

USER-GUIDED GENERATIVE SOURCE SEPARATION

Yutong Wen Minje Kim Paris Smaragdis
University of Illinois at Urbana-Champaign

yutong12@illinois.edu

ABSTRACT

Music source separation (MSS) aims to extract individual instrument sources from their mixture. While most existing methods focus on the widely adopted four-stem separation setup (vocals, bass, drums, and other instruments), this approach lacks the flexibility needed for real-world applications. To address this, we propose GuideSep, a diffusion-based MSS model capable of instrument-agnostic separation beyond the four-stem setup. GuideSep is conditioned on multiple inputs: a waveform mimicry condition, which can be easily provided by humming or playing the target melody, and mel-spectrogram domain masks, which offer additional guidance for separation. Unlike prior approaches that relied on fixed class labels or sound queries, our conditioning scheme, coupled with the generative approach, provides greater flexibility and applicability. Additionally, we design a mask-prediction baseline using the same model architecture to systematically compare predictive and generative approaches. Our objective and subjective evaluations demonstrate that GuideSep achieves high-quality separation while enabling more versatile instrument extraction, highlighting the potential of user participation in the diffusion-based generative process for MSS. Our code and demo page are available at <https://yutongwen.github.io/GuideSep/>.

1. INTRODUCTION

Music source separation (MSS) aims to separate a mixture audio into its constituent sources, typically defined by the instrument. Since the 2015 Signal Separation Evaluation Campaign (SiSEC) [1], the MSS community has largely focused on supervised models to separate songs into four stems: vocals, bass, drums, and others that includes all remaining instruments, a setup commonly referred to as VBDO. Under this framework, numerous recent deep neural network (DNN) models have significantly advanced performance [2–8]. While this setup provides a convenient benchmark, it lacks the flexibility needed for real-world applications: ideally, MSS systems should be able to extract any target instrument of interest.

In this regard, several works have extended MSS beyond the VBDO setup. To enable the separation of arbitrary instruments, the model must first be provided with a condition specifying the target instrument, such as instrument class labels [9–12]. In [9, 11] this conditioning method is shown to work for the VBDO setup, whereas [10] extends this approach to 13 instruments. However, class labels can be vague, as instruments like the guitar may exhibit significant variability within the same label. Moreover, new instrument classes require re-training. Another approach, query-based MSS conditions the model using a sound example, where the model extracts sources similar to the example [13–18]. For instance, Watcharasupat et al. [16] designed a lightweight model capable of instrument-agnostic separation using a single query, while Wang et al. [18] developed a model that accepts up to five queries to improve performance stability. Despite its potential to provide rich information about the target source, query-based separation may be limited in real-world applications where high-quality queries are unavailable. Additionally, MSS models can be conditioned on MIDI score of the target instrument [19–23]. While MIDI information provides a strong and accurate cue, it is often unavailable in many real-world scenarios, such as pop music. Bryan et al. [24–27] proposed an alternative method where users sketch a rough mask on the spectrogram of the mixture to indicate the target source. However, this approach can suffer from ambiguity, as identifying the target instrument’s region in the mixture spectrogram is often challenging. Smaragdis et al. [28] leverages humming as a guidance to separate a target source. Unlike label-based or sound query conditioning, humming offers users greater flexibility when interacting with the system.

In this work, we propose a guided separation (GuideSep) method, a conditional complex-spectrogram domain diffusion model designed to address music source separation beyond the VBDO setup in an instrument-agnostic manner. Building on the observations of existing methods for MSS beyond VBDO, we condition the diffusion model on multiple inputs: a waveform mimicry to a target source and mel-spectrogram domain masks. While MIDI score information is often difficult to obtain in real-world scenarios, users are capable of providing a mimicry by humming or playing the target melody with an instrument of their choice. Additionally, we introduce a rough mask on the mel-spectrogram for the users to further inform the model of the region to focus on. During inference, either or both conditions can be utilized, offering



users a flexible way to specify the target source for separation from the mixture. Our diffusion model is built on EDMSound [29], a complex-spectrogram domain diffusion method designed for both unconditional and label-conditioned audio generation. We modify the model backbone to support multiple conditioning inputs.

Traditionally, audio source separation has been tackled using predictive models¹, which map mixture input to an estimated clean output by minimizing a point-wise loss function [31–33]. While predictive models often struggle with residual noise, artifacts [34] in enhancement tasks, generative models have the potential to produce cleaner results by directly or indirectly modeling the clean prior. In recent years, significant progress has been made in applying generative models to audio separation tasks, particularly in speech enhancement and separation [35–42]. While most music source separation (MSS) methods are still predictive, a few generative approaches have begun to emerge. For instance, Ge et al. proposed a flow-based model, InstGlow [43], which leverages the priors of clean sources to improve separation results within the VBDO setup. Additionally, multi-source diffusion models have been proposed for simultaneous music source separation and generation [44, 45]. These approaches employ a multi-channel diffusion process to model the joint distribution of individual sources and condition on the mixture to sample individual sources during inference, enabling separation. While this formulation provides control over which instrument to synthesize or separate, it is limited to the specific set of instruments the model is trained on.

