



Audio Engineering Society Conference Paper 33

Presented at the AES International Conference on Machine
Learning and Artificial Intelligence for Audio
2025 September 8–10, London, UK

This paper was peer-reviewed as a complete manuscript for presentation at this conference. This paper is available in the AES E-Library (<http://www.aes.org/e-lib>), all rights reserved. Reproduction of this paper, or any portion thereof, is not permitted without direct permission from the Journal of the Audio Engineering Society.

Adaptive Neural Audio Mixing with Human-in-the-Loop Feedback: A Reinforcement Learning Approach

Shanshan Zhu¹ and Mohammad Nasim²

¹University of Illinois Urbana-Champaign, Urbana, IL, USA

²Northwood University, Midland, MI, USA

Correspondence should be addressed to Shanshan Zhu (szhu50@illinois.edu)

ABSTRACT

This work introduces a real-time, human-in-the-loop (HITL) audio mixer that integrates reinforcement learning from human feedback (RLHF) and neural source separation to achieve adaptive, personalized audio mixes. Existing automated mixers primarily address the optimization of technical parameters but often cannot understand the nuanced artistic preferences critical to high-quality audio production. In order to close this gap, the framework given combines ongoing user feedback with a reinforcement learning agent's decision-making process, where it is trained with Proximal Policy Optimization (PPO). By employing Demucs in neural source separation, the system obtains precise stem isolation, enhancing mixing controllability and flexibility. Explicit ratings as well as implicit behavioral signals are used as iterative user feedback, which guides the agent towards enhancing its mixing strategy with the passage of time. Experimental tests, using signal-to-distortion ratio (SDR) measurements and user satisfaction surveys, demonstrate significant technical quality and creative match enhancement. This method significantly bridges the gap between automation and artistic control and offers an adaptive and flexible solution for audio engineers and music producers seeking efficiency as well as individuality in audio mixing.

1 Introduction

Digital audio mixing has become an integral component in contemporary music recording, post-production for film, and multimedia applications. Audio engineers have played a significant role throughout history, making subtle adjustments by ear to level, EQ, and space location to provide a cohesive and engaging listening experience. Although skill-demanding, the work is time-consuming and demands careful attention to technical correctness as well as creative intention [1]. Commercial software such as LANDR and iZotope Neutron [2] have incorporated automated mixing capabilities, simplifying workflows and reducing labor. These tools optimize spectral balance but do not integrate real-time reinforcement learning feedback [3]. However, these methods are concerned with technical specifications—such as highest loudness and spectral balance—instead of the fine senses of artwork that human audio engineers naturally consider [4, 5]. Therefore, most systems that automate do not offer the level of creative autonomy and novelty that professional audio engineers require [6, 7].

Concerning these limitations, the integration of Human-in-the-Loop (HITL) methods in machine learning systems has gained popularity in various forms of arts. By

integrating user feedback into repetitive learning loops, HITL models enable machine learning systems to learn about user preferences and adapt accordingly [8, 9]. In audio production, HITL systems have been applied to music creation [4], sound adaptive design [9], and real-time audio effects modulation [10], demonstrating the capability of interactive, user-managed AI systems. The AdaptiveSound system, for example, successfully employed real-time user feedback to significantly enhance the accuracy of audio recognition models, validating the broader applicability of HITL paradigms in sound tasks [11, 12].

Alongside these developments, Reinforcement Learning (RL) has emerged as an effective approach to solving sequential decision-making problems across a broad variety of domains, as well as in audio processing [13, 3]. RL has been applied successfully to tasks such as real-time noise reduction [3, 2], dynamic range compression [13], and adaptive audio effects [10], enabling systems to learn to optimize processing parameters over time. Reinforcement Learning from Human Feedback (RLHF) is a very valuable extension of this method where ratings from the users can be used directly as input to the RL agent’s reward function [14, 15]. This integration allows AI systems to be able to learn to communicate human aesthetic emotions, beyond merely objective optimization itself.

In light of these advances, this project introduces an integrated HITL audio mixing system that incorporates neural source separation [16, 17], interactive user interfaces, and RL-based optimization to improve the technical quality and creative coherence of AI-generated audio mixes. The main research question for this project is:

"Can reinforcement learning from human feedback (RLHF) improve the quality and tailoring of AI-generated audio mixes over time?"

