

Self-Supervised Voice Denoising Network for Multi-Scenario Human-Robot Interaction.

LI, Mu, XU, Wenjin, ZENG, Chao and WANG, Ning

Available from Sheffield Hallam University Research Archive (SHURA) at:

<https://shura.shu.ac.uk/36218/>

This document is the Published Version [VoR]

Citation:

LI, Mu, XU, Wenjin, ZENG, Chao and WANG, Ning (2025). Self-Supervised Voice Denoising Network for Multi-Scenario Human-Robot Interaction. *Biomimetics*, 10 (9): 603. [Article]

Copyright and re-use policy

See <http://shura.shu.ac.uk/information.html>

Article

Self-Supervised Voice Denoising Network for Multi-Scenario Human–Robot Interaction

Mu Li ¹, Wenjin Xu ¹, Chao Zeng ² and Ning Wang ^{3,*}

¹ Key Laboratory of Autonomous Systems and Networked Control, Ministry of Education, College of Automation Science and Engineering, South China University of Technology, Guangzhou 510640, China; 202320116663@mail.scut.edu.cn (M.L.); wenjinxu@ieee.org (W.X.)

² Department of Computer Science, University of Liverpool, Liverpool L69 3BX, UK; chaozeng@ieee.org

³ School of Computing and Digital Technologies, Sheffield Hallam University, Sheffield S1 2NU, UK

* Correspondence: ningwang.ac@gmail.com

Abstract

Human–robot interaction (HRI) via voice command has significantly advanced in recent years, with large Vision–Language–Action (VLA) models demonstrating particular promise in human–robot voice interaction. However, these systems still struggle with environmental noise contamination during voice interaction and lack a specialized denoising network for multi-speaker command isolation in an overlapping speech scenario. To overcome these challenges, we introduce a method to enhance voice command-based HRI in noisy environments, leveraging synthetic data and a self-supervised denoising network to enhance its real-world applicability. Our approach focuses on improving self-supervised network performance in denoising mixed-noise audio through training data scaling. Extensive experiments show our method outperforms existing approaches in simulation and achieves 7.5% higher accuracy than the state-of-the-art method in noisy real-world environments, enhancing voice-guided robot control.

Keywords: human–robot interaction; voice denoising; self-supervised learning; data synthesis



Academic Editors: Junzhi Yu, Jun Huang and Ruiya Li

Received: 15 June 2025

Revised: 24 August 2025

Accepted: 6 September 2025

Published: 9 September 2025

Citation: Li, M.; Xu, W.; Zeng, C.; Wang, N. Self-Supervised Voice Denoising Network for Multi-Scenario Human–Robot Interaction. *Biomimetics* **2025**, *10*, 603. <https://doi.org/10.3390/biomimetics10090603>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, with the development of Vision–Language–Action (VLA) models [1–4] facilitating significant advancements in robotic manipulation, there has been growing interest in integrating action prompts and voice interaction. However, one of the most critical and unresolved issues is ensuring effective command recognition in human–robot interaction (HRI). Current VLA models often rely on voice encoders (e.g., LLama [5], T5 [6]) to understand task prompts during HRI in real-world applications, but may miss critical instructions due to environmental noise [7]. This leads to the first research question: **(1) How can audio commands be stably transmitted in real-world interactive environments?**

Regarding the first question, supervised training strategies [8,9] have emerged as reliable tools for voice denoising during HRI. Nevertheless, this method requires collecting multiple high-quality audio samples for training, a process that demands a carefully controlled recording environment, resulting in both increased complexity and higher cost. Thanks to previous works [10–12], researchers have employed self-supervised learning to train models for generating voice sequences. Noise variation can also be estimated from noisy audio datasets [13] through latent audio features [14], while Large-Language Model (LLM) [15] enables audio-to-text conversion followed by semantic reasoning to interpret distorted speech. However, many of these systems still struggle in real-world

robotic applications, particularly in multi-scenario or multi-speaker environments. First, existing voice datasets fail to adequately simulate ambient sound, particularly in cases of overlapping human voice, limiting practical performance. Second, interactive audio exhibits discrete properties, limiting the effectiveness of time-domain denoising for frequency feature reconstruction. Finally, LLMs are prone to hallucinations, generating inaccurate or unsafe responses that may produce erroneous robotic commands with potentially harmful consequences. This leads to two further research questions: **(2) How to scale training data for enhanced real-world denoising capability?** **(3) How to effectively denoise audio for real-world HRI applications?**

Through answering questions (1) to (3), a self-supervised voice denoising approach is proposed to enhance robotic audio command transmission quality in real-world applications, as illustrated in Figure 1. In contrast to previous work, our model employs realistic synthetic data (i.e., *multi-environment noise* or *overlapping voices*) in self-supervised training and integrates the convolution and attention network to extract key features from the audio magnitude and phase information. The intuition behind our design is the integration of *scaling data* and *self-supervised denoising* capabilities, where both training inputs and targets are directly generated from noisy signals. The specialized denoising architecture with regularized loss functions delivers results matching supervised training, providing reliable audio for VLAs. A comprehensive comparison of the evaluation results shows that our model outperforms state-of-the-art methods in *simulation benchmarks* and achieves a 7.5% improvement in *real-world manipulation tasks*.

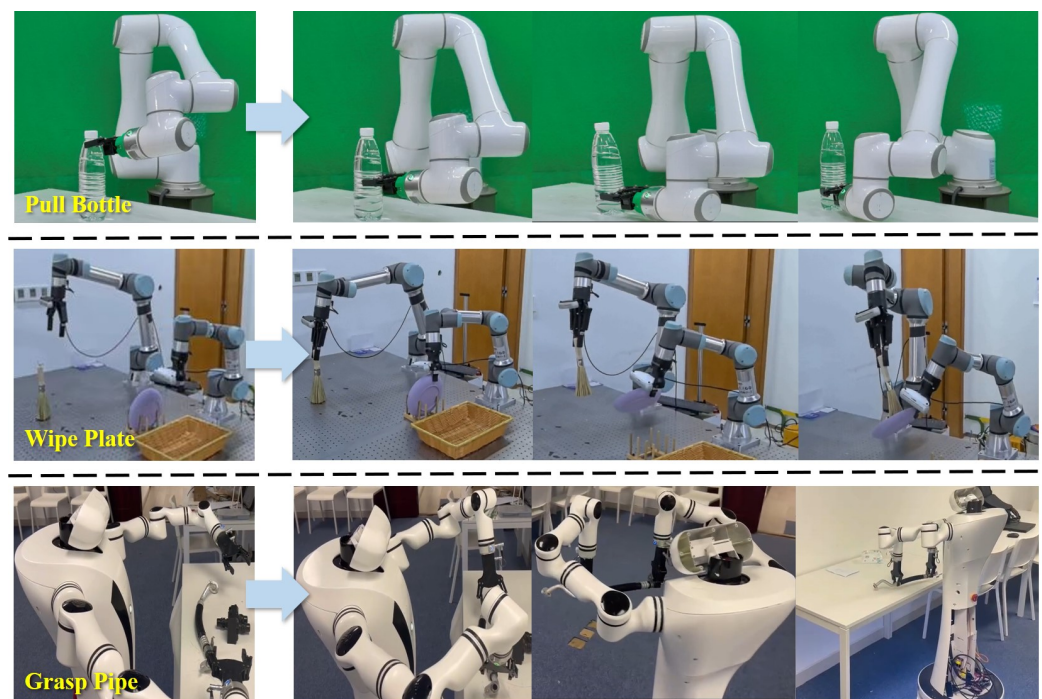


Figure 1. Three different tasks are executed by voice command. Their initial conditions are shown in the first column. The robot starts from an arbitrary position (second column), recognizing voice prompts in noisy environments and executing instruction actions (third and last column).

The contributions of this work are summarized below:

1. We propose training the denoising model with synthetic data, using data scaling to enhance robustness in model application.
2. We present a self-supervised method using a convolutional network to denoise both magnitude and phase components in the frequency domain.

3. We demonstrate proof-of-concept of the proposed approach in both simulation and hardware. For the benefit of the community, we will open-source part of the scaled dataset at <https://tinyurl.com/2ybfz3jt> (accessed on 25 August 2025).

2. Related Work

Synthetic Data Application. Synthetic data can rapidly generate large-scale training sets, enhancing the practicality of datasets [16–18]. Recent advances in voice synthesis include generating various pulses, tones, and altered timbres [19–21]. In contrast, HRI applications require interactive voice in diverse environments. Prior work using public audio datasets for denoising model training often shows poor generalization in complex real-world scenarios. In addition, collecting pure datasets requires significant expense. Therefore, we aim to use audio synthesis to generate diverse datasets for model training, covering voices in multiple scenarios.

Voice Denoising Network. Noise–Clean Training (NCT) [11,22–24], a common audio denoising method, uses clean audio signals as training targets. However, it struggles to handle variable environmental noise in real-world applications. Thus, noise-only training has recently gained wide attention [25,26]; the network can be trained with multiple independent noisy voice signals using self-supervised methods. Xu [14] leverages silent intervals in speech signal to isolate noise-only periods for time-varying noise estimation, and studies like [10,11] demonstrate that voice denoising models can be built from datasets containing diverse noise. Inspired by Noisy-to-Noisy, Wu [12] employs a complex-valued network architecture for multi-dimensional audio denoising and [13] constructs mixed-audio datasets combining robot ego-voice, fan noise, and concurrent human speech to improve low-frequency noise suppression. However, these methods fail to process overlapping audio in multi-speaker environments, making it difficult to extract target signals in interaction dialogues. In our work, we improve the model’s denoising capability for unseen environments by synthesizing audio from diverse scenarios and expanding the training dataset.