While there is growing interest in applying generative methods to MSS, to the best of our knowledge, no prior work has systematically compared generative methods with their direct counterparts. In this work, we address this gap by designing a mask-prediction baseline that shares the exact same model backbone as our diffusion model. We then conduct a systematic evaluation to analyze the differences between the two approaches.

Our contributions can be summarized as follows: 1) We propose GuideSep, one of the first diffusion-based models designed to address music source separation beyond the VBDO setup and we release the codebase 2) We introduce versatile, instrument-agnostic conditions—waveform mimicry conditions and mel-spectrogram domain masks—that are more practical for real-world applications 3) We design a mask-prediction baseline using the same model architecture and conduct a systematic evaluation to analyze the differences between predictive and generative approaches.

2. THE PROPOSED GUIDESEEP METHOD

GuideSep is a diffusion model conditioned by user input. Our approach leverages users’ input describing a source, i.e., the raw waveform of user mimicry to a target source as well as a rough mask in the mel-spectrogram domain.

¹ Some literature refers to predictive models as discriminative or deterministic. Lemerrier et al. [30] note that predictive models encompass both concepts.

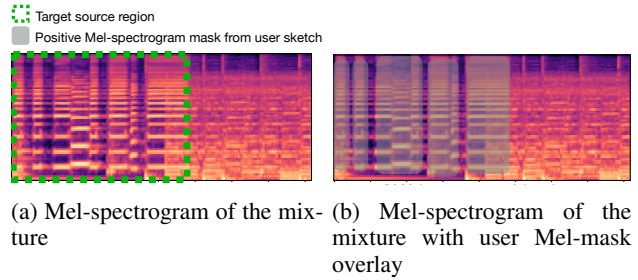


Figure 1: Illustration of an example of positive user-input Mel-spectrogram mask.

2.1 Condition signals

2.1.1 Mimicry condition

The mimicry guidance is a user-provided time-domain waveform, such as a hummed rendition of the target melody or the melody played on another instrument. Due to the lack of real-world data for training, we simulate the mimicry guidance by converting the ground-truth MIDI score of the target source to audio using the FluidSynth library [46]. Real-world mimicry inputs often include off-pitch notes, imperfect timing, and limitations in note range. Additionally, since many instruments and vocal mimicry are monophonic, it poses significant challenges when extracting polyphonic sources, e.g., guitar or piano. We simulate user input via various data augmentation techniques:

- **Off-pitch** melodies are simulated by introducing perturbations to the MIDI notes. Each note has a 50% probability of being pitch-bent, whose amount is randomly sampled from a uniform distribution ranging from -0.4 to $+0.4$ semitones.
- **Imperfect timing** is simulated by introducing variations in the timing of MIDI notes. With a 40% probability, the start and end times of a note are shifted by up to ± 30 milliseconds. The time shift of a note will also be applied to its following notes.
- **Limitation of note range** is simulated by randomly shifting MIDI notes up or down by one octave with a 50% probability.
- **Extraction using non-polyphonic instruments:** We restrict the condition melody to be monophonic to reflect real-world limitations of many instruments and humming. It encourages the model to infer missing notes of the target source using other side information, such as the mel-spectral mask. The choice of a monophonic condition was driven by our focus on human voice guidance; however, this is a limitation of the training data rather than the algorithm itself.

2.1.2 Mel-spectral masks

Our second conditioning input is the user-created mask in the mel-spectrogram domain, that distinguishes regions corresponding to the target source from those of background sources. While being conceptually aligned with [24], GuideSep uses it to condition a deep generative model. Specifically, we define two types of masks: pos-

itive and negative masks to indicate the target and background source regions, respectively. The mel-spectrogram domain is chosen due to its greater interpretability and easier identification of the sources compared to the Fourier transform’s linear frequency scale.

Figure 1 illustrates the process of creating a mel-spectrogram mask based on user input. We implement a user interface where users can sketch on the mel-spectrogram of the mixture with different brush size and confidence level to indicate regions they believe correspond to the target source or background music. In practice, user-provided masks may exclude portions of the target source or unintentionally include regions of background sound. Additionally, in many cases, the target source significantly overlaps the background sources, further complicating the masking process. To simulate these real-world imperfections during training, we generate synthetic user input masks by applying a Gaussian filter, whose standard deviation ranges between 4 to 6, to the ground-truth mel-spectrograms of the target source and the residual sources. In addition, we randomly drop out 40% of patches.

2.2 Conditional complex spectrogram diffusion

Diffusion probabilistic models (DPMs) [47, 48] consist of two key processes: progressively corrupting training data by adding noise until it approximates a normal distribution, and learning to reverse each step of this noise corruption using the same functional form. These models can be generalized as score-based generative models [49], which utilize an infinite number of noise scales, enabling both the forward and backward diffusion processes to be described by stochastic differential equations (SDEs). During inference, the reverse SDE is employed to generate samples numerically, starting from a standard normal distribution.

Complex spectrogram diffusion with EDM: Our work is based on EDMSound [29]. We train our diffusion model using the EDM framework [50], which reformulates the diffusion SDE in terms of noise scales rather than drift and diffusion coefficients. To ensure that the inputs of the neural network are appropriately scaled within the range $[-1, 1]$, as required by the diffusion models, we apply an amplitude transformation to the complex spectrogram inputs. Specifically, we use $\tilde{c} = \beta|c|^\alpha e^{i\angle c}$, as proposed in [42, 51], where $\alpha \in (0, 1]$ is a compression factor that emphasizes time-frequency bins with lower energy, $\angle c$ denotes the phase of the original complex spectrogram c , and $\beta \in \mathbf{R}_+$ is a scaling factor that normalizes amplitudes approximately to the range $[0, 1]$.