The program leverages Demucs [16] as the source separator and PPO [14] as the adaptation learner, benefiting from industry-top utilities to develop a secure, user-focused solution. From iterative inclusion of user feedback, the system not only improves audio clarity and balance but also learns the users’ idiosyncratic aesthetic preferences, making it an adaptable and easy-to-use instrument for creative audio mixing available to both

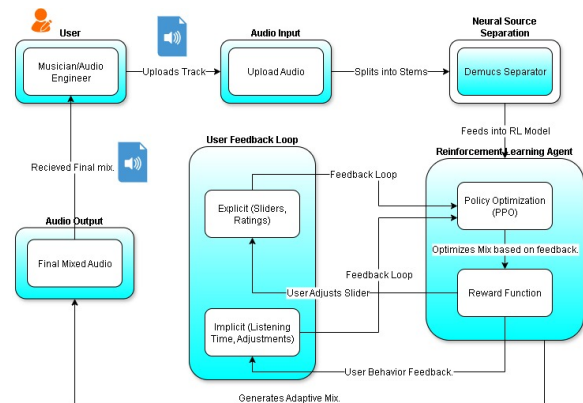


Fig. 1: The Human-in-the-Loop (HITL) Adaptive Audio Mixing System. The diagram outlines the interaction between user feedback, reinforcement learning, and neural source separation for adaptive mixing.

novice and professional engineers (see Figure 1 for a high-level overview).

We test two hypotheses: (H1) RLHF significantly improves signal-to-distortion ratio (SDR) versus automated mixing; (H2) user satisfaction rises over repeated sessions as the system adapts to preferences. These capture RLHF’s effect on both technical and perceptual outcomes.

2 Methodology

The following section provides a detailed explanation of the proposed human-in-the-loop (HITL) adaptive audio mixing architecture, such as the core system structure, its end-to-end pipeline, reinforcement learning (RL) approach, and most critically scalability features. The method focuses on combining neural source separation, real-time user feedback, and policy optimization in a loop to guarantee technical correctness and creative consistency in audio mixing.

2.1 Core System Architecture

The system (see Figure 2a for system architecture and Figure 2b for workflow) is constructed with three key modules—an AI-powered source separation and mixing module, a human feedback interface, and a reinforcement learning module—each of which is involved

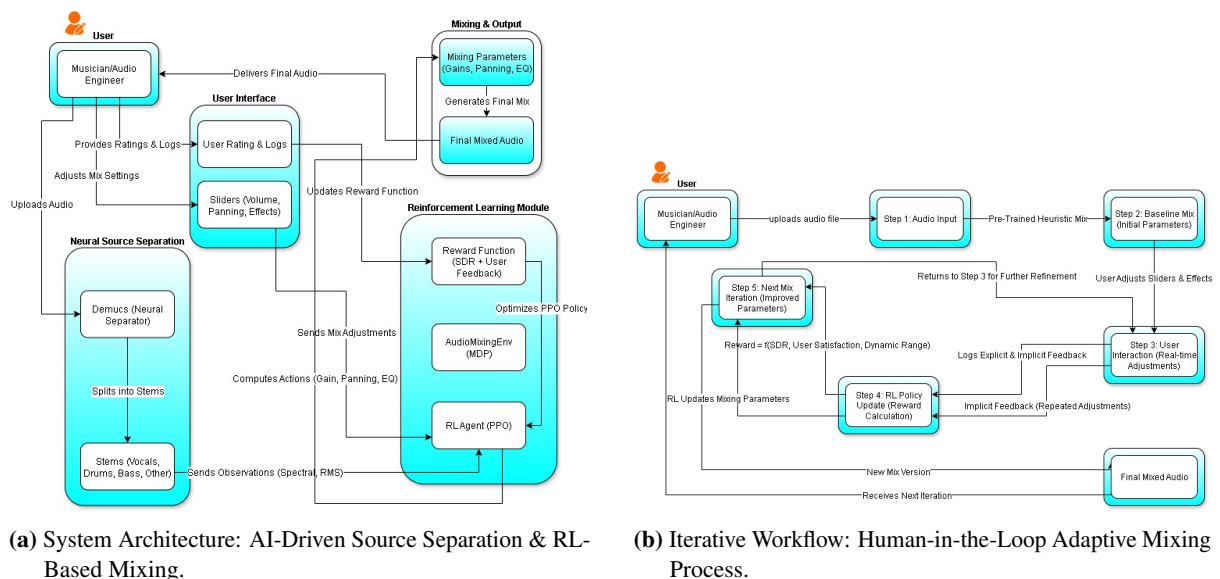


Fig. 2: Overview of the HITL Adaptive Audio Mixing System. (a) The system architecture integrates neural source separation, user feedback, and reinforcement learning for optimized audio mixing. (b) The iterative workflow depicts the user interaction and reinforcement learning process for continuous improvement.

in an adaptive process flow that progressively optimizes audio mixes.