Voice Control in HRI. Voice can be used in HRI to deliver action instructions and guide robot manipulation [27–31]. The development of LLMs [5,6,15] has led to significant advancements in voice reasoning, particularly in environments requiring high adaptability and generation. Lai [28] proposed a natural multimodal fusion-based HRI framework (NMM-HRI) that combines voice commands and deictic postures, leveraging LLMs to generate robot action sequences. In [32], the PlanCollabNL framework combines LLMs with task planning to generate robot action subgoals from abstract natural language goals. Building on the success of VLA models [2–4], voice interaction is gaining significant attention for its ability to provide operational conditions and task requirements for robotic actions through voice prompts. Most VLAs [2–4,33,34], treat the frozen-weight LLM as a policy to generate action commands for robot operation. However, none of the works address that audio distortion in noisy environments further reduces the LLM model’s instruction recognition accuracy. In this work, we aim to use a self-supervised denoising algorithm to optimize audio quality, enabling stable transmission of audio prompts to VLAs.

3. Preliminaries

3.1. Human–Robot Voice Interaction System

In the human–robot voice interaction scenario, we consider there are two equidistant microphone arrays (i.e., simulating human binaural distribution) to capture the interactive sound in the environment, as shown in Figure 2. This ensures recorded voice signals vary randomly and nonlinearly over time. The interaction architecture uses the Z-transform to

convert signals to the time–frequency domain, with the collected signal $M(z)$ composed of the source sound $T(z)$ and the mixing matrix $A(z)$. The following equation holds:

$$M(z) = T(z) \cdot A(z), \quad (1)$$

$$A(z) = \begin{bmatrix} 1 & a_0 z^{-d_0} \\ a_1 z^{-d_1} & 1 \end{bmatrix}, \quad (2)$$

where d_i and a_i denote the fractional delay in samples and the attenuation factor along the indirect signal path, respectively, and $i = 0, 1$ represents the two recording microphones. However, one challenge in voice denoising is that $M(z)$ contains complex environmental interference, and direct application of audio enhancement can easily distort the source sound. It is valuable to separately denoise frequency-domain information (audio magnitude or phase) to enable effective extraction of key semantic information from discontinuous mixed-audio signals.

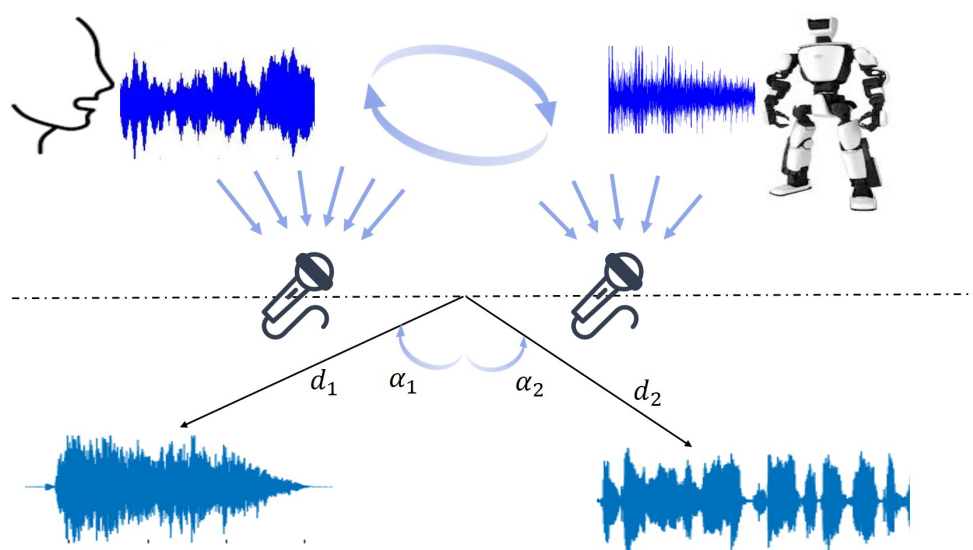


Figure 2. The process of blind source separation: contaminated speech picked up by multiple microphones first, and then the human speech and background noise are separated.

3.2. Self-Supervised Denoising Strategy Foundation

A clean voice dataset for training models demands a huge amount of data; Noise2Noise (N2N) [35] introduces a self-supervised optimization method to overcome the lack of clean data. It uses pairs of independent noisy data m and n (which share the same ground-truth s) for learning. Kashyap [11] proposed that symmetric noisy voice signals in the same environment can also serve as training pairs. Equation (3) indicates the loss function in denoising network f_θ with respect to parameter θ .

$$\arg \min_{\theta} \mathbb{E} \|f_\theta(m) - n\|_2^2. \quad (3)$$

As reported in [36], denoising performance comparable to supervised training with ground truth can be achieved by minimizing Equation (4) as follows:

$$\arg \min_{\theta} \mathbb{E} \|f_\theta(m) - s\|_2^2. \quad (4)$$

Wu [12] proposed that an audio sub-sampler P can be used to transform the noisy audio signal x into a training pair (p_1, p_2) , i.e., $\varepsilon = \mathbb{E}(p_2(n)) - \mathbb{E}(p_1(n))$, which can be directly utilized to train the denoising network:

$$\arg \min_{\theta} \mathbb{E} \|f_{\theta}(p_1(x)) - p_2(x)\|_2^2. \quad (5)$$

Given an ideal denoiser f_{θ}^* trained on clean data and defined by Equation (5), it guarantees that the output of f_{θ}^* for noisy input x is the clean voice y , i.e., $y = f_{\theta}^*(x)$. Therefore, the optimal network is as follows:

$$\begin{aligned} & \mathbb{E}\{f_{\theta}^*(p_1(x)) - p_2(x) - (p_1(f_{\theta}^*(x)) - p_2(f_{\theta}^*(x)))\} \\ &= p_1(y) - \mathbb{E}\{p_2(x)\} - (p_1(y) - p_2(y)) \\ &= p_2(y) - \mathbb{E}\{p_2(x)\} = 0. \end{aligned} \quad (6)$$

Neighbor2Neighbor (Ner2Ner) [37] indicates that a regularization term μ can address the non-zero difference between $p_1(x)$ and $p_2(x)$ in θ . Inspired by [37], we add spectral constraints to minimize the gap between the output and the target audio, while [38] demonstrates that spectral structure optimization enhances the ability of the convolutional network in voice enhancement. Hence, instead of directly optimizing Equation (5), a constrained optimization problem incorporating Equation (6) and the spectral optimization loss is considered:

$$\begin{aligned} & \mathbb{E} \|f_{\theta}(p_1(x)) - p_2(x) - (p_1(f_{\theta}(x)) - p_2(f_{\theta}(x)))\|_2^2 \\ &+ \arg \min_{\theta} \mathbb{E} \|f_{\theta}(p_1(x)) - p_2(x)\|_2^2 \\ &+ \delta \mathbb{E} \|\log(S(f_{\theta}(p_1(x)))) - \log(S(s_2(x)))\|_2^2, \end{aligned} \quad (7)$$

where $S(f_{\theta}(p_1(x)))$ represents the optimized audio spectrum, and $S(s_2(x))$ represents the target audio spectrum.

4. Methodology

Project Overview. The proposed framework comprises three main components: (1) synthetic voice generation for training data expansion (Section 4.1); (2) a self-supervised denoising framework for environmental noise suppression (Sections 4.2–4.5); (3) real-world deployment of denoising model in human–robot interaction scenarios (Section 4.6). The overall workflow of the approach is depicted in Figure 3.

Problem Formulation. The voice denoising model can endow a robot with the ability to recognize commands in *noisy and audio-overlapping* environments. This enables the robot to perform various tasks by leveraging a specific level of environmental perception. Formally, our goal is to enhance the ability of the denoising model to extract and understand conversational information in complex environments.

Model Architecture. As illustrated in Figure 4, a noisy voice signal x is passed through a sub-sampler to generate a noisy training data pair $(p_1(x), p_2(x))$, sharing the same ground truth. We transform $p_1(x)$ into STFT representations as input to the training network, and design the encoding and decoding stages as a hierarchical symmetric U-shaped network. The real and imaginary domains of the audio are denoised separately, with the loss between $f_{\theta}(p_1(x))$ and $p_2(x)$ being calculated to update network weights.

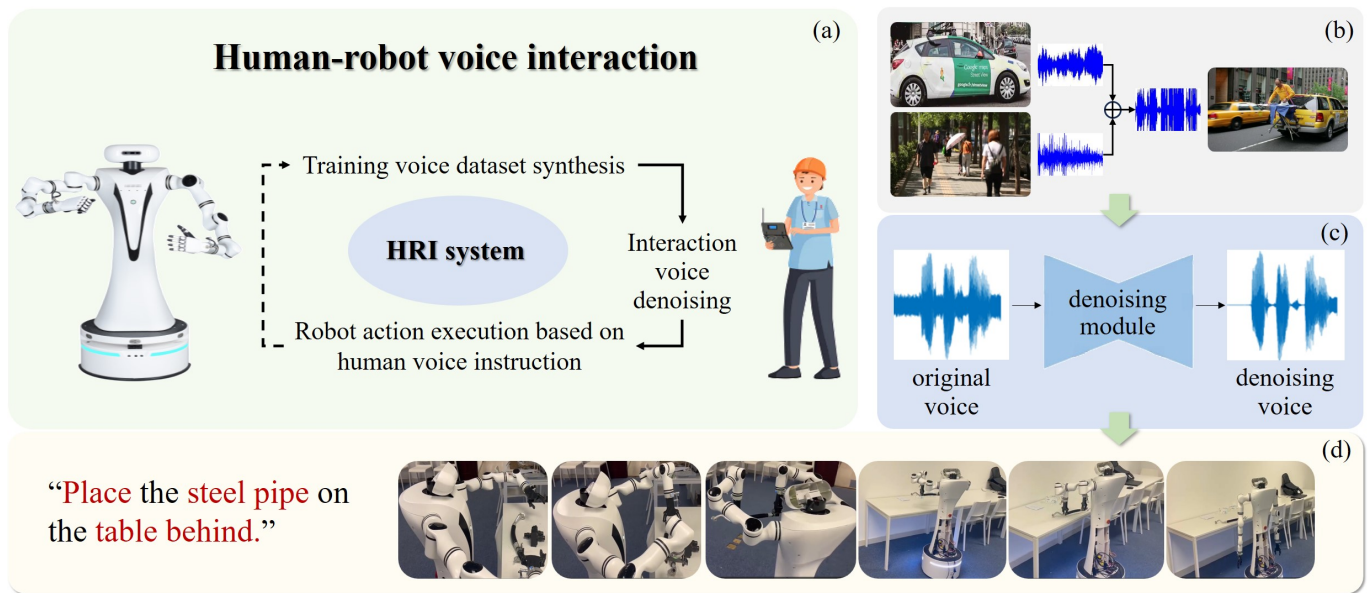


Figure 3. Overview of HRI architecture. (a) presents the system framework, which is componentized into three parts: (b) shows the data synthesis process, i.e., selecting two independent audio sources (car sounds and human voices) and fusing them to generate street audio. (c) indicates voice command denoising, and (d) demonstrates the robot executing actions based on interactive audio instructions.