Adding conditions to EDMSound: To adapt EDMSound for target sound extraction, we modify the network to accept conditional inputs, including the mixture signal, mimicry signal, and spectral masks. Rather than modeling $p(\mathbf{s}|\mathbf{c})$, where \mathbf{s} is the target source and \mathbf{c} is an instrument label, we instead model $p(\mathbf{s}|\mathbf{c}_{\text{mix}}, \mathbf{c}_{\text{mimicry}}, \mathbf{c}_{\text{masks}})$. Here, \mathbf{c}_{mix} corresponds to the music mixture represented in the complex-valued short-time Fourier transform (STFT) domain, while $\mathbf{c}_{\text{mimicry}}$ denotes the mimicry condition in the

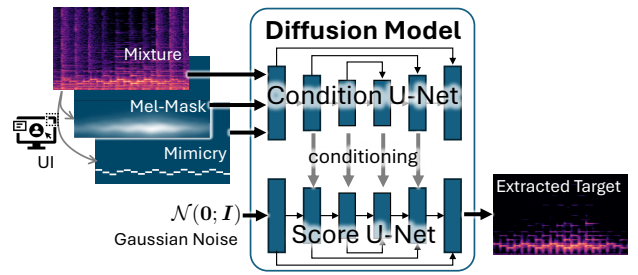


Figure 2: Overview of the GuideSep at inference time. Our model accepts mimicry condition and mel-spectrogram domain masks as guidance from users to extract the target source from the mixture.

form of magnitude STFT, assuming phase information is a distraction when it comes to representing spectrum information. Finally, $\mathbf{c}_{\text{masks}}$ refers to the normalized magnitudes of mel-spectrogram masks, ranged between 0 and 1.

The proposed architecture: Building on insights from prior works [40, 52–54], we design our model as depicted in Figure 2. The architecture comprises two primary U-Net structures. The first one, referred to as the score U-Net, aligns with the original U-Net used in EDMSound. If it were not for conditioning input, this part of the model performs a blind audio synthesis by taking a Gaussian noise sample. The second module, the condition U-Net, is introduced to tame this otherwise entirely generative behavior of the score U-Net. The condition U-Net is dedicated to processing all conditional inputs, including the mixture. These two U-Nets are connected so that the output of each layer in the condition U-Net is element-wise added to its corresponding layer in the score U-Net, spanning both the downsampling and upsampling layers. Since the mel-scale masks are in a different frequency dimension compared to the magnitude and complex spectrograms, we introduce a simple 1-hidden-layer neural network to project the mel-frequency axis onto the spectrogram frequency axis. Since there are three conditions in the form of a spectrogram—mixture, mimicry, and projected masks—we concatenate them along the channel dimension and feed as input to the condition U-Net. Our U-Net architecture is adapted from Imagen [55], chosen for its high sample quality, rapid convergence, and memory efficiency.

Loss function: During training, we optimize the model using preconditioned denoising score matching, following [50]. The training objective is formulated as

$$\mathbb{E}_{\mathbf{s}} \mathbb{E}_{\mathbf{n}} \left[\lambda(\sigma) \| D(\mathbf{s} + \mathbf{n}; \sigma, \mathbf{c}_{\text{mix}}, \mathbf{c}_{\text{mimicry}}, \mathcal{M}(\mathbf{c}_{\text{masks}})) - \mathbf{s} \|_2^2 \right],$$

where $D(\cdot)$ is the EDM weighted neural network, σ is the noise level, $\lambda(\cdot)$ is the loss weighting which is $(\sigma^2 - \sigma_{\text{data}}^2) / (\sigma \cdot \sigma_{\text{data}}^2)$ for the EDM framework, $\mathcal{M}(\cdot)$ denotes the 1-hidden-layer projection network, and $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$ is Gaussian noise.

Inference: Within the EDM framework, the probability flow ordinary differential equation (ODE) can be simplified into a nonlinear ODE, allowing the direct use of standard off-the-shelf ODE solvers, such as high-order Ex-

ponential Integrator (EI)-based ODE solvers [56], specifically multistep DPM solvers [56, 57], for sampling as in EDMSound.

3. EXPERIMENT

We conduct experiments using the Slakh2100 dataset [58] augmented by MoisesDB [59] for training. The Slakh2100 dataset provides an official train-validation-test split, which we utilize as well. We evaluate our model’s performance using the widely adopted signal-to-distortion ratio (SDR) metrics [60, 61].