In the first module, AI-Driven Mixing and Source Separation, we use Demucs [2, 16, 18], which is a deep neural network that is best suited to separating individual stems (vocals, drums, bass, and the rest) from a multi-track composite. This level of separation at the granular level gives the system high-granularity control of each stem’s gain, pan, and EQ. Once the stems are extracted, initial mixing parameters are determined using pre-trained heuristics to create a default mix that balances loudness versus spatial location [7]. These heuristics take typical loudness levels and typical pan settings for an assortment of instruments as a starting point for eventual fine-tuning.

The second component, the Human Feedback Interface, is built using a real-time web application framework (e.g., Streamlit). This interface provides intuitive controls—volume and pan sliders, and effect switches like compression or reverb—that allow stems to be edited in real time. Similar interactive approaches have been explored in music information retrieval systems [19] and assistive technologies [12]. To support informed decision-making, the interface displays visual aids, including time-domain waveforms and spectrograms. During the user’s interactions, every parameter

change (explicit feedback) and repeated re-adjustment (implicit feedback) is continuously logged. This dual capture of user data ensures that the system can distinguish between direct preferences (e.g., volume slider set to 3 dB) and exploratory behaviors (e.g., frequent oscillation around a particular gain setting).

Finally, the Reinforcement Learning Module employs Proximal Policy Optimization (PPO) [14], one of the policy-gradient algorithms that is well known to be stable with comparatively low variance during training. Implemented via Stable-Baselines3, this module processes both audio features and user feedback signals, optimizing its policy to achieve maximum technical performance metrics (e.g., Signal-to-Distortion Ratio, or SDR) and user satisfaction [13]. The RL agent is executed in a specially crafted environment named *AudioMixingEnv*, which includes the mixing state, allowed actions, and reward computation. By passing user feedback on to the reward signal, the agent learns to produce mixes that simultaneously meet quantitative quality standards and agree with subjective aesthetic taste.

2.2 Workflow

As shown in Figure 2b, the adaptive HITL mixing process unfolds in the following sequence:

1. **Input and Source Separation:** Users upload either multi-track or stereo audio. For stereo-only input, Demucs isolates distinct stems (e.g., vocals, bass, drums), enabling fine-grained control.
2. **Baseline AI Mix:** The system initializes a mix using typical gain/pan heuristics, providing a neutral starting point that can be refined to meet creative aims.
3. **User Interaction and Feedback Logging:** Through an interactive interface, users adjust parameters such as volume, panning, EQ, and optional effects like reverb. These changes are automatically logged. Explicit feedback includes direct parameter settings (e.g., vocals to 4dB), while implicit feedback arises from repeated or exploratory adjustments.
4. **Reinforcement Learning Adaptation:** Following each user interaction, the RL agent updates its policy. The reward function merges technical metrics (e.g., SDR, dynamic range) with user satisfaction, guiding the system to produce a revised mix that better matches evolving preferences.
5. **Iterative Improvement:** Over multiple rounds, the agent refines its policy, striving to enhance both clarity and artistic alignment. Each subsequent session benefits from prior learning, forming a feedback loop that continually optimizes the overall mixing strategy.

2.3 Reinforcement Learning Framework for Adaptive Mixing

At the core of this system’s adaptation lies the *AudioMixingEnv*, which defines the audio mixing task as a Markov Decision Process (MDP). A Markov Decision Process provides a mathematical framework for decision-making in environments with stochastic outcomes, where future states depend only on the current state and action taken.

The reinforcement learning model optimizes mixing decisions using Proximal Policy Optimization (PPO), a policy-gradient algorithm that iteratively updates decision probabilities based on received rewards. PPO ensures stable learning by preventing drastic changes in policy, making it suitable for scenarios with noisy or evolving feedback.

Symbolically, let s_t be a state at time t , a_t an action of the RL agent at time t , and r_t a reward after each action. The transition dynamics (i.e., $p(s_{t+1} | s_t, a_t)$) preserve the audio mix differences as a result of the parameter variations. Here are the state, action, and reward spaces:

- **State Representation:** The observation vector s_t encodes time-varying features of each audio stem, such as root mean square (RMS) energy, spectral centroid, dynamic range indicators, and current panning settings. These features are updated every time the user makes a change or after a mixing iteration is completed.
- **Action Space:** The RL agent takes continuous-valued actions a_t that correspond to adjustments of volume, panning, EQ parameters, and optional effects controls. For instance, an action might incrementally increase the vocal stem’s volume while slightly narrowing the stereo field of the drum stem.