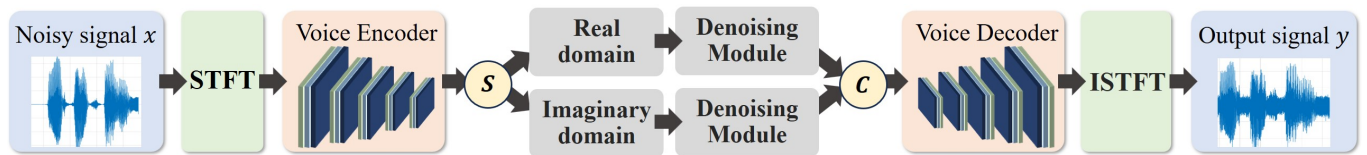


Figure 4. Overall framework of the proposed self-supervised network. During encoding, the network dynamically extracts multi-scale local and global features. The decoder then fuses these features to reconstruct voice spectral features. S separates signal frequency spectrum; C connects real and imaginary domains.

4.1. Voice Dataset Synthesis

To enhance adaptability to complex environments with noisy human voices, as shown in Figure 3b, two training datasets are generated via audio synthesis, covering *multiple scenarios* and *overlapping voices*:

- **Multi-scenario dataset:** To synthesize audio in a realistic voice dataset across various environments, we employ PyDub (<https://pydub.com/>, (accessed on 25 August 2025)) to overlap diverse UrbanSound8K (US8K) [39] noise onto clean audio signals from the Voice Bank dataset [40], which includes 28 speakers for training and 2 for testing. The complete noisy speech sample is generated by truncating or repeating the noise to cover the entire voice segment.
- **Multi-speaker dataset:** Using the same methodology as for generating the multi-scenario dataset, we combine the TIMIT dataset [41], which contains recordings of 630 speaker utterances, with the Voice Bank dataset [40] to generate an audio dataset of overlapping speaker voices. However, direct training on multi-speaker audio fails to achieve effective noise reduction performance. Hence, we employ Kullback–Leibler Divergence [42] to post-process the synthesized audio, thereby generating training data that can be effectively utilized in real-world HRI scenarios (more details in Appendix C).

4.2. Model Encoding and Decoding Stages

Voice Encoder. The encoding stage consists of four consecutive encoders for multi-scale voice spectral feature extraction. Each encoder's downsampling block contains a complex convolution layer with a complex batch normalization layer (CBN) and a multilayer perceptron (MLP). It employs a complex filter $W = (A + iB)$. For a complex vector $h = c + id$, it performs complex convolution via two independent real convolutions formulated as $W * h = (A * c - B * d) + i(B * c + A * d)$.

Voice Decoder. The decoder and encoder are symmetric architectures. Each decoder's upsampling block contains a complex transposed convolution layer, followed by CBN and nonlinear MLP [3]. The decoding stage focuses on reconstructing the spectrogram locally and globally using voice features.

4.3. Self-Supervised Denoising Strategy

We illustrate the optimization function of the self-supervised strategy in Section 3.2, which incorporates the audio spectrum constraint. To prove the feasibility of the optimization problem, the *function rationalization* and *solution existence* of Equation (7) are analyzed.

Non-negativity of the Objective Function. For any θ , the squared Euclidean norm is non-negative; since expectation is the integral of the objective function, each term of the objective function is non-negative. We set

$$\begin{cases} L_1(\theta) = \mathbb{E} \|f_\theta(p_1(x)) - p_2(x)\|_2^2, \\ L_2(\theta) = \mathbb{E} \|f_\theta(p_1(x)) - p_2(x) - p_1(f_\theta(x)) + p_2(f_\theta(x))\|_2^2, \\ L_3(\theta) = \mathbb{E} \|\log(S(f_\theta(p_1(x)))) - \log(S(s_2(x)))\|_2^2. \end{cases} \quad (8)$$

Then, the function expectation is expressed as follows:

$$\begin{cases} \int_{x,y} \|L_1(\theta)\|_2^2 f_{X,Y}(x,y) dx dy \geq 0, \\ \int_{x,y} \|L_2(\theta)\|_2^2 f_{X,Y}(x,y) dx dy \geq 0, \\ \int_{x,y} \|L_3(\theta)\|_2^2 f_{X,Y}(x,y) dx dy \geq 0, \end{cases} \quad (9)$$

where $f_{X,Y}(x,y)$ represents the joint probability density function of two-dimensional random variables x and y . The objective function $L(\theta) = L_1(\theta) + \mu L_2(\theta) + \delta L_3(\theta) \geq 0$, where $\delta \geq 0$.

Existence of Solutions within the Parameter Space. The particular parameter space Ω is assumed to be a compact set (bounded and closed) in \mathbb{R}^n . Since f_θ is continuous with respect to θ (a differentiable function is continuous) and p_1 and p_2 are measurable functions, by the property of the composition of continuous functions. There exist

$$\begin{cases} g_1(\theta, x) = \|f_\theta(p_1(x)) - p_2(x)\|_2^2, \\ g_2(\theta, x) = \|f_\theta(p_1(x)) - p_2(x) - p_1(f_\theta(x)) + p_2(f_\theta(x))\|_2^2, \\ g_3(\theta, x) = \|\log(S(f_\theta(p_1(x)))) - \log(S(s_2(x)))\|_2^2. \end{cases} \quad (10)$$

Suppose that f_θ is bounded, i.e., there exist constants C_1 and C_2 such that $|f_\theta(p_1(x))| \leq C_1$ and $|p_1(f_\theta(x))| \leq C_2$ for all $\theta \in \Omega$.

It can be shown that there exist integrable functions $M_1(x)$ and $M_2(x)$ such that $|g_1(\theta, x)| \leq M_1(x)$ and $|g_2(\theta, x)| \leq M_2(x)$. Since logarithmic operations exhibit continuity within their respective domains, $g_3(\theta, x)$ is continuous in θ , and there exists an integrable function $|g_3(\theta, x)| \leq M_3(x)$ for all $\theta \in \Omega$.

Hence, the expectations $L_1(\theta)$, $L_2(\theta)$, and $L_3(\theta)$ are continuous in θ . Therefore, $L(\theta) = L_1(\theta) + L_2(\theta) + \delta L_3(\theta)$ is continuous in θ . According to the Lebesgue dominated convergence theorem [43], on the compact parameter space Ω , there always exists $\theta^* \in \Omega$ such that $L(\theta^*) = \min L(\theta)$.

4.4. Denoising Network

Conventional 2D convolution kernels with fixed geometries usually struggle to capture detailed features in complex spectral data, particularly in sub-sampled voice signals. As formulated in Equations (1) and (2), directly separating the target voice from the noisy signal $M(z)$ in the noisy environment is challenging.

To address this limitation, separating the real and imaginary components of the audio spectrum enables the denoising model to optimize magnitude and phase independently. Audio processing without component separation fails to establish a clear phase spectrogram structure [44], which is essential for perceptual quality in voice denoising. Hence, we aim to estimate the real and imaginary spectrograms of the target voice from the noisy signal via complex spectral mapping, simultaneously capturing *signal structure* and preserving *phase information*.

Therefore, our goal is to separately denoise the information in different dimensions (*real and imaginary*) to enable more effective extraction of audio features. The denoising network model π is composed of a convolutional multi-head attention architecture followed by a ResNet module (shown in Figure 5). Taking the real part X_r and imaginary part X_i of the voice spectral data as input, the model produces a complex-valued output Y defined as

$$Y_r = \pi(X_r) - \pi(X_i), \quad (11)$$

$$Y_i = \pi(X_r) + \pi(X_i), \quad (12)$$

where $\pi(X_r)$ and $\pi(X_i)$ represent the real and imaginary parts of the denoising model, and $Y = Y_r + jY_i$ denotes the model's complex value output. This approach enables precise processing and reconstruction of spectrogram amplitude and phase data, while preserving voice contextual details. It improves the ability of robots to accurately understand key instructions during interactions (more details in Section 4.6).