3.1 Training and model details

3.1.1 The datasets

Slakh2100 is a synthetic dataset of waveform-MIDI-aligned music dataset containing 2,100 tracks in total around 145 hours of audio. In our training process, instead of using the original mix from the dataset, we generate training data through random mixing. This way allows for nearly infinite variations of training samples. While this approach may result in the loss of some musical context, previous work [62] has demonstrated that random mixing can improve MSS model performance. To enhance our model’s performance on real-world music, we utilize the MoisesDB dataset [59] to construct background sources. MoisesDB is a comprehensive multitrack dataset designed for source separation beyond 4-stems, featuring 240 previously unreleased songs by 47 artists across twelve high-level genres, in total approximately 14 hours of audio. During random mixing, we randomly select 3 to 6 sources from the MoisesDB dataset to serve as background music, while the target source is drawn from the Slakh2100 dataset. The background and target sources are mixed at signal-to-noise ratios (SNR) ranging from -5 dB to 5 dB. All The input audio is converted to single channel and resampled to 16 kHz, and then trimmed or padded to around 4.1 seconds for batched training.

3.1.2 Dropout strategies

To ensure that the model can process any combination of input types, we incorporate dropout strategies during training. This allows the model to operate with an incomplete set of conditions, such as the mimicry-only or mel-spectrogram-mask-only cases. To this end, we randomly drop out either the mimicry condition or mel-spectrogram masks, ensuring that the model learns to predict the target source even when provided with partial conditioning information. Additionally, we empirically observe that the model benefits from a mimicry-only conditioned synthesis tasks, which happens when we randomly drop the mixture input c_{mix} during training. This encourages the model to infer the target source from melodic guidance alone. Specifically, during training, we drop 30% of the mimicry condition, 70% of the mel-masks, and 10% of the mixture. The high dropout rate for mel masks is intentional and tuned using the validation split, as they provide a strong cue to the target source. By reducing their presence, the

model is encouraged to focus more on learning from the mimicry condition.

3.1.3 The model architecture

For both score and condition U-Net modules, we utilize an efficient U-Net architecture adapted from the open-source Imagen implementation², which is known for memory efficiency and fast convergence. Both U-Nets incorporate downsampling and upsampling blocks, each containing two ResNet blocks with a self-attention layer that uses two attention heads. The bottleneck dimension is 128. The complete model has 93.3 million trainable parameters.

The input to the condition U-Net consists of three types: a complex spectrogram c_{mix} , a magnitude spectrogram c_{mimicry} , both with a window size of 512 samples and a hop size of 256 samples, and the mel-spectrogram masks c_{mask} , which share the same hop size and consist of 80 mel-frequency bins. Eventually, it is a five-channel spectrogram input: two for the complex spectrogram, one for the magnitude spectrogram, and two for the positive (i.e., target source) and negative (i.e., background music) masks.

The score U-Net, as an autoencoder, defines a two-channel spectrograms as its input and output representation, where the two channels represent the real and imaginary components of the complex spectrogram, respectively. Note that in the very beginning of the sampling process, the input spectrogram to the score U-Net is noise sampled from Gaussian. Additionally, we condition the network on logarithmically scheduled noise levels σ .

3.1.4 Inference

For inference, we employ an EI-based DPM sampler [56, 57]. To ensure compatibility between the EDM framework samplers and arbitrary training objectives during inference, we implement input rescaling as needed. Specifically, we rescale both the noisy inputs and noise levels to align with the network’s original training-time scales. The results, presented in Section 4, are obtained using an 8-step sampler configuration.

3.1.5 Training details

Our model is trained with a batch size of 36 and a learning rate of 1×10^{-4} using the Adam optimizer. The training process runs for 300k updates. We used two NVIDIA L40S GPUs, and trained for ten days.

3.2 Baselines

To the best of our knowledge, no existing work offers a fair comparison, as our method introduces a novel conditioning approach. However, we design a traditional mask-prediction model to compare the proposed generative approach against. The baseline shares the same twin U-Net architecture and structural details as our diffusion backbone. In particular, the input to the score U-Net portion is the magnitude spectrogram of the mixture, while the input to the condition U-Net consists of the magnitude spectrogram of the mimicry condition and the masks. The model

² <https://github.com/lucidrains/imagen-pytorch>

Model	Piano	Guitar	Bass	Strings	Brass	Synth	Pipe	Reed	Organ	Chromatic Percussion	Overall
Ours (full)	8.34 ±0.11	10.53 ±0.09	11.97 ±0.12	9.64 ±0.12	9.15 ±0.38	9.25 ±0.20	15.58 ±0.27	13.78 ±0.24	13.44 ±0.22	11.53 ±0.36	10.46
Baseline (full)	7.03±0.09	8.72±0.08	8.69±0.06	9.06±0.13	8.03±0.31	8.00±0.17	14.99±0.26	11.94±0.23	11.25±0.23	9.62±0.31	8.74
Ours (mimicry only)	7.46±0.11	9.96±0.10	11.19±0.13	8.63±0.14	7.95±0.45	8.13±0.23	14.43±0.34	13.14±0.27	12.39±0.25	8.74±0.43	9.60
w/ pseudo-masks	7.99±0.11	10.18±0.10	9.87±0.15	8.72±0.15	8.21±0.40	8.19±0.25	14.81±0.31	13.20±0.26	12.20±0.29	8.26±0.55	9.56
Ours (positive mask only)	7.86±0.11	10.17±0.09	11.45±0.13	9.48±0.12	8.97±0.38	9.14±0.19	15.19±0.28	13.42±0.25	13.08±0.23	11.09±0.38	10.09
Ours (humming)*	-	-	-	-	-	-	-	-	-	-	13.61
Frequency (%)	20.71	27.89	17.79	15.23	2.65	4.74	2.72	3.06	3.43	1.78	-

