Siemens

Anhalt University of Applied Sciences, Bernburger Str. 55, Koethen, Germany

{*anastasiia.sapeha, ibrahim.kovan, subashkumar.rajanayagam, kirill.karpov, maksim.gering, dmitry.kachan, eduard.siemens*}*@hs-anhalt.de*

Keywords: Speech Recognition, Noise-Robust Speech Processing, Voice Activity Detection, Automatic Speech Recognition, Word Error Rate, Speech Processing Pipeline, Real-Time Factor (RTF), Controlled Noise, GTA-5 Dataset, AMI Dataset, Factor Analysis.

Abstract: Automatic speech recognition (ASR) deployed in real environments is strongly affected by background noise and by upstream processing decisions such as noise suppression and voice activity detection (VAD). This paper provides a controlled, pipeline-level comparison of common front-end and segmentation choices and quantifies their joint impact on both transcription accuracy and computational efficiency. We evaluate a fixed three-stage pipeline consisting of noise suppression, VAD-based segmentation, and speech-to-text (ASR) transcription. Noise suppression is instantiated as either no suppression or DeepFilterNet; VAD is instantiated as none, Silero VAD, or WebRTC VAD; and ASR is instantiated as wav2vec 2.0 or faster-whisper (large-v3). In-vehicle noise is added to clean speech using a representative car noise profile and speech-active SNR calibration, covering multiple SNR levels and a clean reference condition. Experiments are conducted on datasets representing scripted utterances and meeting-style conversational speech. We report Word Error Rate (WER) and Real-Time Factor (RTF) for all 12 pipeline configurations across SNR conditions, and analyze main effects and interactions between pipeline components. Results highlight that ASR choice dominates the accuracy–efficiency trade-off, while noise suppression and VAD introduce dataset- and recognizer-dependent effects, including non-additive interactions that can change the sign and magnitude of preprocessing gains under different noise regimes.

1 INTRODUCTION

Automatic speech recognition (ASR) systems are increasingly deployed in real-world environments where background noise, reverberation, and non-speech acoustic events are unavoidable. Under such conditions, transcription accuracy can degrade substantially, making robustness to acoustic variability a practical requirement. Although noise robustness has been addressed through front-end enhancement, robust acoustic modeling, and noise-tolerant training strategies, overall performance in deployment is often determined by interactions between upstream processing decisions within an integrated speech processing pipeline.

In operational settings, ASR is frequently preceded by noise suppression and voice activity detection (VAD). VAD-based segmentation can reduce irrelevant non-speech input and computational load, but segmentation errors may truncate low-energy speech or distort boundaries, increasing deletions and substitutions and ultimately worsening Word Error Rate (WER). The joint impact of noise conditions, enhancement, segmentation, and recognition is therefore difficult to infer from component-level evaluation and is best assessed at the pipeline level.

In addition to recognition accuracy, computational efficiency is often critical in real-time and resource-limited deployments. Real-Time Factor (RTF) is adopted as an efficiency metric, defined as the ratio between processing time and audio duration, enabling direct comparison of different pipeline configurations in terms of real-time feasibility. Model initialization and startup latency are excluded from RTF to focus on steady-state inference performance.

A fixed three-stage pipeline is evaluated, in which noise suppression is followed by voice activity detection and speech-to-text transcription. The noise

suppression stage is instantiated as either no suppression or DeepFilterNet; the VAD stage as no VAD, Silero, or WebRTC; and the ASR stage as wav2vec2 or faster-whisper. Clean and noise-degraded conditions are evaluated under controlled in-vehicle noise, including a clean baseline processed by the same pipeline. Experiments are conducted on three speech datasets representing scripted speech, meeting conversations, and spontaneous dialogue. Recognition accuracy (WER) and computational efficiency (RTF) are reported for all method combinations, enabling comparison of accuracy–efficiency trade-offs under realistic noise conditions. Analyses are performed separately per dataset and language.