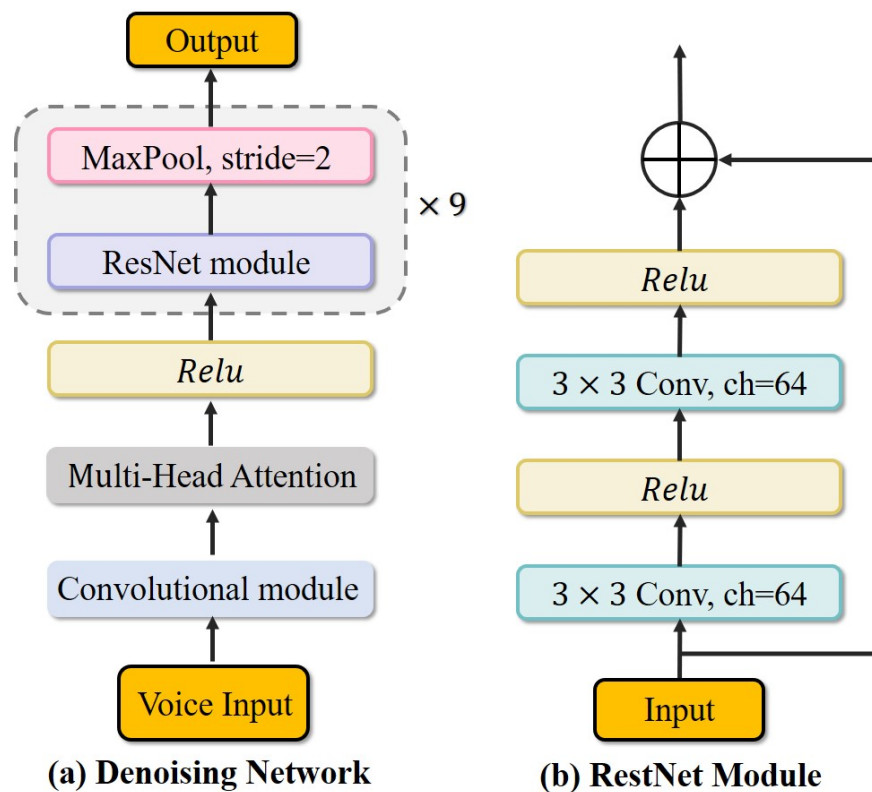


Figure 5. Detailed neural network architecture of the denoising module.

4.5. Training Loss

Our primary objective is to satisfy the constraints of sub-sampled audio pairs by solving the optimization function in Equation (7) for voice denoising. Hence, the loss function consists of L_{basic} and L_{reg} .

Basic Loss. We define the basic loss L_{basic} [45] as a combination of the waveform loss and an STFT loss to fit the target audio waveform:

$$L_{\text{basic}} = \frac{1}{TF} \sum_{t=0}^{T-1} \sum_{f=0}^{F-1} (|S_r(t, f)| + |S_i(t, f)| - |\hat{S}_r(t, f)| - |\hat{S}_i(t, f)|) + \frac{1}{N} \sum_{i=0}^{N-1} (n_i - \hat{n}_i)^2, \quad (13)$$

where n_i and \hat{n}_i denote the i -th sub-sampled audio pair and its denoised version, respectively, with N being the total number of speech samples. Let S and \hat{S} be the spectrograms of the sub-sampled and its denoised version, respectively, where T represents the number of frames and F represents the number of frequency bins.

Regularization Loss. We also need to consider the regularization loss discussed in Section 3.2. The regularization loss L_{reg} , formulated as $L_{\text{reg}} = L_{\text{reg1}} + L_{\text{reg2}}$, is defined as follows:

$$L_{\text{reg1}} = \|f_{\theta}(p_1(x)) - p_2(x) - (p_1(f_{\theta}(x)) - p_2(f_{\theta}(x)))\|_2^2, \quad (14)$$

$$L_{\text{reg2}} = \|\log S(f_{\theta}(p_1(x))) - \log S(s_2(x))\|_2^2. \quad (15)$$

The total loss L is the sum of the basic loss and the regularization loss, as $L = L_{\text{basic}} + L_{\text{reg1}} + \delta L_{\text{reg2}}$. The performance of the model can be enhanced by adjusting the value of the parameters δ (for more details see Section 5.3).

4.6. Human–Robot Voice Interaction System

In this work, voice recognition is established as a precondition for robot task execution during HRI. As described in Section 4.4, the denoising model takes noisy speech x as input and outputs audio y , which contains the task instructions for guiding robot action. It enables the robot to more effectively identify key semantics in interaction by reducing background noise and overlapping voices. Based on the action information (i.e., *pick*, *grasp*, *pull*), the system executes various actions to interact with the environment. We develop an interaction system that integrates **voice recognition** and **action execution**. The VLA model is employed to generate robot action, with denoising voice serving as both the supporting condition for language prompts and the prerequisite for action execution.

An example is shown in Algorithm 1. When the interaction command is “Grasp the steel pipe and place it on the table behind”, the robot identifies the key semantics of the task instruction in the voice:

- **Grasp:** target task action;
- **Steel Pipe:** target object;
- **Table Behind:** placing position.

This approach provides sufficient task instructions and action information. Additionally, it offers an efficient approach to verify whether the audio has been effectively denoised. If the key semantic information within the denoised voice remains difficult to discern, the robot will be unable to execute the task action. Meanwhile, to enhance edge-case handling, we design action termination prompts (i.e., *stop*, *no*, *wait*). They can

instantaneously terminate inference actions during HRI, prompting the robot to await subsequent task instructions (see Section 5.5 for a detailed discussion of our test).

Algorithm 1 Human–robot voice interaction

Input: noisy voice
Output: robot action
 ROBOTTASK(noisy_voice)
 interaction_audio \leftarrow **Denoise**(noisy_voice)
 instruction \leftarrow **RecognizeVoice**(interaction_audio)
 action \leftarrow **VLAModel**(instruction)
 return RobotExecute(action)

5. Experiment

In Section 5.2, we compare the denoising ability and generalization results of our method with previous self-supervised methods in simulation environments. The effectiveness of each component is validated in Section 5.3. In Section 5.4, we present application results of our method in real-world scenarios, including single-arm, dual-arm, and humanoid robot tasks. The effectiveness of termination prompts is tested in Section 5.5, demonstrating they can achieve closed-loop voice interaction control.

5.1. Experiment Setup

Dataset. Noisy signals from three distinct environments are used to evaluate algorithm robustness under varied conditions. In addition to the synthetic dataset introduced in Section 4.1, clean speech is augmented with white Gaussian noise to generate a synthetic white-noise training dataset. The training set includes 6000 white-noise voices as *Env1*, randomly selects 10,000 voices from the multi-scenario dataset as *Env2*, and takes 8000 voices from the multi-speaker dataset as *Env3*.

Training Details. In the denoising model, most convolutional layers adopt a 3×3 kernel and a 2×2 stride, while the middle two layers in the encoder and decoder utilize a 2×1 stride. We employ an Adam optimizer with a learning rate of 0.001, and the complex-valued voice spectrograms for model input are obtained via STFT with a 64 ms Hamming window and 16 ms hop size. All experiments were conducted on a workstation equipped with an Intel Core i9-14900K CPU and an NVIDIA RTX 4090 GPU for both data training and evaluation.

5.2. Evaluation in Simulation

Metric. To evaluate the denoising effect quality, we use five metrics in the simulation experiment [46–48]: signal-to-noise ratio (SNR), segmental signal-to-noise ratio (SSNR), wide-band perceptual evaluation of speech quality (PESQ-WB), narrow-band perceptual evaluation of speech quality (PESQ-NB), and short-term objective intelligibility (STOI).

Baselines. To comprehensively evaluate our model, comparisons are made with state-of-the-art baselines in voice denoising and self-supervised learning, including NCT [11], NNT [11], NerNT [10], and ONT [12]. These methods use noisy voice for model training and testing in the Env1–Env3 datasets. For more details on the differences in the training processes of the methods, see Appendix D.

Result. To ensure fair experimental evaluation, all baseline models are pre-trained on the same synthetic datasets. Table 1 shows that our model has achieved notable improvements in different environments and outperformed other baselines. Moreover, it has the ability to denoise in multi-speaker overlapping scenarios. This is because (1) the scaling dataset enhances self-supervised denoising performance across diverse scenarios and speaker backgrounds, and (2) the convolutional multi-head attention network

improves feature extraction from discrete time–frequency representations (e.g., real and imaginary components).

Table 1. Voice denoising evaluation with different methods.

Dataset	Network	SNR	SSNR	PESQ-NB	PESQ-WB	STOI
Env1	NCT	17.323 ± 3.488	4.047 ± 4.738	2.655 ± 0.428	1.891 ± 0.359	0.655 ± 0.18
	NNT	16.937 ± 3.973	3.752 ± 4.918	2.597 ± 0.462	1.943 ± 0.375	0.645 ± 0.018
	NerNT	—	—	—	—	—
	ONT	17.563 ± 2.596	8.389 ± 2.961	2.690 ± 0.347	1.878 ± 0.293	0.833 ± 0.066
	ONT + rTSTM	18.137 ± 2.122	9.077 ± 2.437	2.643 ± 0.317	2.003 ± 0.282	0.839 ± 0.067
	ONT + cTSTM	18.209 ± 2.095	9.088 ± 2.222	2.811 ± 0.288	1.997 ± 0.276	0.847 ± 0.09
	Ours	18.671 ± 3.093	9.628 ± 3.002	2.961 ± 0.508	2.166 ± 0.326	0.864 ± 0.073
Env2	NCT	3.44 ± 3.457	−0.684 ± 3.767	1.773 ± 0.326	1.326 ± 0.190	0.52 ± 0.188
	NNT	3.99 ± 5.451	−0.402 ± 5.084	2.147 ± 0.535	1.55 ± 0.372	0.593 ± 0.221
	NerNT	3.537 ± 3.465	−1.336 ± 3.105	1.787 ± 0.26	1.249 ± 0.126	0.569 ± 0.177
	ONT	5.64 ± 2.18	−0.28 ± 0.69	2.15 ± 0.189	1.76 ± 0.39	0.54 ± 0.56
	ONT + rTSTM	5.71 ± 2.62	−0.23 ± 1.36	2.23 ± 0.27	1.83 ± 0.35	0.68 ± 0.27
	ONT + cTSTM	5.86 ± 3.44	−0.15 ± 3.6	2.41 ± 0.304	1.97 ± 0.21	0.72 ± 0.019
	Ours	6.029 ± 3.431	−0.074 ± 0.993	2.818 ± 0.556	2.123 ± 0.26	0.87 ± 0.23
Env3	NCT	3.75 ± 3.899	−1.915 ± 3.664	1.924 ± 0.313	1.37 ± 0.208	0.562 ± 0.201
	NNT	4.428 ± 3.166	−1.824 ± 4.558	2.445 ± 0.481	1.77 ± 0.41	0.634 ± 0.199
	NerNT	4.464 ± 3.858	−1.837 ± 3.714	2.321 ± 0.351	1.484 ± 0.256	0.651 ± 0.198
	ONT	4.89 ± 1.62	−1.52 ± 2.45	2.58 ± 0.317	1.58 ± 0.273	0.662 ± 0.16
	ONT + rTSTM	5.14 ± 2.63	−1.45 ± 2.36	2.61 ± 0.17	1.69 ± 0.31	0.674 ± 0.18
	ONT + cTSTM	5.327 ± 3.309	−1.391 ± 4.22	2.65 ± 0.49	1.84 ± 0.324	0.69 ± 0.21
	Ours	5.416 ± 1.232	−1.223 ± 2.458	2.727 ± 0.437	1.991 ± 0.189	0.712 ± 0.16

We validate the model’s effectiveness by comparing its denoising performance with ONT [12] using a real-world utterance. Figure 6 presents the spectral changes before and after model processing. As shown in Figure 6a, the original interactive voice is heavily corrupted by noise, making it difficult to directly discern speech features (annotated in red). From Figure 6b,c, it is evident that the proposed method achieves significantly better performance in audio denoising. Figure 6c retains more command audio harmonics with less distortion than Figure 6b, demonstrating our method’s superior performance in a real-world scenario.