Table 1: SDR (dB) results with 95% confidence interval (higher values indicate better performance) for ten instrument classes in the Slakh2100 test split. The results include GuideSep (our method) under various input conditions and the mask-prediction baseline. The best scores are highlighted in bold. For asterisk (*) please refer to Section 4.2.

outputs a non-binary mask by applying a sigmoid function after the output layer, which is then used to compute the target source magnitude spectrogram through element-wise multiplication with the input mixture magnitude spectrogram. The final waveform is reconstructed by combining the predicted magnitude spectrogram with the phase information from the original mixture. We train the mask-prediction baseline using the L2 reconstruction loss in the magnitude spectrogram domain, with the same learning rate, batch size, and number of updates as our diffusion model. Note that, due to the absence of time-step conditional inputs and differences in input channels, the mask-prediction baseline contains 80.3 million parameters.

4. EVALUATION AND DISCUSSION

We evaluate our model on the official test split of the Slakh2100 dataset. The mimicry condition signals are synthesized as described in Section 2.1, using randomly selected virtual instruments from the FluidSynth library. Similarly, the positive and negative masks are simulated following the same procedure outlined in Section 2.1. For evaluation, we group the instrument classes in Slakh2100 into ten broader categories, where drum tracks are excluded from the target sources, because our synthesis method does not apply to them. The evaluation results are presented in Table 1.

In the first two rows in Table 1, we present the results of our model and the mask-prediction baseline. Both models utilize the mimicry condition and mel-spectrogram masks during inference, denoted with ‘(full)’ in the table. The results demonstrate that our model consistently outperforms the mask-prediction baseline across all instrument classes. Given that the mask-prediction baseline shares the same model backbone, training data, and configuration as our diffusion model, the performance gap highlights the benefits of using diffusion approach. In the listening test, we observe that the mask-prediction baseline often reconstructs target sources which still contain interferences. In contrast, while our diffusion model may occasionally exhibit inexact timbre, it generally generates cleaner target sources. This can be attributed to the diffusion model learning a prior distribution of clean sources, which biases its outputs toward cleaner results. Although our findings align to the well-known behavior of generative models, our experiments are limited to the particular choice of the dif-

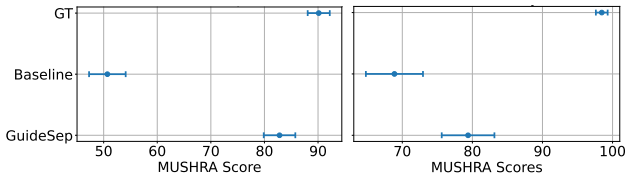
fusion model and a masking-based baseline with a matching architecture, leaving more general arguments to future work. We also observe that both models work better for the monophonic sources than the polyphonic ones, such as piano, guitar, strings, and synth, where our strictly monophonic mimicry condition is not informative enough. As a result, the models may struggle with missing notes from chords, extracting the wrong target instrument, or even extracting multiple instruments when they share a similar melody, which is common in music. For results on the real-world conditions, please refer to our demo page.

4.1 Subjective Listening Tests

In addition to the SDR results, we conduct a subjective listening test to further evaluate our model. We modify webMUSHRA [63] so the test comprises two sections: the first assesses the overall quality of the model’s separation results, while the second focuses specifically on evaluating the timbre of the reconstructed target source. In the first section, each question presents the music mixture as a reference. Participants are asked to compare and rate four stimuli: the ground truth, the mixture itself (i.e., the hidden reference), and the predictions from our model and the mask-prediction baseline. Participants are unaware that one of the stimuli is the actual ground truth and are instead told that the three stimuli are potential reconstructions of a target source. The participants are asked to first identify the mixture and assign it a score of 0, then rate the remaining stimuli (e.g., with 100 being a perfect match) based on how closely they resemble the target source in the mixture. This part consists of ten trials, with each mixture sample randomly selected from a different instrument class in the Slakh2100 test split. We use the mixture as a reference instead of the ground-truth source in order to measure the listener’s opinion on the “synthesized” source without introducing any prejudice.

To make up the modification introduced in the first part, the second part is dedicated to evaluating the potential artifact specific to the generative models, i.e., the timbre change. This time, each trial presents the ground-truth target source as the reference. Participants compare and rate three stimuli: the hidden reference (i.e., the ground truth itself) and the predictions from our model and the mask-prediction baseline. However, ratings are based on timbre similarity to the reference, with 100 indicating an exact match and 0 representing a completely different timbre.

Participants are instructed to focus solely on the timbre of the target source while disregarding any interference or artifacts. Both parts use the same set of music samples, but the second part is presented only after participants complete the first part to avoid bias, ensuring they remain unaware that the ground truth was included in the first part.



(a) Sec. 1, MUSHRA result on separation quality (b) Sec. 2, MUSHRA result on timbre similarity

Figure 3: Mean MUSHRA Score with 95% confidence interval of the subjective listening test on separation quality and separation timbre quality.