2 RELATED WORK

Robust automatic speech recognition (ASR) under noisy and mismatched acoustic conditions has been an active area of research for decades. Early studies addressed robustness primarily through signal processing techniques, noise estimation, and statistical acoustic modeling, often relying on accurate voice activity detection (VAD) to separate speech from background noise [1]. These works highlighted that errors in speech segmentation can significantly affect downstream recognition performance, particularly in non-stationary noise environments where reliable speech–non-speech discrimination is challenging.

Recent advances in deep learning have led to substantial improvements in ASR performance. Large-scale end-to-end and self-supervised models enable strong recognition accuracy across diverse datasets and acoustic conditions [2]. However, such models are typically evaluated on pre-segmented or cleanly prepared audio, while their behavior within realistic processing pipelines that include noisy inputs and imperfect segmentation remains less explored. Recent benchmarking efforts report considerable variability in Word Error Rate (WER) across datasets, noise conditions, and speaker characteristics, highlighting the need for controlled and reproducible evaluation protocols [3].

Noise suppression and speech enhancement are commonly employed as front-end components to improve intelligibility under adverse acoustic conditions. However, prior work has shown that gains in perceptual speech quality or signal-to-noise ratio do not necessarily translate into lower ASR error rates, and may even degrade recognition performance due to enhancement-induced speech distortion and mismatch with acoustic models [4]. These findings suggest that the impact of noise suppression should be evaluated in conjunction with downstream components, rather than in isolation.

Voice Activity Detection remains a critical intermediate stage in speech processing pipelines. Traditional VAD approaches based on energy thresholds or statistical divergence are computationally efficient but often degrade under low-SNR or non-stationary noise [1]. More recent neural VAD models demonstrate improved robustness by learning noise-invariant representations, including architectures designed for real-time or resource-constrained scenarios [5]. In addition, integrating noise suppression prior to VAD has been shown to improve frame-level VAD performance in degraded environments, indicating that segmentation quality itself is sensitive to front-end processing [6].

Despite improvements in VAD accuracy, multiple studies report that frame-level VAD metrics such as accuracy or AUROC correlate weakly with downstream ASR performance, particularly when segmentation boundary errors occur [7]. Analyses of streaming and API-based ASR systems further demonstrate that VAD failures may lead to the complete loss of user utterances, which can be more detrimental than transcription errors alone in interactive settings [8]. These observations highlight the importance of evaluating segmentation and recognition jointly within an integrated pipeline.

Beyond recognition accuracy, real-time and streaming ASR systems must also satisfy constraints on computational efficiency and processing latency. Several studies explicitly compare recognition quality and inference speed, revealing trade-offs between WER and end-to-end processing time [9]. These findings motivate treating runtime and latency as first-class evaluation metrics along with transcription accuracy.

While prior work often focuses on individual components such as noise suppression, VAD, or ASR models under fixed conditions, fewer studies systematically evaluate complete speech processing pipelines using multiple alternative methods at each stage. This work addresses this gap by analyzing a full speech processing pipeline under controlled noise conditions, comparing different combinations of noise characteristics, VAD algorithms, and speech-to-text systems, and evaluating their impact on both recognition accuracy and processing efficiency.

3 EXPERIMENTAL PIPELINE AND SETUP

Figure 1 provides an overview of the speech processing pipeline evaluated in this study. Clean speech recordings are evaluated both in a clean baseline condition and after transformation into noisy signals under controlled acoustic conditions. The resulting audio is then processed by a fixed sequence of components consisting of noise suppression (NS), voice activity detection (VAD), and automatic speech recognition (ASR), followed by quantitative evaluation. All combinations of pipeline components are evaluated.

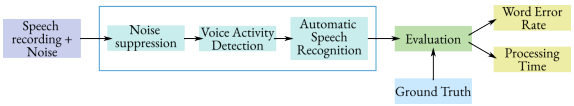

Figure 1: Overview of the evaluated speech processing pipeline.

Evaluated methods:

- Noise Suppression: `none`, `deepfilternet`
- VAD: `none`, `silero`, `webrtc`
- ASR: `wav2vec2`, `faster-whisper`

We selected the evaluated components to represent common, reproducible design choices in practical ASR pipelines while keeping the full-factorial grid compact. For noise suppression, we compare `none` against DeepFilterNet as a widely used, off-the-shelf neural speech enhancement method that can be integrated as a drop-in front end without task-specific training or tuning. This pairing provides a clear control condition and allows us to quantify when enhancement helps or harms downstream recognition under non-stationary in-vehicle noise, including potential distortion-induced mismatches.