Generalization Evaluation. We evaluate the model’s generalization ability in unseen scenarios and real-world multi-speaker voices. Table 2 presents the denoising results of PESQ-NB and SNR under different scenario conditions, using training and testing datasets sourced from *Env2*. The model is trained using interactive voices under industrial noise (i.e., factory environment) and tested using interactive voices with street music (i.e., street environment). The results indicate that the model can successfully adapt to an unseen scenario.

Table 2. Generalization evaluation of unseen scenarios.

Training	Testing	Network	SNR	PESQ-NB
Factory	Street	NCT	3.645 ± 3.676	1.8 ± 0.46
		NNT	3.948 ± 3.285	2.121 ± 0.467
		NerNT	3.799 ± 3.293	2.114 ± 0.459
		ONT	4.38 ± 2.31	2.33 ± 0.39
		ONT + rTSTM	4.65 ± 2.35	2.58 ± 0.74
		ONT + cTSTM	4.91 ± 2.46	2.73 ± 0.4
		Ours	5.11 ± 4.825	2.92 ± 0.435

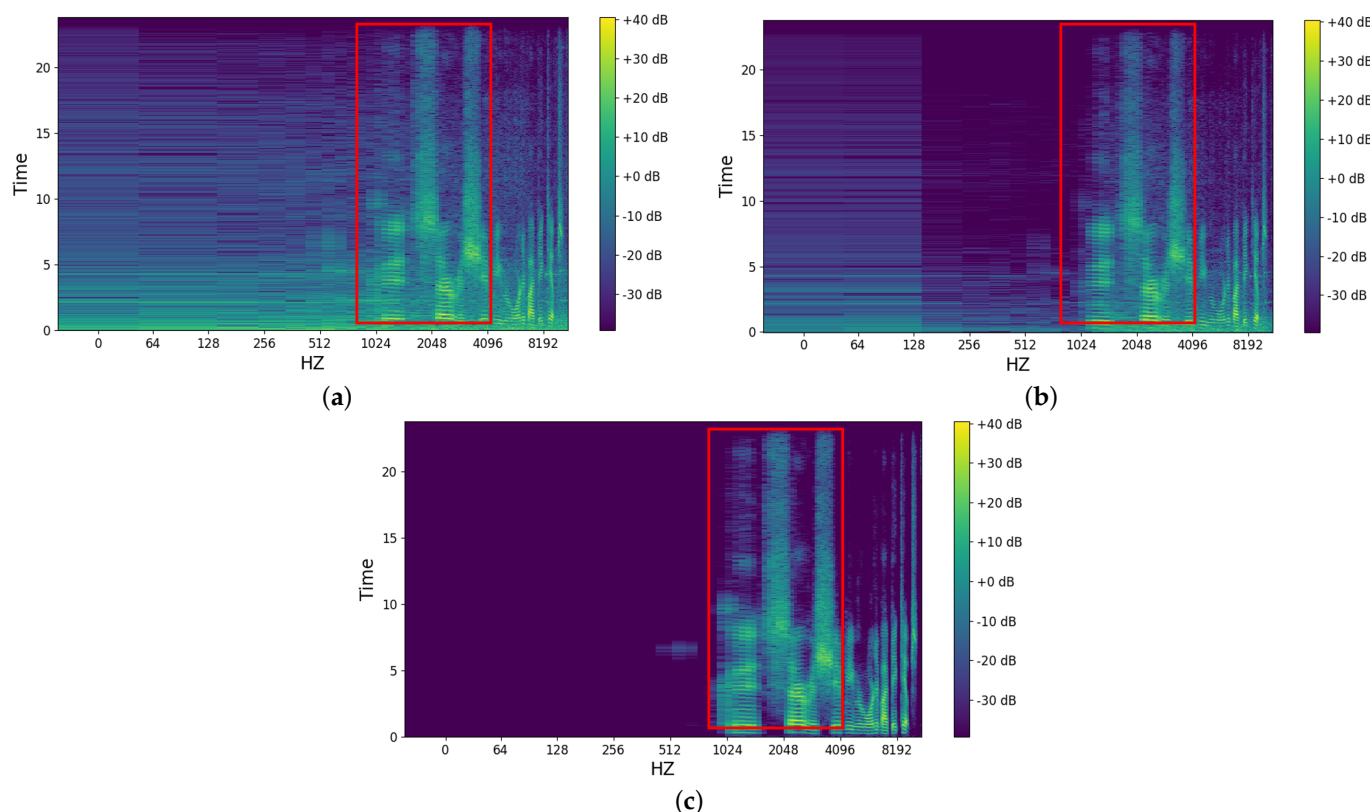


Figure 6. (a) displays the **original** noise signal spectrum. (b) illustrates the denoising output of the **ONT** method, and (c) depicts the output of **our** proposed method. Audio denoising should preserve the command audio signal (**annotated in red**) while eliminating background noise.

Table 3 presents the the PESQ-NB evaluation results for multi-speaker voices from Env3, where k denotes the number of speakers. When $k = 0$ (signifying no overlapping noise), we select three values (0, 1, 2) for the experiment. The presented method maintains effective noise reduction as k increases, achieving the highest voice quality compared to other methods.

Table 3. Generalization evaluation of multi-speaker voices.

Network	$k = 0$	$k = 1$	$k = 2$
NCT	2.172 ± 0.367	1.924 ± 0.313	1.78 ± 0.269
NNT	2.583 ± 0.392	2.445 ± 0.481	2.17 ± 0.635
NerNT	2.56 ± 0.26	2.321 ± 0.351	2.13 ± 0.29
ONT	2.63 ± 0.16	2.58 ± 0.317	2.24 ± 0.439
ONT + rTSTM	2.68 ± 0.21	2.61 ± 0.17	2.29 ± 0.524
ONT + cTSTM	2.71 ± 0.23	2.65 ± 0.49	2.38 ± 0.15
Ours	2.802 ± 0.376	2.727 ± 0.437	2.41 ± 0.29

5.3. Ablation Study

Analysis of Regularization Loss. Here we analyze the sensitivity of the regularization loss of the model optimization function in Equation (7). The SNR evaluation results are shown in Table 4, with five values (0, 1, 2, 4, 10) selected for experiments. The results demonstrate that model performance degrades when δ exceeds 2. This is because excessive constraints induce critical information loss, thereby leading to a poorer denoising effect. However, when δ is zero, the model fails to bridge the gap between the audio sub-sampler and ground truth, causing training instability.

Table 4. Ablation of regularization loss weights.

Noise	$\delta = 0$	$\delta = 1$	$\delta = 2$	$\delta = 4$	$\delta = 10$
Env1	2.832	2.961	2.879	2.793	2.524
Env2	2.783	2.818	2.837	2.774	2.328
Env3	2.673	2.727	2.711	2.668	2.192

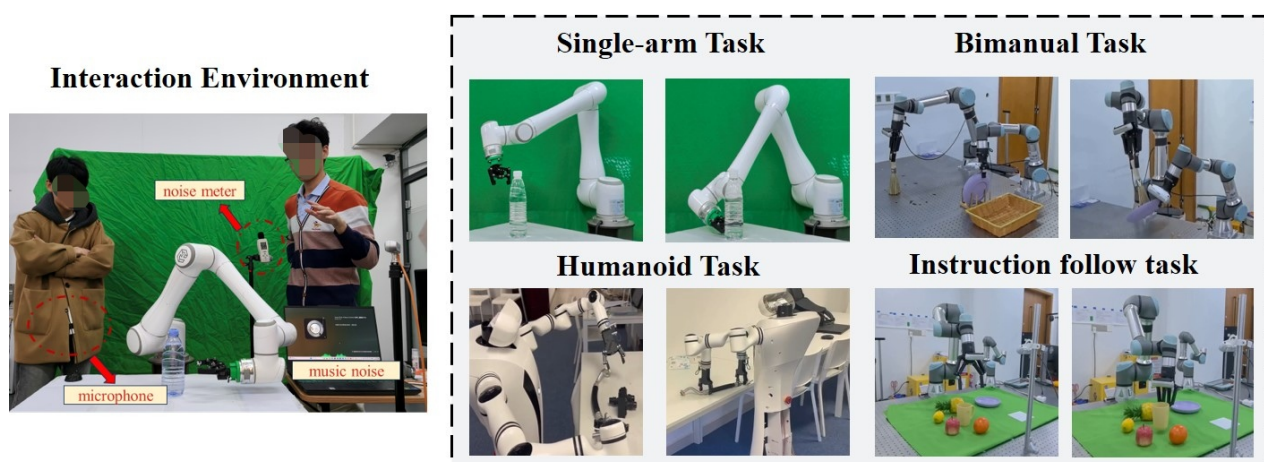
Analysis of Network Module. Next, we analyze the sensitivity of different modules: (a) spectrogram training without separating real and imaginary components (denoted as Net-1), (b) feature extraction using convolutions without the attention module (denoted as Net-2), and (c) feature extraction using only the attention module (denoted as Net-3). Table 5 presents the denoising results, with the full model demonstrating the best performance. Two key reasons are identified: **(1)** Decomposing voice into real and imaginary parts can denoise time–frequency and phase simultaneously, enhancing information extraction from discontinuous audio. **(2)** The convolutional attention module can effectively suppress overlapping noise and extract critical voice features.