A total of 13 participants took part in the subjective listening test, and the results from both parts are presented in Figure 3. In the first part, where participants rated the separation quality, our model scored 82.82 ± 2.95 , the ground truth scored 90.13 ± 2.06 , and the mask-prediction baseline scored 50.69 ± 3.41 . Notably, despite the mask-prediction baseline having a relatively small SDR difference from our model, the listening test revealed a significant gap in perceptual evaluation. This suggests that users may perceive a cleaner target source prediction as more satisfactory, even if a slightly noisier prediction achieves a decent sample-wise similarity to the target source.

In the second part, where participants rated timbre preservation, our model scored 79.38 ± 3.76 , surpassing the mask-prediction baseline, which scored 68.88 ± 4.07 . In theory, the mask-prediction baseline could preserve the original timbre better, but our subjective listening test results suggest otherwise. Based on the listeners’ feedback, we speculate that this outcome is influenced by the nature of the target source extraction task, where multiple sources in a musical piece may share similar melodic patterns. As a result, the mask-prediction baseline’s output can be contaminated by interfering similar melodies, which can be perceived as a timbral change rather than artifacts.

4.2 Ablation

Beyond evaluating our model with both conditioning signals, we conduct an ablation study to assess its performance under different input conditions.

Mimicry-only: We evaluate the model using the ‘(mimicry only)’ setup (Table 1). We observe a slight overall decrease in performance, indicating that while mel-masks contribute to improved performance, the model remains effective even when conditioned solely on the melody signal.

Pseudo-masks: When only the mimicry condition is available, we can generate pseudo mel-masks using the mimicry condition and the mixture. Specifically, we use the Gaussian-blurred mel-spectrogram of the mimicry con-

dition as the positive mel-mask and the blurred mixture as the negative mel-mask with the standard deviation set to be 5. In Table 1, although the overall SDR score is slightly lower compared to using only the mimicry condition, the model performs better with pseudo-masks for 7 out of 10 instrument classes. This suggests that pseudo-masks can generally enhance the model’s performance at no additional cost. The bass class is an exception, likely due to its limited high-frequency content, which sets it apart from other instruments. Consequently, the mel-spectrogram mask may be misleading in this case. A different standard deviation for the Gaussian filter could work better, while it involves an additional hyperparameter search.

Mel-masks-only: Another case is when only the mel-masks are used for conditioning. We observe that the results are generally better than those obtained using only the mimicry condition, indicating that mel-masks serve as highly effective conditioning signals.

Humming-only: Although in our training, mimicry condition do not include humming, we evaluate our model to assess its generalization to unseen mimicry condition, such as humming. Since we cannot easily synthesize humming from MIDI, we utilize the HumTrans dataset [64], a MIDI-humming aligned dataset, resulting in an evaluation dataset of approximately 16.6 hours. Since HumTrans melodies do not coincide with our test songs, an ideal source separation setup is impossible to design. Instead, we synthesize background sources by randomly mixing 3 to 6 sources from the MoisesDB dataset, following our training procedure described in Section 3.1. The target source is synthesized from the MIDI information aligned to the humming, using the method outlined in Section 2.1 with augmentation. As the virtual instruments are sampled from the FluidSynth library, which is not directly comparable to the Slakh2100 benchmark, we report only an overall SDR result, whose mean is 13.61 dB. This score exceeds the overall SDR result of our model on the Slakh2100 benchmark, demonstrating that the mimicry condition can generalize to humming during inference. However, the strong performance could also be attributed to the random mixing used during evaluation, which simplifies the task of target source extraction for the model.

5. CONCLUSION

We introduced GuideSep, a diffusion-based music source separation model that enables flexible, instrument-agnostic separation using waveform mimicry conditions and mel-spectrogram masks, and released the codebase. Our results demonstrate that this approach achieves high-quality separation while offering greater adaptability compared to traditional class-based methods. Additionally, our comparison with a mask-prediction baseline provides insights into the strengths of generative models for MSS. This work highlights the potential of diffusion models in advancing more versatile and user-controllable source separation.