For segmentation, we evaluate `none`, WebRTC VAD, and Silero VAD to cover three typical deployment regimes: no explicit segmentation, a lightweight real-time classical VAD (WebRTC) commonly used in streaming/RTC systems, and a popular neural VAD (Silero) often adopted for improved robustness under low SNR and conversational dynamics. For ASR, `faster-whisper` (large-v3) is used as the baseline recognizer due to its strong robustness in diverse acoustic conditions and its practical optimized inference implementation; `wav2vec2` is included as a representative self-supervised ASR alternative with different modeling assumptions and runtime characteristics. Together, these choices span realistic engineering trade-offs in accuracy, segmentation reliability, and computational efficiency while enabling systematic analysis of main effects and non-additive interactions across the $2 \times 3 \times 2$ pipeline grid.

All combinations are evaluated: 2 (denoise) $\times$ 3 (vad) $\times$ 2 (asr) $=$ 12 pipelines. Each pipeline is evaluated at each configured SNR value.

3.1 Datasets

The experimental evaluation was conducted on two speech corpora selected to cover scripted and conversational conditions and different recording setups: a GTA-V derived dataset [10] and the AMI Meeting Corpus [11]. To keep the computational budget comparable, approximately three hours of audio were used from each corpus; the exact file manifests and all data-preparation scripts were released in the accompanying repository.[1] Fixed subsets were curated manually and are fully specified via released manifests.

For the GTA-V corpus, we selected 3,886 subtitle-aligned scripted utterances, corresponding to approximately three hours of audio. The subset includes all non–cut-scene recordings, while files containing Chinese-language speech were excluded in order to maintain language consistency across the experiments. From the AMI Meeting Corpus, we used the headset-mix recordings, which amount to roughly 100 hours of spontaneous multi-speaker meeting audio. This dataset provides naturally occurring conversational speech recorded in realistic meeting scenarios.

Dataset-provided transcripts were normalized and were used as reference text for per-file WER computation; results were reported separately for each dataset (and language) to avoid cross-domain aggregation effects.

3.2 Noise Modeling

To simulate in-vehicle acoustic interference under controlled and reproducible conditions, background noise is added to the speech signals using a single representative in-car noise profile. Noise recordings were collected during car trips, resulting in more than 130 recordings. Candidate recordings were characterized using power spectral density (PSD) representations, and k-means clustering was applied to group acoustically similar recordings. The representative profile was selected from the largest cluster as the recording closest to the cluster centroid in feature space.

The selected profile is illustrated in Fig. 2 using its waveform and log-Mel spectrogram.

For each utterance, the representative noise recording is used as the noise signal $v[n]$. Length

[1] `github.com/AnastasiiaSapeha/ICAIIT_Audio_pipeline`

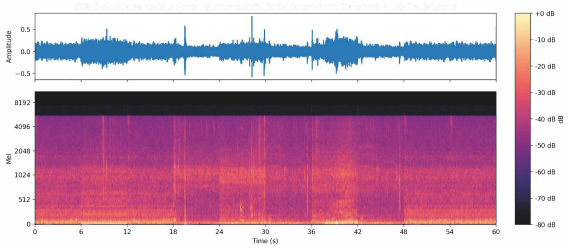

Figure 2: Waveform and log-Mel spectrogram of the selected in-vehicle noise profile.

matching to the clean speech $x[n]$ is achieved deterministically by truncation if the noise recording is longer than the utterance and by cyclic repetition followed by truncation if it is shorter. Additive mixing [12] is performed according to Eq. (1) to obtain the noisy mixture $y[n]$.

$$y[n] = x[n] + \alpha v[n]. \tag{1}$$

The target $\mathrm{SNR}_{\mathrm{dB}}$ is enforced over speech-active samples only to avoid bias from long silent regions. A binary speech-activity mask $m[n] \in \{0,1\}$ is defined for SNR calibration, where $m[n] = 1$ denotes speech-active samples. The mask $m[n]$ is obtained using an active speech level estimate (ITU-T P.56) and is used only for SNR calibration. The mean-square powers P_x and P_v are defined for clean speech and length-matched noise, respectively, and are computed only over speech-active samples by averaging $x[n]^2$ and $v[n]^2$ over indices with $m[n] = 1$. If speech-activity masking is disabled for SNR calibration, $m[n] \equiv 1$ is used and powers are computed over the full utterance.