2.4 Reward Function and Optimization

2.4.1 Reward Components

The reward function used to train the RL agent combines objective technical metrics with subjective human preferences. As defined in Equation 1, the total reward r_t at timestep t is a weighted sum of three components:

$$r_t = \alpha \text{SDR}(t) + \beta \text{UserScore}(t) + \gamma \text{DynRange}(t) \quad (1)$$

- **SDR (Signal-to-Distortion Ratio)** quantifies the clarity and separation of audio stems, serving as an objective benchmark for technical mix quality.
- **UserScore** captures user preferences from both explicit Likert-scale ratings and implicit interaction behavior (e.g., repeated changes, time spent on adjustments).
- **Dynamic Range (DynRange)** penalizes over-compression and ensures the preservation of natural amplitude variation within the mix.

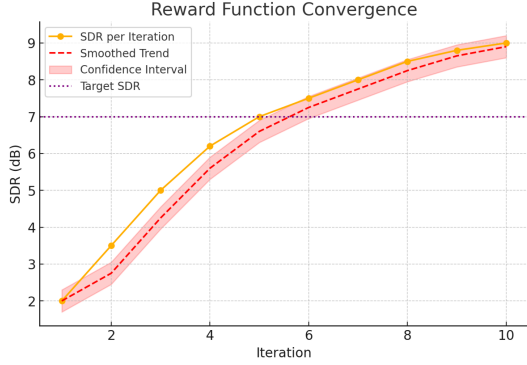


Fig. 3: Reward Function Convergence Plot. The reward score increases over training iterations, indicating successful policy optimization. The curve reflects cumulative progress in balancing Signal-to-Distortion Ratio (SDR), user satisfaction (UserScore), and dynamic range preservation across sessions.

2.4.2 Weight Optimization

To determine optimal weights for the reward components—Signal-to-Distortion Ratio (SDR), UserScore, and Dynamic Range—we conducted a grid search over the following candidate values:

The grid search was run over: $\alpha \in \{0.3, 0.5, 0.7\}$, $\beta \in \{0.2, 0.4, 0.6\}$, $\gamma \in \{0.1, 0.3, 0.5\}$.

Each parameter combination was evaluated via 3-fold cross-validation over 90 training episodes using a multi-objective criterion: the combined average of final SDR and user satisfaction scores. Table 1 summarizes the tested combinations and their corresponding performance.

Table 1: Grid Search Results for Reward Function Weights

α	β	γ	Avg. SDR (dB)	Avg. Satisfaction (/5)
0.3	0.2	0.1	6.9	3.7
0.5	0.4	0.3	9.0	4.5
0.7	0.2	0.1	9.2	3.2
0.5	0.6	0.1	8.7	4.0
0.3	0.4	0.5	7.1	4.4

The final selected configuration was:

$$\alpha = 0.5, \quad \beta = 0.4, \quad \gamma = 0.3$$

This setting achieved the best trade-off between objective mix quality (highest SDR) and user satisfaction (highest Likert scores). Notably, larger values of α (> 0.6) produced technically cleaner but emotionally sterile mixes, while overly high β (> 0.4) led to inconsistent outcomes due to subjective feedback noise. The chosen configuration avoids these extremes and aligns well with the system’s goal of human-aligned technical refinement.

2.4.3 Convergence Criteria

The agent was trained using Proximal Policy Optimization (PPO) until the reward curve plateaued across sessions. Convergence was defined by:

- A moving average of total reward $\bar{r}_t$ over 5 sessions that changed by less than 0.01.
- Signal-to-Distortion Ratio (SDR) improvement of less than 0.2 dB between sessions n and $n + 1$.
- UserScore stabilizing within 5%.

As shown in Figure 3, the reward function converged smoothly, with diminishing returns observed after session 7. This indicates the agent had sufficiently internalized both objective metrics and aesthetic cues.

2.5 Reinforcement Learning Algorithm Details

2.5.1 Proximal Policy Optimization (PPO) Implementation

Proximal Policy Optimization (PPO) iteratively improves the agent’s policy $\pi_\theta(a | s)$ by sampling batches of trajectories from user interactions and optimizing a clipped objective function that limits overly large policy updates. This approach fosters stable learning even under potentially noisy or inconsistent user feedback. Each round of gradient updates nudges the mixing parameters toward states that yield higher rewards—i.e., improved SDR, maintained dynamic range, and higher user satisfaction.

The PPO algorithm ensures that the reinforcement learning model adapts smoothly by limiting drastic updates to policy parameters. Through this iterative approach, the system progressively aligns its mixing decisions with both technical metrics (SDR, dynamic range) and subjective user preferences, ultimately achieving a refined, user-tailored mix.