6. REFERENCES

- [1] N. Ono, Z. Raffi, D. Kitamura, N. Ito, and A. Litkus, "The 2015 signal separation evaluation campaign," in *Latent Variable Analysis and Signal Separation*, E. Vincent, A. Yeredor, Z. Koldovský, and P. Tichavský, Eds. Cham: Springer International Publishing, 2015, pp. 387–395.
- [2] S. Rouard, F. Massa, and A. Défossez, "Hybrid transformers for music source separation," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.
- [3] J. Chen, S. Vekkot, and P. Shukla, "Music source separation based on a lightweight deep learning framework (dtnet: Dual-path tfc-tdf unet)," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 656–660.
- [4] R. Sawata, N. Takahashi, S. Uhlich, S. Takahashi, and Y. Mitsufuji, "The whole is greater than the sum of its parts: improving music source separation by bridging networks," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2024, no. 1, p. 39, 2024.
- [5] W. Tong, J. Zhu, J. Chen, S. Kang, T. Jiang, Y. Li, Z. Wu, and H. Meng, "SCNet: Sparse compression network for music source separation," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 1276–1280.
- [6] N. Takahashi and Y. Mitsufuji, "D3Net: Densely connected multidilated densenet for music source separation," *arXiv preprint arXiv:2010.01733*, 2020.
- [7] Y. Luo and J. Yu, "Music source separation with band-split rnn," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1893–1901, 2023.
- [8] W.-T. Lu, J.-C. Wang, Q. Kong, and Y.-N. Hung, "Music source separation with band-split rope transformer," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 481–485.
- [9] G. Meseguer-Brocal and G. Peeters, "Conditioned-U-Net: Introducing a control mechanism in the unet for multiple source separations," *arXiv preprint arXiv:1907.01277*, 2019.
- [10] O. Slizovskaia, L. Kim, G. Haro, and E. Gomez, "End-to-end sound source separation conditioned on instrument labels," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 306–310.
- [11] P. Seetharaman, G. Wichern, S. Venkataramani, and J. Le Roux, "Class-conditional embeddings for music source separation," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 301–305.
- [12] D. Samuel, A. Ganeshan, and J. Naradowsky, "Meta-learning extractors for music source separation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 816–820.
- [13] P. Smaragdis, "User guided audio selection from complex sound mixtures," in *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, 2009, pp. 89–92.
- [14] J. H. Lee, H.-S. Choi, and K. Lee, "Audio query-based music source separation," *arXiv preprint arXiv:1908.06593*, 2019.
- [15] E. Manilow, G. Wichern, and J. Le Roux, "Hierarchical musical instrument separation," in *ISMIR*, 2020, pp. 376–383.
- [16] K. N. Watcharasupat and A. Lerch, "A stem-agnostic single-decoder system for music source separation beyond four stems," *arXiv preprint arXiv:2406.18747*, 2024.
- [17] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, "Zero-shot audio source separation through query-based learning from weakly-labeled data," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 4, 2022, pp. 4441–4449.
- [18] Y. Wang, D. Stoller, R. M. Bittner, and J. P. Bello, "Few-shot musical source separation," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 121–125.
- [19] S. Ewert and M. B. Sandler, "Structured dropout for weak label and multi-instance learning and its application to score-informed source separation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2277–2281.
- [20] M. Miron, J. Janer, and E. Gómez, "Monaural score-informed source separation for classical music using convolutional neural networks," in *ISMIR*, vol. 2017, 2017, pp. 55–62.
- [21] M. Gover, "Score-informed source separation of choral music," 2020.
- [22] A. J. Munoz-Montoro, J. J. Carabias-Orti, P. Vera-Candeas, F. J. Canadas-Quesada, and N. Ruiz-Reyes, "Online/offline score informed music signal decomposition: application to minus one," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2019, pp. 1–30, 2019.

- [23] Y.-N. Hung, G. Wichern, and J. Le Roux, "Transcription is all you need: Learning to separate musical mixtures with score as supervision," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 46–50.
- [24] N. J. Bryan, G. J. Mysore, and G. Wang, "ISSE: An interactive source separation editor," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2014, pp. 257–266.
- [25] N. J. Bryan and G. J. Mysore, "Interactive user-feedback for sound source separation," in *International Conference on Intelligent User-Interfaces (IUI), Workshop on Interactive Machine Learning*. Santa Monica, 2013.
- [26] N. Bryan and G. Mysore, "An efficient posterior regularized latent variable model for interactive sound source separation," in *International conference on machine learning*. PMLR, 2013, pp. 208–216.
- [27] N. J. Bryan and G. J. Mysore, "Interactive refinement of supervised and semi-supervised sound source separation estimates," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 883–887.
- [28] P. Smaragdis and G. J. Mysore, "Separation by "humming": User-guided sound extraction from monophonic mixtures," in *2009 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2009, pp. 69–72.
- [29] G. Zhu, Y. Wen, M.-A. Carbonneau, and Z. Duan, "EDMSound: Spectrogram based diffusion models for efficient and high-quality audio synthesis," *arXiv preprint arXiv:2311.08667*, 2023.
- [30] J.-M. Lemerrier, J. Richter, S. Welker, E. Moliner, V. Välimäki, and T. Gerkmann, "Diffusion models for audio restoration: A review [special issue on model-based and data-driven audio signal processing]," *IEEE Signal Processing Magazine*, vol. 41, no. 6, pp. 72–84, 2025.
- [31] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 27, no. 8, pp. 1256–1266, 2019.
- [32] Y. Hu, Y. Liu, S. Lv, M. Xing, S. Zhang, Y. Fu, J. Wu, B. Zhang, and L. Xie, "DCCRN: Deep complex convolution recurrent network for phase-aware speech enhancement," *arXiv preprint arXiv:2008.00264*, 2020.
- [33] H. J. Park, B. H. Kang, W. Shin, J. S. Kim, and S. W. Han, "MANNER: Multi-view attention network for noise erasure," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7842–7846.
- [34] J. Pirklbauer, M. Sach, K. Fluyt, W. Tirry, W. Wardah, S. Moeller, and T. Fingscheidt, "Evaluation metrics for generative speech enhancement methods: Issues and perspectives," in *Speech Communication; 15th ITG Conference*. VDE, 2023, pp. 265–269.
- [35] E. Postolache, G. Mariani, M. Mancusi, A. Santilli, L. Cosmo, and E. Rodolà, "Latent autoregressive source separation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 8, 2023, pp. 9444–9452.
- [36] Y. C. Subakan and P. Smaragdis, "Generative adversarial source separation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 26–30.
- [37] B. Chen, C. Wu, and W. Zhao, "SEPDIF: Speech separation based on denoising diffusion model," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.
- [38] S. Lutati, E. Nachmani, and L. Wolf, "Separate and diffuse: Using a pretrained diffusion model for improving source separation," *arXiv preprint arXiv:2301.10752*, 2023.
- [39] R. Scheibler, Y. Ji, S.-W. Chung, J. Byun, S. Choe, and M.-S. Choi, "Diffusion-based generative speech source separation," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.
- [40] R. Scheibler, Y. Fujita, Y. Shirahata, and T. Komatsu, "Universal score-based speech enhancement with high content preservation," *arXiv preprint arXiv:2406.12194*, 2024.
- [41] Z. Guo, Q. Wang, J. Du, J. Pan, Q.-F. Liu, and C.-H. Lee, "A variance-preserving interpolation approach for diffusion models with applications to single channel speech enhancement and recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.
- [42] J. Richter, S. Welker, J.-M. Lemerrier, B. Lay, and T. Gerkmann, "Speech enhancement and dereverberation with diffusion-based generative models," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2351–2364, 2023.
- [43] G. Zhu, J. Darefsky, F. Jiang, A. Selitskiy, and Z. Duan, "Music source separation with generative flow," *IEEE Signal Processing Letters*, vol. 29, pp. 2288–2292, 2022.
- [44] G. Mariani, I. Tallini, E. Postolache, M. Mancusi, L. Cosmo, and E. Rodolà, "Multi-source diffusion models for simultaneous music generation and separation," *arXiv preprint arXiv:2302.02257*, 2023.