Table 5. Ablation of different network modules.

Dataset	Network	SNR	SSNR	PESQ-NB	PESQ-WB	STOI
Env1	Net-1	16.831	3.996	2.655	1.728	0.598
	Net-2	18.274	7.811	2.864	1.921	0.816
	Net-3	17.348	5.921	2.715	1.814	0.726
	Ours	18.671	9.268	2.96	2.166	0.864

5.4. Evaluation in Real World

Hardware Setup. We perform real-world experiments using various robots: the Elite robot arm (7 DOFs), the dual-arm UR robot (12 DOFs), and the Realman humanoid robot (14 DOFs). All robotic arms are equipped with 1 DOF grippers, and external microphones are used for voice interaction, as shown in Figure 7.

**Figure 7.** Robot setup and examples for real-world manipulation tasks. We validate the application of our algorithm on different robotic tasks.

Tasks and Metric. As shown in Figure 7 (right), we designed different tasks to evaluate model performance in real-world noisy scenarios, testing basic actions (e.g., *pull*, *grasp*, *wipe*) through voice interaction. The robots would recognize critical information in noisy interactive voice and execute instructed actions. The action success rate, calculated across multiple experiments, served as the primary metric.

Implementation Details. Figure 7 (left) shows the interactive environment. We designed two experimental conditions to evaluate real-world model performance: (1) computer-simulated urban music environments (<https://pixabay.com/sound-effects/search/city/>, (accessed on 25 August 2025)), and (2) overlapping speech with interfering speakers during voice interaction. The microphone collects interactive voice, and the noise meter measures environmental noise. We consider 50–70 dB as normal noise and above 70 dB as high-frequency noise (<https://www.epa.gov/archive/epa/aboutepa/epa-identifies-noise-levels-affecting-health-and-welfare.html>, (accessed on 25 August 2025)), and select RRnoise [49], DeepFilterNet [50], and ONT [12] as baselines with frozen weights to provide the denoised audio containing action information to the robot.

Result. Table 6 presents the instruction recognition success rates of these strategies, indicating that our model exhibits excellent generalization capabilities across various scenarios and overlapping interactive voices with shorter response times. It achieves notable improvements in noisy and overlapping environments, particularly under 70–80 dB mixed-audio conditions, outperforming DeepFilterNet [50] and ONT [12] by 15% and 7.5% in success rate, respectively. Moreover, it can recognize action information in a discrete noisy voice, as shown in Figure 8. The enhanced performance is attributed to the use of training data augmented with overlapping voices, improving denoising capability in multi-speaker scenarios through accurate modeling of mixed-speech discontinuity.

Table 6. Voice denoising evaluation in real word.

Method	Urban Music (dB)		Multi-Speaker (dB)		Avg Time (s)
	50–70	70–80	50–70	70–80	
RRnoise	0.91	0.87	0.78	0.67	4.2
DeepFilterNet	0.94	0.91	0.82	0.72	1.9
ONT	0.92	0.90	0.84	0.79	3.1
Llama-Omni	0.98	0.97	0.86	0.82	3.7
Ours	0.96	0.94	0.89	0.85	1.2

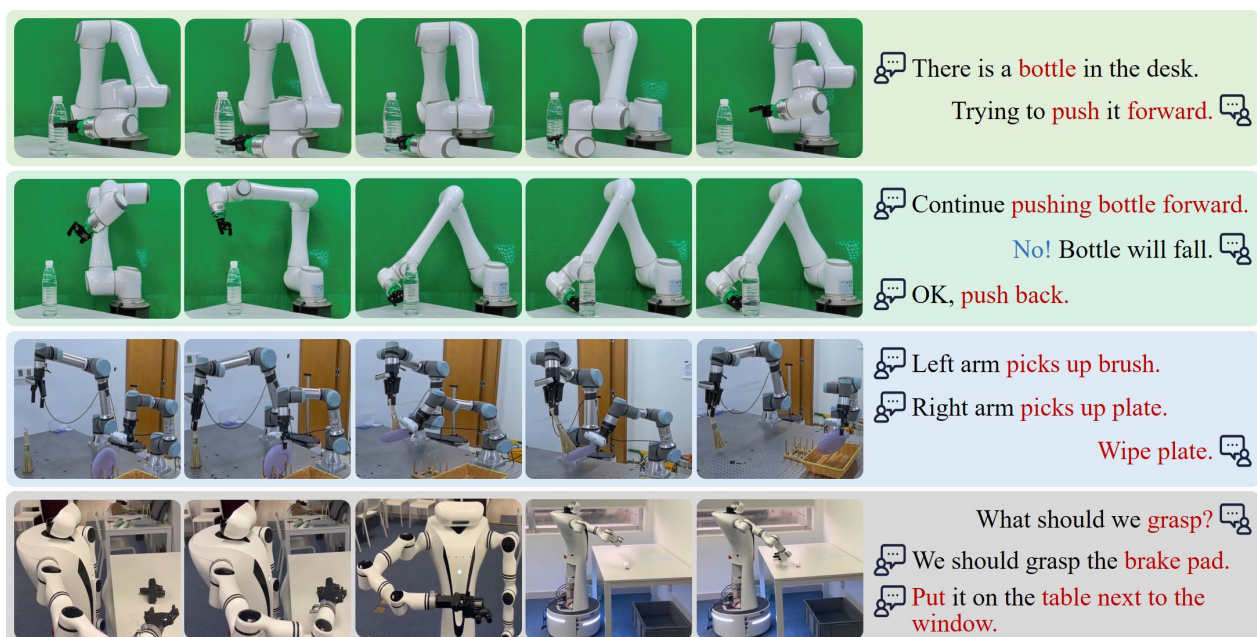


Figure 8. The results of multi-turn voice interaction tasks. Task instructions (red) and termination prompts (blue) are used to guide and adjust robot actions through discrete dialogues.

The proposed model is also compared with state-of-the-art LLMs. The result indicates that Llama-Omni [15] handles ambient noise more effectively in a musical environment, demonstrating stronger robustness. However, its performance declines significantly in a multi-speaker scenario, whereas the proposed method achieves higher recognition success rates under the tested conditions. The performance degradation may occur because the multi-speaker conversational audio in its training data is out of distribution (OOD), making it difficult for the reasoning chain to infer target instructions. Notably, our model contains eight hundred thousand more parameters, which is relatively minor when compared to an 8B-parameter LLM backbone [5], while achieving shorter inference time for audio recognition.

5.5. Terminate Action Test

To verify the feasibility of interruptible interaction, we tested task-action changes through termination prompts during the interaction, as shown in Figure 9. The robot first executes the instruction “Pick the orange beside the apple”. At $t = 7$ s, the “Wait” interrupt stops the action, and the robot returns to the initial pose. When $t = 10$ s, a new instruction “Grasp the yellow cup handle and move to the white mat” initiates a new task. Termination words and instructions allow us to adjust action information, achieving a closed-loop HRI. Our mind is focused on improving the safety of the execution of actions through interrupt words, allowing the model to handle execution actions in *unseen* environments in a controlled manner.

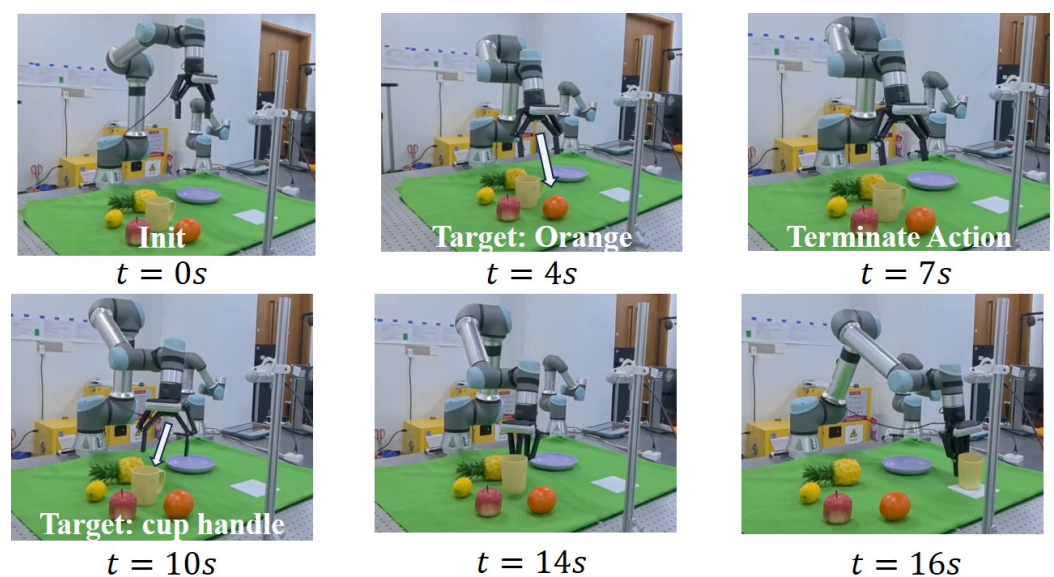


Figure 9. Termination prompt application task. During initiation, the audio command “Pick the orange beside the apple” is transmitted. Upon execution start, the current action is aborted via “wait” ($t = 7$ s) and a new command is provided ($t = 10$ s). The UR arm moves to the new target object, achieving closed-loop voice interaction control.

6. Conclusions

In this paper, we introduce a self-supervised strategy for voice denoising, enabling effective HRI in unseen noisy real-world environments. Specifically, our method generates multi-scenario noisy voice datasets through data synthesis, constructs an effective complex-valued denoising network, and recognizes critical action information from discrete mixed audio. With the rapid advancement of HRI, this study focuses on enhancing recognition efficiency for action commands in noisy environments.