The noise scaling factor α is computed as in Eq. (2), ensuring that the desired SNR is achieved on speech-active samples; active-speech-level-based scaling prior to additive mixing has been used in related speech enhancement evaluations [13].

$$\alpha = \sqrt{\frac{P_x}{P_v}}\, 10^{-\mathrm{SNR}_{\mathrm{dB}}/20}. \tag{2}$$

To evaluate robustness under varying noise conditions, SNR values (in dB) are selected on a quasi-logarithmic scale: $\{0, 2, 4, 7, 10, 14, 20, 70\}$.

3.3 Noise Suppression

Two noise suppression settings are evaluated: `deepfilternet` and `none`. In the none condition, the input signal is forwarded unchanged. DeepFilterNet is applied using deepfilternet 0.5.6. No task-specific tuning is performed.

By comparing pipelines with and without noise suppression at each SNR level, the study assesses whether front-end enhancement improves downstream segmentation and recognition accuracy, and how it affects computational efficiency.

3.4 Voice Activity Detection

Voice activity detection is applied to segment speech from non-speech regions prior to recognition in order to reduce the amount of irrelevant audio forwarded to the speech-to-text systems and to limit the impact of background noise and silence on downstream recognition performance.

Three VAD settings are evaluated: `none`, `silero`, and `webrtc`. In the none condition, no explicit segmentation is performed and the full input signal is forwarded to the ASR system. Silero VAD is a neural frame-level detector widely used in practical speech processing pipelines. Silero VAD is applied with a threshold of 0.35 and minimum speech/silence durations of 250/100 ms at 16 kHz. The WebRTC voice activity detector is included as a widely used signal-processing-based VAD originally developed for streaming and interactive speech communication systems and optimized for low-latency operation. WebRTC VAD is applied with aggressiveness 2 and 30 ms frames at 16 kHz.

The VAD stage produces time-aligned speech segments that are forwarded to the speech-to-text systems. VAD performance is not evaluated using standalone segmentation metrics; instead, its effect is assessed indirectly through its impact on downstream recognition accuracy and computational efficiency at the pipeline level.

3.5 Automatic Speech Recognition Systems

Speech recognition is performed using `wav2vec2` and `faster-whisper`. Faster-Whisper is applied using faster-whisper 1.1.0 with ctranslate2 4.6.0 and the model faster-whisper-large-v3.

wav2vec2 inference is performed using the Hugging Face stack. For English datasets, wav2vec2-base-960h is used. No additional language-model decoding or task-specific adaptation is applied.

All ASR systems operate on the segments produced by the VAD stage (or on the unsegmented signal when VAD is disabled) and generate textual hypotheses evaluated using WER and RTF.

3.6 Evaluation Metrics

Pipeline performance is evaluated using Word Error Rate (WER) and Real-Time Factor (RTF). WER mea-

sures recognition accuracy by comparing ASR outputs with ground-truth transcripts, while RTF characterizes computational efficiency relative to real-time processing.

Before computing recognition accuracy, reference (ground truth) and hypothesis (ASR output) texts are normalized to reduce the influence of formatting and orthographic variation. The normalization procedure includes conversion to lowercase, removal of punctuation symbols, and whitespace normalization.

After normalization, recognition accuracy is computed using Word Error Rate according to Equation 3, which measures the normalized number of word-level errors between a reference transcription and the corresponding ASR hypothesis.

$$\mathrm{WER} = \frac{S + D + I}{N}, \tag{3}$$

where S denotes substitutions, D deletions, I insertions, and N the number of words in the reference transcription. WER is computed per audio file after normalization using the `jiwer` 3.1.0 library, which performs word-level alignment based on Levenshtein distance.