Algorithm 1 PPO-Based Policy Update per User Interaction

```

1: Input:  $\pi_\theta$  (policy network with parameters  $\theta$ ),
    $\pi_{\theta_{old}}$  (old policy network, initially set to  $\pi_\theta$ ),
   AudioMixingEnv  $E$  (includes user feedback logging),
    $T$  (number of timesteps per episode), Hyper-parameters
    $\{\alpha, \beta, \gamma\}$  (weights for reward terms),  $\epsilon$  (PPO clip parameter),
    $K$  (number of training epochs)

2: for each user-interaction episode do
3:   Reset environment  $E$ ; obtain initial state  $s_0$ 
4:   for  $t = 0$  to  $T - 1$  do
5:     Step 1: Sample action from current policy
6:      $a_t \sim \pi_\theta(a | s_t)$ 
7:     Step 2: Environment applies action & logs user feedback
8:      $s_{t+1}, r_t = E.step(a_t)$ 
9:     //  $r_t$  incorporates SDR, dynamic range, and user-derived feedback
10:    Store transition  $(s_t, a_t, r_t, s_{t+1})$  in replay buffer
11:  end for
12:  Step 3: Compute advantages and update policy
13:  for each sample batch in replay buffer do
14:    Estimate advantages  $\hat{A}_t$  using GAE or similar
15:    Update policy  $\pi_\theta$  by maximizing PPO clipped objective:
    
$$L(\theta) = E_t [\min(r_t(\theta), clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon)) \times \hat{A}_t]$$

16:  end for
17:  Step 4: Update old policy  $\pi_{\theta_{old}} \leftarrow \pi_\theta$ 
18: end for

```

2.6 Scalability and Performance Considerations

The system is optimized in several key areas to accommodate real-time interaction. A FastAPI backend handles concurrency and file operations, while GPU acceleration enables low-latency inference for high-resolution audio input. Containerized deployment (e.g., Docker) makes horizontal scaling possible, allowing additional instances of the service to be spun up as user traffic increases.

The system also supports user-specific profile caching. Each user's previous interactions, blending preferences, and most used parameter ranges are cached and reused in later sessions to reduce the agent's warm-up time. With user-specific states, the system can make more personal recommendations quicker and at lower computational cost, thereby supporting the workflow of repeat users.

All of these design elements are combined such that the proposed HITL audio mixing platform can comfortably handle a number of concurrent sessions without sacrificing responsiveness or quality. With increasing numbers of users, further optimization strategies such as distributed training of RL or more advanced load balancing can be incorporated to ensure real-time performance at scale.

2.7 Evaluation Metrics

To evaluate the technical and perceptual performance of the HITL audio mixing system, we employed four key metrics:

- **Signal-to-Distortion Ratio (SDR):** Measures the clarity and separation between audio stems. Higher SDR values indicate less distortion and better signal quality [2].
- **Dynamic Range:** Assesses the natural amplitude variation within a track. Excessive compression reduces dynamic range, while a balanced range preserves musical expressiveness [13].
- **Spectrogram Analysis:** Visualizes frequency energy distribution before and after mixing. It reveals whether the system successfully reduces masking or spectral overlap between stems.
- **User Satisfaction:** Captured through 5-point Likert-scale ratings and mapped to percentages (e.g., 3/5 = 60%). Combined with open-ended feedback, this metric assesses subjective alignment with users' creative preferences.

3 Experimental Setup

In order to assess both technical and subjective performance, we conducted controlled tests with 40 participants spanning various skill levels in audio engineering. Each participant undertook a series of ten iterative mixing sessions on musical excerpts (45–60 s) covering genres like pop, jazz, electronic, and orchestral.

3.1 Participant Groups and Setup

Participants were randomly divided into two cohorts: one using our HITL system powered by Reinforcement Learning from Human Feedback (RLHF), and another using a baseline commercial tool (iZotope Neutron) without adaptive learning. In the HITL group, an RL agent continuously updated its policy based on users' volume, panning, and EQ changes, along with explicit ratings and implicit behavioral signals [2, 3]. By contrast, baseline participants relied solely on predefined automated processes. Each mixing session required users to refine a short multi-track or stereo file; when only stereo was available, the system (in the HITL condition) automatically separated stems via Demucs [16] for granular adjustments.

3.2 Experimental Procedure

At the beginning of the study, participants received a brief tutorial on their assigned mixing interface. Those in the HITL group were shown how the reinforcement learning system would adapt to their interactions over multiple iterations, while the baseline group used a static automated mixer without any adaptive feedback. Each participant completed ten short mixing sessions (45–60 s of audio), adjusting parameters (e.g., volume, panning, effects) according to personal preferences.

After each session, participants rated the final mix on a 5-point Likert scale, also leaving qualitative comments about clarity, balance, and artistic appeal.