Limitations. Due to hardware limitations, real-world tasks in a laboratory cannot fully replicate outdoor environmental sounds, posing challenges for practical algorithm application. Moreover, convolutional attention networks cannot effectively separate and transfer speech attributes (e.g., content, pitch, timbre) [19], limiting style-preserving denoising. In future research, we will adapt flow matching for discrete speech modeling and deploy it on the kid-sized humanoid robot for outdoor validation.

Author Contributions: Conceptualization, M.L. and W.X.; methodology, M.L. and N.W.; software, M.L.; validation, M.L. and W.X.; writing—original draft preparation, M.L. and W.X.; writing—review and editing, C.Z. and N.W.; supervision, M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the UKRI Guarantee funding for Horizon Europe MSCA Postdoctoral Fellowships under Grant No. EP/Z00117X/1.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Method Overview

Figure A1 illustrates the overall framework of our method, which consists of the following components:

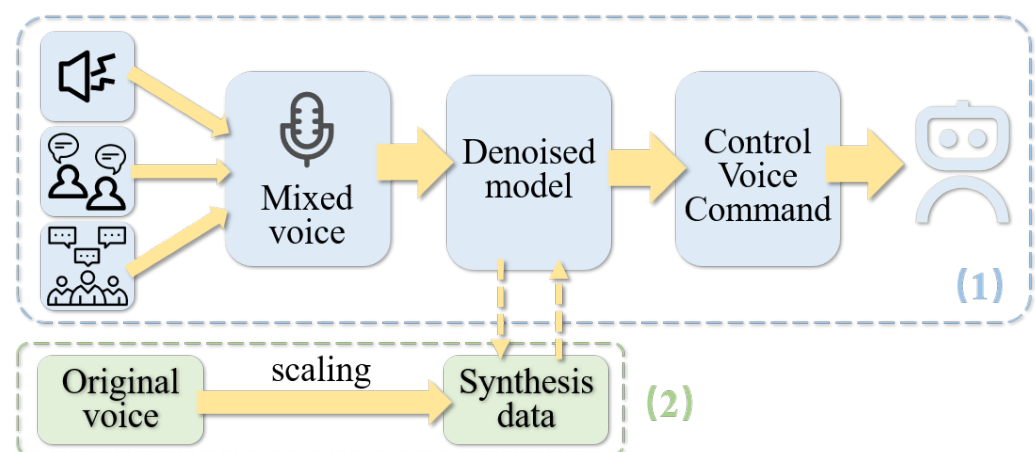


Figure A1. Overview of method framework. (1) shows that in the real-world application, the microphone collects environmental sound, which is processed by the denoising model to extract a control command for robot execution. (2) presents synthetic data to generate model training dataset, enhancing self-supervised denoising performance.

Data Scaling. Through synthetic data techniques, the original voice dataset is scaled to encompass diverse real-world noise scenarios, enhancing denoising robustness. The synthesis process and parameters are detailed in Appendices B and C. A subset of the scaled dataset is open-sourced (<https://tinyurl.com/2ybfz3jt>, (accessed on 25 August 2025)) for community reproducibility.

Denoising Model. We employ a convolutional attention network with self-supervised learning in the audio frequency domain to denoise robotic control commands, eliminating environmental background noise and overlapping speech interference. The self-supervised strategy and model architecture are detailed in Sections 3.2 and 4.3.

Action Execution. The denoised voice commands are processed by a fine-tuned VLA model to execute robotic manipulation tasks. Task completion accuracy serves as the primary metric for evaluating HRI performance. Implementation details and experimental results are documented in Appendix E and Section 4.6.

Appendix B. Implementation Details of Audio Synthesis

The synthetic dataset is constructed by systematically mixing clean speech from the Voice Bank corpus with noise samples from UrbanSound8K (8732 labeled urban sounds across 10 categories, e.g., street music, car horn) and speech samples from TIMIT (630 speakers with diverse phonetic variations). This two-stage mixing process generates (1) a multi-scenario dataset with real-world diverse environmental noise, and (2) a multi-speaker dataset with speech overlap from concurrent speakers. During the audio synthesis process, our model accommodates single-channel audio clips of variable durations as input. To construct the synthetic dataset, we employ the PyDub library. All input audio clips are normalized to a uniform duration, which is determined by the temporal length of the original clean speech samples from the Voice Bank corpus.

Subsequently, all audio clips are resampled to 48 kHz. A key parameter β (adjustable from 0 to 20) is defined to control the signal-to-noise ratio. Higher β (e.g., 20) indicates severe noise degradation, while lower β (e.g., 0) implies minimal noise interference. During audio synthesis, β is randomly selected with systematic variation to generate samples across multiple SNR conditions. This approach scales up the dataset with diverse noise-contaminated scenarios. Finally, these resampled signals are processed through STFT to generate complex-valued spectrograms with a 1024-point FFT, a 64 ms Hamming window, and a 16 ms hop length.

However, we find that when β exceeds 15, the generated multi-speaker dataset exhibits excessive noise levels, adversely affecting the denoising performance of self-supervised models trained on this dataset. Consequently, we perform post-processing on the synthetic data. Detailed procedures are documented in Appendix C.

Appendix C. Implementation Details of Multi-Speaker Dataset Generation

As described in Section 4.1 and Appendix B, audio synthesis is employed to generate multi-speaker scenarios for simulation training. However, the synthetic multi-speaker dataset easily exhibits excessively low SNR, posing challenges to training denoising models and hindering their application in real-world environments. Therefore, Kullback–Leibler Divergence [42] is adopted as the objective function to post-process the synthesized multi-speaker voice, improving the quality of the training data.

We assume that a demixing model $T'(z) = M(z) \cdot A^{-1}(z)$ exists and construct an optimized objective function $D(P_0, P_1)$ using the unmixed time-domain signals t'_i :

$$P_i = \frac{|t'_i(n)|}{\|t'_i\|_1}, \quad (16)$$

$$D(P_0, P_1) = \sum_n [P_0(n) \log\left(\frac{P_0(n)}{P_1(n)}\right) + P_1(n) \log\left(\frac{P_1(n)}{P_0(n)}\right)], \quad (17)$$

where n denotes the number of collected samples. $P_i(n) \geq 0$ for all n and $\sum_{n=0}^{\infty} P_i(n) = 1$ for $i = 0, 1$ (simulate human binaural reception). By minimizing the objective function, a synthetic multi-speaker dataset for audio denoising training is generated, which proves instrumental for real-world HRI applications.

Appendix D. Implementation Details of the Training Strategy for the Methods

We compared different training strategies of the methods. As shown in Table A1, the analysis focuses on the following aspects: (1) whether to train with noisy voice only, (2) whether to apply audio separation, and (3) whether to employ the convolutional multi-

head attention module (CAM). The Voice Bank dataset [40] serves as the clean source, while the training dataset is augmented with synthetic noisy data. All methods are trained for 500 epochs, with results shown in Tables 1–3.

Table A1. Comparison of different training strategies of the methods.

Methods	w/o Clean Voice	with Audio Separation	with CAM
NCT	×	×	×
NNT	✓	×	×
NerNT	✓	×	×
ONT	✓	✓	×
Ours	✓	✓	✓

Appendix E. Real-World Deployment Details

Algorithm 1 illustrates the experimental deployment framework. After the denoising model processes the interaction voice, SpeechRecognition (<https://pypi.org/project/SpeechRecognition/>, (accessed on 25 August 2025)) is employed to convert the audio into text instructions for the robot, serving as the language input for the VLA model. We use a VLA model [3] trained on robot action data (e.g., *pull*, *grasp*, *wipe*) to operate the robot. The robot only moves correctly when the denoised audio captures clear key semantics from the language prompt. Model inference is performed on an NVIDIA RTX 4090 GPU to control UR and Elite robotic arms at 10 Hz. For the Realman humanoid robot, a 384-core NVIDIA Volta GPU with 48 Tensor Cores serves as the action client in the control system, executing tasks at 5 Hz. The denoising network completes inference in 0.4 s, while the VLA model generates actions in about 1.6 s, keeping total processing time under 3 s.

References

- Chi, C.; Xu, Z.; Feng, S.; Cousineau, E.; Du, Y.; Burchfiel, B.; Tedrake, R.; Song, S. Diffusion policy: Visuomotor policy learning via action diffusion. *Int. J. Robot. Res.* **2023**, *42*, 02783649241273668.
- Kim, M.J.; Pertsch, K.; Karamcheti, S.; Xiao, T.; Balakrishna, A.; Nair, S.; Rafailov, R.; Foster, E.; Lam, G.; Sanketi, P.; et al. Openvla: An open-source vision-language-action model. *arXiv* **2024**, arXiv:2406.09246.
- Liu, S.; Wu, L.; Li, B.; Tan, H.; Chen, H.; Wang, Z.; Xu, K.; Su, H.; Zhu, J. Rdt-1b: A diffusion foundation model for bimanual manipulation. *arXiv* **2024**, arXiv:2410.07864. [[CrossRef](#)]
- Intelligence, P.; Black, K.; Brown, N.; Darpinian, J.; Dhabalia, K.; Driess, D.; Esmail, A.; Equi, M.; Finn, C.; Fusai, N.; et al. $\pi 0$. 5: A vision-language-action model with open-world generalization. *arXiv* **2025**, arXiv:2504.16054. [[CrossRef](#)]
- Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. The llama 3 herd of models. *arXiv* **2024**, arXiv:2407.21783. [[CrossRef](#)]
- Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 5485–5551.
- Tao, F.; Busso, C. End-to-end audiovisual speech recognition system with multitask learning. *IEEE Trans. Multimed.* **2020**, *23*, 1–11. [[CrossRef](#)]
- Devis, N.; Demerlé, N.; Nabi, S.; Genova, D.; Esling, P. Continuous descriptor-based control for deep audio synthesis. In Proceedings of the ICASSP 2023—2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.
- Sun, Y.; Yang, L.; Zhu, H.; Hao, J. Funnel Deep Complex U-Net for Phase-Aware Speech Enhancement. In Proceedings of the Interspeech, Brno, Czech Republic, 30 August–3 September 2021; pp. 161–165.
- Fujimura, T.; Koizumi, Y.; Yatabe, K.; Miyazaki, R. Noisy-target training: A training strategy for DNN-based speech enhancement without clean speech. In Proceedings of the 2021 29th European Signal Processing Conference (EUSIPCO), Dublin, Ireland, 23–27 August 2021; pp. 436–440.
- Kashyap, M.M.; Tambwekar, A.; Manohara, K.; Natarajan, S. Speech denoising without clean training data: A noise2noise approach. *arXiv* **2021**, arXiv:2104.03838. [[CrossRef](#)]
- Wu, J.; Li, Q.; Yang, G.; Li, L.; Senhadji, L.; Shu, H. Self-supervised speech denoising using only noisy audio signals. *Speech Commun.* **2023**, *149*, 63–73. [[CrossRef](#)]