Computational efficiency is evaluated using the Real-Time Factor for the end-to-end pipeline defined in Equation 4.

$$\mathrm{RTF} = \frac{T_{\text{processing}}}{T_{\text{audio}}}, \tag{4}$$

where $T_{\text{processing}}$ denotes the effective processing time and T_{audio} the duration of the corresponding input audio. RTF values below 1 indicate faster-than-real-time processing, whereas values above 1 indicate slower-than-real-time operation.

RTF is computed based on steady-state inference performance. Model initialization and startup latency are excluded from the reported values, as the evaluation focuses on continuous processing rather than one-time model loading overhead. The audio duration used for RTF computation corresponds to the original input signal before VAD-based segmentation, ensuring consistent comparison across different pipeline configurations.

3.7 Hardware and Execution Environment

All experiments were conducted on a single hardware platform to ensure consistent and reproducible performance measurements across all pipeline configurations. The experimental setup was based on an Intel NUC8i5BEH2 mini-PC equipped with an Intel Core i5-8259U processor, 32 GB of DDR4 memory, and a 512 GB NVMe solid-state drive.

Compute-intensive stages of the pipeline, including neural noise suppression and selected speech-to-text systems, were accelerated using an external NVIDIA GeForce RTX 3090 GPU with 24 GB of memory. All pipeline components were executed locally, and GPU acceleration was applied during inference.

4 RESULTS

ASR accuracy is evaluated under controlled addi-tive noise by sweeping SNR, and the averaged WER is reported for all combinations of NS, VAD, and ASR backend on AMI and GTA.

Fig. 3 shows that WER decreases monotonically with increasing SNR for all configurations. Across the full SNR range, faster-whisper consistently out-performs wav2vec2, with the largest gap at low SNR (0–7dB) and a persistent separation at higher SNR (14–20 dB), indicating that the ASR backend is the dominant factor for WER on AMI. The effect of NS is backend-dependent: for faster-whisper it is largely neutral or slightly detrimental, while wav2vec2 ben-efits more clearly from NS, particularly at low SNR. VAD has a comparatively minor influence on WER, as curves for different VAD variants largely overlap within each ASR/NS group.

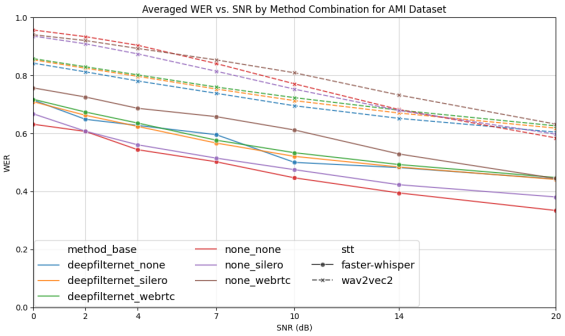

Figure 3: Average WER versus SNR on AMI for all NS/VAD/ASR configurations.

Fig.4 exhibits the same overall trend: WER improves with SNR, with most gains between 0 and 10dB and saturation toward higher SNR. Faster-whisper achieves the lowest WER for all SNR values and shows limited sensitivity to VAD, whereas wav2vec2 remains substantially worse despite improving with SNR. As on AMI, NS primarily benefits wav2vec2 (especially at low SNR), while its effect on faster-whisper is small.

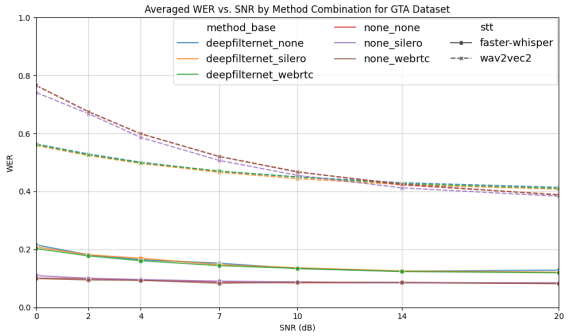

Figure 4: Average WER versus SNR on GTA for all NS/VAD/ASR configurations.

Since the real-time factor (RTF) shows negligible dependence on SNR in our experiments, runtime is analyzed separately using a factorial design over NS/VAD/ASR, without an SNR sweep.