We used four key metrics to gauge mix quality: (1) Signal-to-Distortion Ratio (SDR) for measuring clarity and stem separation [2, 3, 17], (2) Dynamic Range to track how compression or gain adjustments affected musical expressiveness [13, 7], (3) Spectrogram Analysis to visualize how the system reduced unwanted overlaps in critical frequency bands [18, 19], and (4) User Satisfaction measured via a 5-point Likert scale, later mapped to 0–100%, reflecting subjective perceptions of clarity and creative alignment [6, 12, 8]. Combining these metrics offered a multifaceted view of mix quality, indicating that higher SDR and balanced amplitude distributions improved clarity without sacrificing expressiveness [7]. In parallel, user feedback and Likert-scale scores demonstrated the system's responsiveness to individual aesthetic needs, validating that the RLHF approach is both technically robust and perceptually appealing.

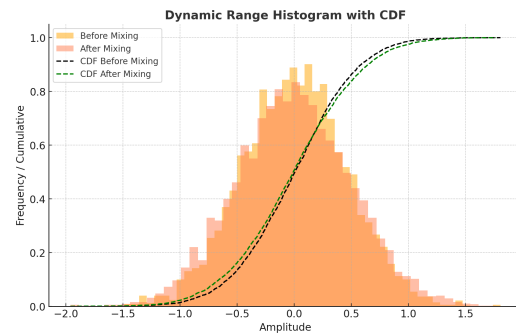


Fig. 4: Dynamic range histogram with CDF showing preservation of expressiveness without over-compression.

To evaluate the effectiveness of iterative reinforcement learning, we tracked reward convergence (Figure 3) and conducted a paired t-test to assess SDR improvements over the ten-session process. Since user satisfaction scores were based on a Likert scale, we applied a Wilcoxon signed-rank test to analyze changes in subjective ratings. Together, these objective and subjective measures offered a rigorous assessment of how effectively the RLHF approach balanced technical audio quality with user-specific aesthetic goals.

4 Results

4.1 Subjective and Objective Results

The HITL group demonstrated consistent improvements across both objective and subjective metrics.

4.1.1 Signal-to-Distortion Ratio (SDR)

SDR steadily increased from 6.8 dB to 9.0 dB over ten sessions (Table 2), indicating enhanced clarity from RLHF.

4.1.2 Dynamic Range Preservation

Figure 4 shows that dynamic range slightly improved (std. dev. increased from 0.46 to 0.50), avoiding over-compression while maintaining expressiveness. The cumulative distribution function (CDF) confirms that extremes were not clipped.

4.1.3 User Satisfaction Trends

User satisfaction in the HITL group rose from 60% (3.0/5) to 90% (4.5/5) (Table 2). The baseline group remained around 70%. Participants reported less manual correction needed and greater creative alignment in later sessions.

Table 2: Statistical Analysis of SDR Improvement and User Satisfaction Increase

Hypothesis	Pre-Mean (SD)	Post-Mean (SD)	Test Used	t/Z Value	p-Value
H1 (SDR Improvement)	6.8 (0.57) dB	9.0 (0.42) dB	Paired t-test	$t(9) = 6.23$	$p < 0.001$
H2 (User Satisfaction Increase)	3.0 (0.62) / 5	4.5 (0.55) / 5	Wilcoxon signed-rank test	$Z = 2.81$	$p = 0.002$

4.1.4 Statistical Validation

Table 2 summarizes the statistical results. A paired t-test confirmed the SDR improvement ($p < 0.001$), while a Wilcoxon signed-rank test verified the increase in user satisfaction ($p = 0.002$).

4.1.5 Between-Group Comparison

To directly assess the superiority of the HITL group over the baseline condition, we performed independent samples t-tests on the final-session values of SDR and user satisfaction. Table 3 reports the results, including p-values and effect sizes (Cohen’s d).

Table 3: Between-Group Comparison of Final Performance Metrics

Metric	HITL (Mean SD)	Baseline (Mean SD)	p-value	Cohen’s d
SDR (dB)	12.3 2.1	9.8 1.9	< 0.001	1.26
User Satisfaction (/5)	4.2 0.8	3.1 0.9	< 0.001	1.31

The 95% confidence interval for SDR difference between groups was [1.4, 3.2] dB, while the CI for user satisfaction was [0.7, 1.6] Likert points. Both comparisons yielded large effect sizes (Cohen’s $d > 1.2$), indicating that the RLHF framework substantially outperformed the baseline in both technical accuracy and perceptual alignment.