- [45] T. Karchkhadze, M. R. Izadi, and S. Dubnov, “Simultaneous music separation and generation using multi-track latent diffusion models,” *arXiv preprint arXiv:2409.12346*, 2024.
- [46] D. Henningsson and F. Team, “Fluidsynth real-time and thread safety challenges,” in *Proceedings of the 9th International Linux Audio Conference, Maynooth University, Ireland*, 2011, pp. 123–128.
- [47] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.
- [48] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, “Deep unsupervised learning using nonequilibrium thermodynamics,” in *International conference on machine learning*. PMLR, 2015, pp. 2256–2265.
- [49] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, “Score-based generative modeling through stochastic differential equations,” *arXiv preprint arXiv:2011.13456*, 2020.
- [50] T. Karras, M. Aittala, T. Aila, and S. Laine, “Elucidating the design space of diffusion-based generative models,” *Advances in neural information processing systems*, vol. 35, pp. 26 565–26 577, 2022.
- [51] C. Breithaupt and R. Martin, “Analysis of the decision-directed snr estimator for speech enhancement with respect to low-snr and transient conditions,” *IEEE transactions on audio, speech, and language processing*, vol. 19, no. 2, pp. 277–289, 2010.
- [52] J. Serrà, S. Pascual, J. Pons, R. O. Araz, and D. Scaini, “Universal speech enhancement with score-based diffusion,” *arXiv preprint arXiv:2206.03065*, 2022.
- [53] S.-L. Wu, C. Donahue, S. Watanabe, and N. J. Bryan, “Music ControlNet: Multiple time-varying controls for music generation,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 2692–2703, 2024.
- [54] H. F. García, O. Nieto, J. Salamon, B. Pardo, and P. Seetharaman, “Sketch2Sound: Controllable audio generation via time-varying signals and sonic imitations,” *arXiv preprint arXiv:2412.08550*, 2024.
- [55] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, “Photorealistic text-to-image diffusion models with deep language understanding,” *Advances in neural information processing systems*, vol. 35, pp. 36 479–36 494, 2022.
- [56] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, “DPM-Solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 5775–5787, 2022.
- [57] ———, “DPM-Solver++: Fast solver for guided sampling of diffusion probabilistic models,” *arXiv preprint arXiv:2211.01095*, 2022.
- [58] E. Manilow, G. Wichern, P. Seetharaman, and J. Le Roux, “Cutting music source separation some Slakh: A dataset to study the impact of training data quality and quantity,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2019.
- [59] I. Pereira, F. Araújo, F. Korzeniowski, and R. Vogl, “MoisesDB: A dataset for source separation beyond 4-stems,” *arXiv preprint arXiv:2307.15913*, 2023.
- [60] E. Vincent, R. Gribonval, and C. Févotte, “Performance measurement in blind audio source separation,” *IEEE transactions on audio, speech, and language processing*, vol. 14, no. 4, pp. 1462–1469, 2006.
- [61] R. Scheibler, “SDR—medium rare with fast computations,” in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 701–705.
- [62] C.-B. Jeon, G. Wichern, F. G. Germain, and J. Le Roux, “Why does music source separation benefit from cacophony?” in *2024 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*. IEEE, 2024, pp. 873–877.
- [63] M. Schoeffler, S. Bartoschek, F.-R. Stöter, M. Roess, S. Westphal, B. Edler, and J. Herre, “web-MUSHRA—a comprehensive framework for web-based listening tests,” *Journal of Open Research Software*, vol. 6, no. 1, 2018.
- [64] S. Liu, X. Li, D. Li, and Y. Shan, “HumTrans: A novel open-source dataset for humming melody transcription and beyond,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 7915–7919.