4.1 Statistical Analysis

The main and interaction effects of the pipeline components are quantified using a linear mixed-effects factorial model with a random intercept per utterance. For WER, the following model is fitted:

$$\mathrm{WER}_i = \beta_0 + \mathbf{X}_i\beta + \beta_1\,\mathrm{SNR}_i + \beta_2\,\mathrm{SNR}_i^2 + u_{a(i)} + \varepsilon_i,$$

where X_i encodes the full factorial design over noise suppression (NS), VAD, and ASR backend (including all two- and three-way interactions), and also includes their interactions with SNR_i. The term $u_{a(i)} \sim \mathcal{N}(0,\sigma_u^2)$ is a random intercept for the audio segment a(i) to account for repeated measurements per utterance, and $\varepsilon_i \sim \mathcal{N}(0,\sigma^2)$ is the residual. Runtime is analyzed analogously using $\log(\mathrm{RTF})$ as the dependent variable.

Across both datasets, the mixed-effects factorial analysis shows that WER is dominated by the ASR backend: wav2vec2 yields markedly higher error rates than faster-whisper across configurations. Noise suppression (DeepFilterNet) exhibits a significant but ASR-dependent effect, being neutral to slightly adverse for faster-whisper while mitigating the elevated WER of wav2vec2 (NS$\times$ASR). SNR has a strong nonlinear influence on WER with diminishing gains at higher SNR, and significant ASR$\times$SNR and NS$\times$SNR interactions indicate that wav2vec2 benefits more from improving acoustic conditions, whereas denoising effectiveness decreases as SNR increases. Dataset-specific differences are observed for VAD: on GTA, VAD choice (Silero/WebRTC) does not produce measurable effects on WER, while on AMI VAD effects are statistically detectable (with WebRTC showing the largest degradation and Silero a smaller penalty), partially attenuated when wav2vec2 is used; higher-order interactions (including NS$\times$ASR$\times$VAD) further indicate non-additive coupling among pipeline components.

For runtime, the mixed-effects analysis of $\log(\mathrm{RTF})$ indicates that computational performance is primarily governed by the ASR backend and strong interactions with preprocessing. Noise suppression and VAD generally increase processing time, whereas wav2vec2 is faster in isolation; however, its speed advantage is substantially reduced when combined with DeepFilterNet and VAD, demonstrating that runtime cannot be modeled as an additive sum of module costs and must be assessed at the level of complete pipeline configurations. SNR plays a secondary role for runtime: on GTA it introduces a statistically significant but comparatively small nonlinear modulation with component-dependent interactions, while on AMI it shows no significant main effect and only weak interaction terms. Overall, accuracy is driven by ASR selection and SNR, with backend-dependent denoising effects and dataset-specific VAD sensitivity, whereas runtime is dominated by architectural configuration and its non-additive couplings across stages.

As a secondary confirmation of the mixed-effects factorial analysis, we additionally report partial effect sizes (η_p^2) from a classical factorial ANOVA. The motivation is twofold: (i) η_p^2 provides an intuitive, standardized summary of the relative importance of factors and interactions that is easy to compare across datasets and metrics, and (ii) it serves as a robustness check that the qualitative ranking of influential components is not an artifact of the mixed-effects specification. Since ANOVA assumes independent observations and our design contains repeated measurements per audio segment, this analysis is used for descriptive corroboration only; all inferential conclusions are based on the MixedLM models.

For WER (Fig.5), the effect-size ranking is consistent with the MixedLM findings: the ASR backend explains the largest share of variance for both datasets ($\eta_p^2 \approx 0.95$ on AMI and ≈ 0.99 on GTA). Noise suppression and its interaction with ASR (NS$\times$ASR) also contribute non-negligibly, supporting backend-dependent denoising effects. Dataset-specific differences are visible for VAD: its contribution is negligible on GTA ($\eta_p^2 \approx 0$) but noticeable on AMI ($\eta_p^2 \approx 0.33$), matching the MixedLM result that VAD effects are detectable primarily for AMI.