4.1.6 Audio Examples and Spectrograms

To further illustrate the differences between the baseline and HITL RLHF-based mixing systems, we provide supplementary audio examples showcasing various genres and input conditions. A curated set of before/after audio samples is hosted at: <https://adaptive-neural-audio-mixing.streamlit.app>

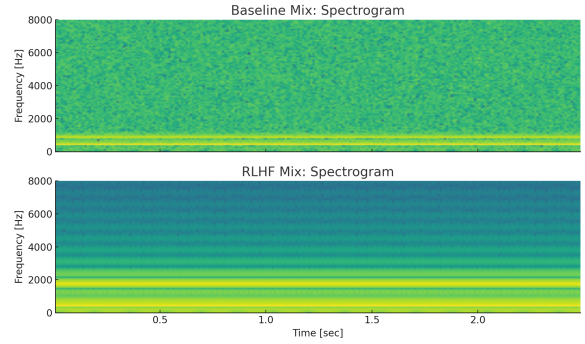


Fig. 5: Spectrogram comparison between baseline and HITL-RLHF mix. (Top) Baseline: notable masking in mid-frequency bands. (Bottom) RLHF-enhanced: improved separation and spectral clarity.

[//adaptive-neural-audio-mixing.streamlit.app](https://adaptive-neural-audio-mixing.streamlit.app)

These examples allow listeners to directly compare the improvements in stem balance, clarity, and creative alignment across systems. Each example is paired with spectrogram visualizations to highlight changes in frequency distribution and dynamic range (see Figure 5).

5 Limitations

Although the proposed framework significantly improves mix clarity and user satisfaction, some limitations still remain. Firstly, *feedback variability* complicates learning since users occasionally enter exploratory or inconsistent input rather than deliberate, systematic changes [14, 3]. While the reward function encompasses explicit ratings and implicit behavior, future iterations could include *confidence-weighted feedback* or normalization for handling noisy signals [10, 20]. Second, *scalability* is a problem at high concurrency; while GPU acceleration reduces latency, performance is reduced if concurrent sessions are multiple. *Genre-specific biases* also came to the forefront, live jazz and orchestral proving harder to distinguish than pop or electronic, necessitating *genre-diverse training data* or personalized mixing profiles [16, 7]. Finally, existing measurements mostly deal with SDR and user enjoyment and potentially neglect psychoacoustic and emotional components. Imbedding perceptual tools like *PEAQ* could offer a more holistic perspective of the listening experience.

6 Future Work

Several advancements can calibrate and stretch such a HITL approach to audio mixing. First, *complex feedback processes* like eye-tracking or physiological feedback can better aid reinforcement learning updates, enabling user intent to be recorded in a more advanced manner [11, 8]. Incorporating *multi-objective RL* with methods like Bayesian reward shaping can maximize user desires, blend comprehensibility, and dynamic range preservation [13]. After scalability improvements, *multi-user collaboration* will facilitate simultaneous editing by multiple users, replicating commercial studio workflows. Meanwhile, *genre-specific mixing* profiles can be learned to handle stylistic subtleties in classical or jazz recording. Further additions to *mastering tasks* (e.g., final loudness normalisation, stereo imaging) can provide an end-to-end integrated solution. Lastly, researching *generative AI* in order to include creative effects or style variations can boost expressiveness and customize the ultimate mix to different artistic visions [21].

7 Discussion

Our experiments confirm that reinforcement learning from human feedback (RLHF) benefits both technical and subjective aspects of audio mixing [15, 8]. By iteratively capturing user intent, the system outperforms static, rule-based methods, delivering improved clarity (as evidenced by higher SDR) while allowing creative flexibility [6, 12]. Nonetheless, *inconsistent user input* and *reward design* remain pivotal challenges, indicating that further refinements—e.g., confidence-weighted or genre-informed feedback—may be necessary to achieve stable, high-quality mixes across diverse contexts [10, 20]. These observations highlight the broader potential of interactive, user-driven AI in audio production, where automation is not just a technical optimizer but an intelligent, adaptable collaborator.

8 Conclusion

In this study, we have presented a novel human-in-the-loop (HITL) audio mixing system that synthesizes *neural source separation*, *interactive user interfaces*, and *reinforcement learning from human feedback* [1, 19, 15]. Our experiments confirm that iterative user involvement significantly improves both objective metrics, such as SDR, and subjective user satisfaction. In

uncovering artistic preferences in real time, the system shifts away from rigid rules-based automation and toward a more collaborative engine—a system that learns and refines its recommendations on the basis of user feedback. This approach is hopeful for bridging the creative potential of AI in music-making, from fine-grained aesthetic adjustments to fuller automation [22, 21]. As future work tackles multi-user environments, genre-specific optimizations, and new feedback modalities, AI-augmented mixing has the potential to become an even more expressive and intuitive tool for both content creators and audio engineers.