13. Li, Y.; Hindriks, K.; Kunneman, F. Single-Channel Robot Ego-Speech Filtering during Human-Robot Interaction. In Proceedings of the 2024 International Symposium on Technological Advances in Human-Robot Interaction, Boulder, CO, USA, 9–10 March 2024; pp. 20–28.
14. Xu, R.; Wu, R.; Ishiwaka, Y.; Vondrick, C.; Zheng, C. Listening to sounds of silence for speech denoising. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 9633–9648.
15. Fang, Q.; Guo, S.; Zhou, Y.; Ma, Z.; Zhang, S.; Feng, Y. LLaMA-Omni: Seamless Speech Interaction with Large Language Models. *arXiv* **2024**, arXiv:2409.06666.
16. Ghose, S.; Prevost, J.J. Autofoley: Artificial synthesis of synchronized sound tracks for silent videos with deep learning. *IEEE Trans. Multimed.* **2020**, *23*, 1895–1907. [[CrossRef](#)]
17. Lin, F.; Hu, Y.; Sheng, P.; Wen, C.; You, J.; Gao, Y. Data scaling laws in imitation learning for robotic manipulation. *arXiv* **2024**, arXiv:2410.18647. [[CrossRef](#)]
18. Lei, Y.; Yang, S.; Wang, X.; Xie, Q.; Yao, J.; Xie, L.; Su, D. Unisyn: An end-to-end unified model for text-to-speech and singing voice synthesis. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 13025–13033.
19. Choi, H.Y.; Lee, S.H.; Lee, S.W. Dddm-vc: Decoupled denoising diffusion models with disentangled representation and prior mixup for verified robust voice conversion. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; Volume 38, pp. 17862–17870.
20. Liu, P.; Glas, D.F.; Kanda, T.; Ishiguro, H. Data-driven HRI: Learning social behaviors by example from human–human interaction. *IEEE Trans. Robot.* **2016**, *32*, 988–1008. [[CrossRef](#)]
21. Liu, Y.; Wang, D. Divide and conquer: A deep CASA approach to talker-independent monaural speaker separation. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *27*, 2092–2102. [[CrossRef](#)] [[PubMed](#)]
22. Baby, D.; Verhulst, S. Sergan: Speech enhancement using relativistic generative adversarial networks with gradient penalty. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 106–110.
23. Rouard, S.; Massa, F.; Défossez, A. Hybrid Transformers for Music Source Separation. In Proceedings of the 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 23), Rhodes Island, Greece, 4–10 June 2023.
24. Fu, S.W.; Yu, C.; Hsieh, T.A.; Plantinga, P.; Ravanelli, M.; Lu, X.; Tsao, Y. Metricgan+: An improved version of metricgan for speech enhancement. *arXiv* **2021**, arXiv:2104.03538.
25. Xie, C.; Wu, Y.C.; Tobing, P.L.; Huang, W.C.; Toda, T. Direct noisy speech modeling for noisy-to-noisy voice conversion. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 6787–6791.
26. Ohlenbusch, M.; Rollwage, C.; Doclo, S. Multi-microphone noise data augmentation for DNN-based own voice reconstruction for hearables in noisy environments. In Proceedings of the ICASSP 2024—2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Republic of Korea, 14–19 April 2024; pp. 416–420.
27. Zinchenko, K.; Wu, C.Y.; Song, K.T. A study on speech recognition control for a surgical robot. *IEEE Trans. Ind. Inform.* **2016**, *13*, 607–615. [[CrossRef](#)]
28. Lai, Y.; Yuan, S.; Nassar, Y.; Fan, M.; Gopal, A.; Yorita, A.; Kubota, N.; Rätsch, M. Natural multimodal fusion-based human–robot interaction: Application with voice and deictic posture via large language model. *IEEE Robot. Autom. Mag.* **2025**, in press. [[CrossRef](#)]
29. Liu, J.; Chen, H.; An, P.; Liu, Z.; Zhang, R.; Gu, C.; Li, X.; Guo, Z.; Chen, S.; Liu, M.; et al. Hybridvla: Collaborative diffusion and autoregression in a unified vision-language-action model. *arXiv* **2025**, arXiv:2503.10631.
30. Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; Dabis, J.; Finn, C.; Gopalakrishnan, K.; Hausman, K.; Herzog, A.; Hsu, J.; et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv* **2022**, arXiv:2212.06817.
31. Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; Chen, X.; Choromanski, K.; Ding, T.; Driess, D.; Dubey, A.; Finn, C.; et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv* **2023**, arXiv:2307.15818.
32. Izquierdo-Badiola, S.; Canal, G.; Rizzo, C.; Aleny, G. Plancollabl: Leveraging large language models for adaptive plan generation in human-robot collaboration. In Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA), Yokohama, Japan, 13–17 May 2024; pp. 17344–17350.
33. Zitkovich, B.; Yu, T.; Xu, S.; Xu, P.; Xiao, T.; Xia, F.; Wu, J.; Wohlhart, P.; Welker, S.; Wahid, S.; et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In Proceedings of the Conference on Robot Learning, Atlanta, GA, USA, 6–9 November 2023; pp. 2165–2183.
34. Nvidia, J.B.; Castaneda, F.; Cherniadev, N.; Da, X.; Ding, R.; Fan, L.; Fang, Y.; Fox, D.; Hu, F.; Huang, S.; et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv* **2025**, arXiv:2503.14734.
35. Lehtinen, J.; Munkberg, J.; Hasselgren, J.; Laine, S.; Karras, T.; Aittala, M.; Aila, T. Noise2Noise: Learning image restoration without clean data. *arXiv* **2018**, arXiv:1803.04189. [[CrossRef](#)]

36. Pang, T.; Zheng, H.; Quan, Y.; Ji, H. Recorrupted-to-recorrupted: Unsupervised deep learning for image denoising. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 2043–2052.
37. Huang, T.; Li, S.; Jia, X.; Lu, H.; Liu, J. Neighbor2neighbor: Self-supervised denoising from single noisy images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14781–14790.
38. Zhou, L.; Gao, Y.; Wang, Z.; Li, J.; Zhang, W. Complex spectral mapping with attention based convolution recurrent neural network for speech enhancement. *arXiv* **2021**, arXiv:2104.05267. [\[CrossRef\]](#)
39. Salamon, J.; Jacoby, C.; Bello, J.P. A dataset and taxonomy for urban sound research. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 1041–1044.
40. Veaux, C.; Yamagishi, J.; King, S. The voice bank corpus: Design, collection and data analysis of a large regional accent speech database. In Proceedings of the 2013 International Conference Oriental COCOSDA Held Jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE), Gurgaon, India, 25–27 November 2013; pp. 1–4.
41. Garofolo, J.S.; Lamel, L.F.; Fisher, W.M.; Pallett, D.S.; Dahlgren, N.L.; Zue, V.; Fiscus, J.G. *TIMIT Acoustic-Phonetic Continuous Speech Corpus*; National Institute of Standards and Technology (NIST): Gaithersburg, MD, USA, 1993. Available online: <https://catalog.ldc.upenn.edu/LDC93s1> (accessed on 25 August 2025).
42. Golokolenko, O.; Schuller, G. A fast stereo audio source separation for moving sources. In Proceedings of the 2019 53rd Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 3–6 November 2019; pp. 1931–1935.
43. Bartle, R.G. *The Elements of Integration and Lebesgue Measure*; John Wiley & Sons: Hoboken, NJ, USA, 2014.
44. Tan, K.; Wang, D. Complex spectral mapping with a convolutional recurrent network for monaural speech enhancement. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 6865–6869.
45. Pandey, A.; Wang, D. Densely connected neural network with dilated convolutions for real-time speech enhancement in the time domain. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6629–6633.
46. Rix, A.W.; Beerends, J.G.; Hollier, M.P.; Hekstra, A.P. Perceptual evaluation of speech quality (PESQ)—A new method for speech quality assessment of telephone networks and codecs. In Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (Cat. No. 01CH37221), Salt Lake City, UT, USA, 7–11 May 2001; pp. 749–752.
47. Hu, Y.; Loizou, P.C. Evaluation of objective quality measures for speech enhancement. *IEEE Trans. Audio Speech Lang. Process.* **2007**, *16*, 229–238. [\[CrossRef\]](#)
48. Xu, Z.; Strake, M.; Fingscheidt, T. Deep noise suppression maximizing non-differentiable PESQ mediated by a non-intrusive PESQNet. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2022**, *30*, 1572–1585. [\[CrossRef\]](#)
49. Valin, J.M. A hybrid DSP/deep learning approach to real-time full-band speech enhancement. In Proceedings of the 2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP), Vancouver, BC, Canada, 29–31 August 2018; pp. 1–5.
50. Schroter, H.; Escalante-B, A.N.; Rosenkranz, T.; Maier, A. DeepFilterNet: A low complexity speech enhancement framework for full-band audio based on deep filtering. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 7407–7411.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.