For runtime (RTF; Fig.6), partial η_p^2 values are uniformly high across ASR, VAD, NS, and their interactions, indicating that runtime is largely configuration-driven and exhibits strong non-additive

coupling between stages. This corroborates the mixed-effects runtime analysis, where large interaction terms imply that computational cost cannot be decomposed into independent module contributions.

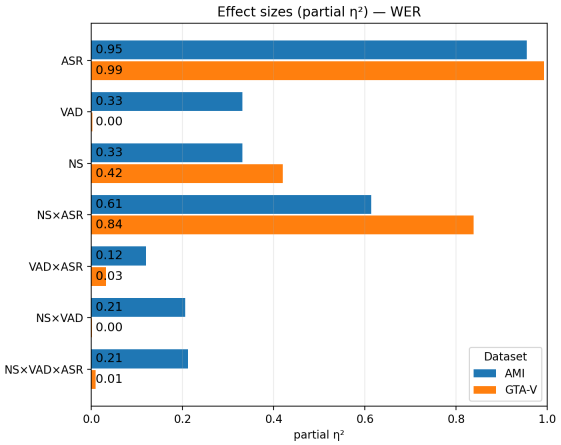

Figure 5: Partial effect sizes (η_p^2) from factorial ANOVA for WER across AMI and GTA-V.

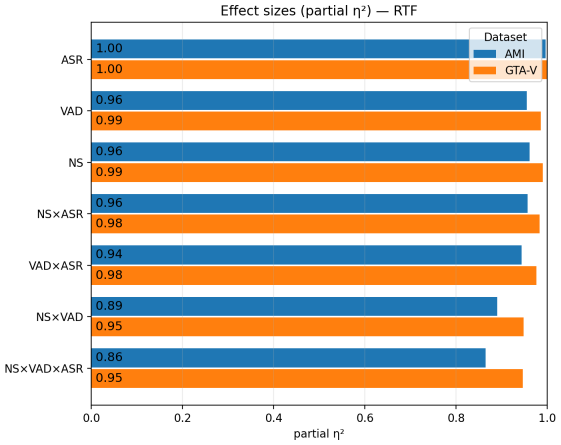

Figure 6: Partial effect sizes (η_p^2) from factorial ANOVA for RTF across AMI and GTA-V.

5 CONCLUSIONS

This paper presented a controlled, pipeline-level evaluation of a three-stage ASR chain comprising noise suppression (none vs. DeepFilterNet), VAD-based segmentation (none, Silero, WebRTC), and ASR backends (wav2vec 2.0, faster-whisper large-v3). Experiments were conducted under additive in-vehicle noise using a representative car-noise profile and speech-active SNR calibration over multiple SNR levels, while reporting both recognition accuracy (WER) and computational efficiency (RTF).

Across both datasets, the principal outcome is that ASR backend choice dominates the accuracy–efficiency trade-off. Faster-whisper consistently achieved lower WER, whereas wav2vec2 incurred higher error rates but offered lower computational cost when used without additional stages. Noise suppression and VAD introduced non-additive, model-dependent effects: denoising tended to benefit wav2vec2 under noisy conditions while being neutral to slightly adverse for faster-whisper, and VAD effects were dataset-dependent (negligible on GTA, but statistically detectable on AMI). The mixed-effects factorial analysis further revealed strong interactions between pipeline stages, demonstrating that preprocessing gains cannot be treated as independent and should be assessed at the level of complete pipeline configurations.

Regarding the evaluation corpora, AMI represents meeting-style conversational speech, whereas GTA provides scripted, studio-recorded utterances. Despite these differences, we observed consistent qualitative dynamics across datasets: WER improved nonlinearly with SNR, the relative ordering of ASR backends remained stable, and backend-dependent denoising effects were reproduced. At the same time, the strength of VAD-related effects differed across datasets. Overall, this cross-dataset agreement supports using GTA-style scripted, high-quality recordings as a practical benchmark for controlled pipeline studies, provided that additive noise is calibrated in a reproducible manner.