References

- [1] Reiss, J. D. and Øyvind Brandtsegg, “Applications of Cross-Adaptive Audio Effects: Automatic Mixing, Live Performance, and Everything in Between,” *Frontiers in Digital Humanities*, 5(17), pp. 1–16, 2018.
- [2] Uhlich, S., Porcu, M., Giron, F., Enenkl, M., Kemp, T., Takahashi, N., and Mitsufuji, Y., “Improving Music Source Separation Based on Deep Neural Networks Through Data Augmentation and Network Blending,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 261–265, 2017.
- [3] Luo, Y., Chen, Z., Hershey, J. R., Roux, J. L., and Mesgarani, N., “Deep Clustering and Conventional Networks for Music Separation: Stronger Together,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2017, pp. 61–65, 2017.
- [4] Sturm, B. L. and Ben-Tal, O., “Taking the Models back to Music Practice: Evaluating Generative Transcription Models built using Deep Learning,” *Journal of Creative Music Systems*, 2(1), pp. 1–15, 2017.
- [5] Pulkki, V., “Virtual Sound Source Positioning Using Vector Base Amplitude Panning,” *Journal of Audio Engineering Society*, 45(6), pp. 456–466, 1997.
- [6] Amershi, S., Cakmak, M., Knox, W. B., and Kulesza, T., “Power to the People: The Role

- of Humans in Interactive Machine Learning,” *AI Magazine*, 35(4), pp. 105–120, 2014.
- [7] Pulkki, V., “Spatial Sound Reproduction with Directional Audio Coding,” *Journal of Audio Engineering Society*, 55(6), pp. 503–516, 2007.
- [8] Mosqueira-Rey, E., Hernández-Pereira, E., Alonso-Ríos, D., Pazos-López, S., and Iglesias, R., “Human-in-the-Loop Machine Learning: A State of the Art,” *Artificial Intelligence Review*, 56(4), pp. 3005–3054, 2023.
- [9] Do, H., Dang, Q., Huang, J. Z., and Jain, D., “AdaptiveSound: An Interactive Feedback-Loop System to Improve Sound Recognition for Deaf and Hard of Hearing Users,” in *Proceedings of the 25th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '23)*, volume abs/1911.13254, pp. 18:1–18:12, 2023.
- [10] Fails, J. A. and Jr., D. R. O., “Interactive Machine Learning,” in *Proceedings of the 8th International Conference on Intelligent User Interfaces*, volume 8, pp. 39–45, 2003.
- [11] Katan, S., Grierson, M., and Fiebrink, R., “Using Interactive Machine Learning to Support Interface Development Through Workshops with Disabled People,” in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, 1, pp. 251–254, 2015.
- [12] Yuksel, B. F., Fazli, P., Mathur, U., Bisht, V., Kim, S. J., Lee, J. J., Jin, S. J., Siu, Y.-T., Miele, J. A., and Yoon, I., “Human-in-the-Loop Machine Learning to Increase Video Accessibility for Visually Impaired and Blind Users,” in *Proceedings of the 2020 ACM Designing Interactive Systems Conference (DIS '20)*, 1, pp. 47–60, 2020.
- [13] Drossos, K., Virtanen, T., Schuller, G., and Mimilakis, S., “Deep Neural Networks for Dynamic Range Compression in Mastering Applications,” in *Proceedings of the 2016 International Conference on Digital Audio Effects (DAFx-16)*, pp. 1–6, 2016.
- [14] Lambert, N., “Reinforcement Learning from Human Feedback,” *ArXiv preprint*, abs/2504.12501, 2025.
- [15] Chai, Y., Sun, H., Fang, H., Wang, S., Sun, Y., and Wu, H., “MA-RLHF: Reinforcement Learning from Human Feedback with Macro Actions,” 2025.
- [16] D’efosse, A., Usunier, N., Bottou, L., and Bach, F. R., “Music Source Separation in the Waveform Domain,” 2019.
- [17] Grais, E. M. and Plumbley, M. D., “Single channel audio source separation using convolutional denoising autoencoders,” in *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 1265–1269, 2017.
- [18] Stoller, D., Ewert, S., and Dixon, S., “Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation,” 2018.
- [19] Pons, J. and Serra, X., “musicnn: Pre-Trained Convolutional Neural Networks for Music Audio Tagging,” 2019.
- [20] Chen, Q. Z., Schnabel, T., Nushi, B., and Amer-shi, S., “HINT: Integration Testing for AI-Based Features with Humans in the Loop,” in *Proceedings of the 27th International Conference on Intelligent User Interfaces (IUI '22)*, 1, pp. 1–17, 2022.
- [21] Qian, J., Liu, X., Yu, Y., Jiang, Q., Wang, H., and Li, H., “Stripe-Transformer: Deep Stripe Feature Learning for Music Source Separation,” *Journal of Audio, Speech, and Music Processing*, 2023(2), pp. 1–14, 2023.
- [22] Justus, A. A., “Music Generation Using Human-In-The-Loop Reinforcement Learning,” 2023.