As a practical guideline, the best accuracy-first configuration is obtained by selecting faster-whisper and avoiding unnecessary preprocessing stages: DeepFilterNet and VAD do not yield consistent WER gains for faster-whisper and may increase runtime. Conversely, the best efficiency-first configuration is obtained by using wav2vec2 with minimal preprocessing, while enabling DeepFilterNet when operating at low SNR, where it provides the most consistent accuracy improvements for wav2vec2.

Future work. The evaluation uses a single representative in-car noise profile and fixed off-the-shelf component settings; results may vary under different noise types, domains, or hardware constraints. In addition, the number of long-form recordings is limited for some datasets, which constrains statistical stability. Future work will extend the benchmark to multiple noise profiles and environments, include additional recognizers and streaming-oriented latency metrics, and analyze segmentation quality with dedicated boundary/error measures alongside WER and RTF.

6 ACKNOWLEDGMENTS

This work was supported by the European Regional Development Fund (ERDF/EFRE) and the State of Saxony-Anhalt within the programme *Sachsen-Anhalt WISSENSCHAFT Forschung und Innovation (EFRE) 2021–2027*, project ReSeDiUm (grant no. ZS/2023/12/182669).

We acknowledge support by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) and the Open Access Publishing Fund of Anhalt University of Applied Sciences.

REFERENCES

[1] J. Ramírez, J. C. Segura, C. Benítez, Á. de la Torre, and A. Rubio, "A new voice activity detector using subband order-statistics filters for robust speech recognition," in ICASSP, 2004.

[2] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in NeurIPS, 2020.

[3] K. Kuhn, V. Kersken, B. Reuter, N. Egger, and G. Zimmermann, "Measuring the accuracy of automatic speech recognition solutions," ACM Transactions on Accessible Computing, vol. 16, 12 2023.

[4] S. Braun and H. Gamper, "Effect of noise suppression losses on speech distortion and asr performance," in ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2022, pp. 996–1000.

[5] J. H. Ko, J. Fromm, M. Philipose, I. Tashev, and S. Zarar, "Limiting numerical precision of neural networks to achieve real-time voice activity detection," in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 2236–2240.

[6] M. R. Prasad, S. B. Gowda, M. B. Talawar, and N. Jagadisha, "Integrated noise suppression techniques for enhancing voice activity detection in degraded environments," International Journal of Speech Technology, vol. 27, pp. 987–995, 2024.

[7] S. Tong, N. Chen, Y. Qian, and K. Yu, "Evaluating vad for automatic speech recognition," in 2014 12th International Conference on Signal Processing (ICSP), 2014, pp. 2308–2314.

[8] K. Yamamoto, R. Takeda, and K. Komatani, "Analysis of voice activity detection errors in API-based streaming ASR for human-robot dialogue," in Proceedings of the 15th International Workshop on Spoken Dialogue Systems Technology. Bilbao, Spain: Association for Computational Linguistics, May 2025, pp. 245–253. [Online]. Available: https://aclanthology.org/2025.iwsds-1.26/

[9] C. Arriaga, A. Pozo, J. Conde, and A. Alonso, "Assessing latency in asr systems: A methodological perspective for real-time use," 2025. [Online]. Available: https://arxiv.org/abs/2409.05674

[10] A. Sapeha, E. Sariiev, M. Sapeha, I. Kovan, S. Rajanayagam, K. Karpov, M. Gering, D. Kachan, and E. Siemens, "GTA-NarrativeTraj: Language-aware trajectory prediction from GPS and dialogue in an open-world simulator," in Proc. Int. Conf. Appl. Innov. IT, vol. 13, no. 5, pp. 193–199, doi: 10.25673/122853.

[11] J. Carletta, "Unleashing the killer corpus: experiences in creating the multi-everything ami meeting corpus," Language Resources and Evaluation, vol. 41, no. 2, pp. 181–190, 2007.

[12] D. Orel and H. A. Varol, "Noise-robust automatic speech recognition for industrial and urban environments," in IECON 2023- 49th Annual Conference of the IEEE Industrial Electronics Society, 2023, pp. 1–6.

[13] M. Kolbæk, Z.-H. Tan, S. H. Jensen, and J. Jensen, "On loss functions for supervised monaural time-domain speech enhancement," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 825–838, 2020.