CARNATIC SINGING VOICE SEPARATION USING COLD DIFFUSION ON TRAINING DATA WITH BLEEDING

Genís Plaja-Roglans[⋆] **Marius Miron**[†] **Adithi Shankar**[⋆] **Xavier Serra**[⋆]
[⋆]Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
[†]Earth Species Project
genis.plaja@upf.edu

ABSTRACT

Supervised music source separation systems using deep learning are trained by minimizing a loss function between pairs of predicted separations and ground-truth isolated sources. However, open datasets comprising isolated sources are few, small, and restricted to a few music styles. At the same time, multi-track datasets with source bleeding are usually found larger in size, and are easier to compile. In this work, we address the task of singing voice separation when the ground-truth signals have bleeding and only the target vocals and the corresponding mixture are available. We train a *cold diffusion* model on the frequency domain to iteratively transform a mixture into the corresponding vocals with bleeding. Next, we build the final separation masks by clustering spectrogram bins according to their evolution along the transformation steps. We test our approach on a Carnatic music scenario for which solely datasets with bleeding exist, while current research on this repertoire commonly uses source separation models trained solely with Western commercial music. Our evaluation on a Carnatic test set shows that our system improves Spleeter on interference removal and it is competitive in terms of signal distortion. Code is open sourced. [1]

1. INTRODUCTION

Music source separation (MSS) is a core task in the field of music information retrieval (MIR) in which the aim is to automatically separate the different sources in a musical mixture. In this work, we focus on separating the singing voice. In recent years, impressive performance for this difficult and highly undetermined problem has been achieved through the use of deep learning (DL) approaches [1]. Traditionally, MSS models operate on time-frequency representations [1–3], and more recently on waveforms [4, 5], however, the latter are prone to introduce artifacts to the estimated sources. While the combination of both domains has also shown impressive performance [6, 7], these models tend to be large in size and require extensive amounts of computational power, especially for the training stage.

Supervised MSS approaches, which currently lead the field, require fully-isolated multi-track recordings for the target sources. Data of this kind are scarce and constrained to few musical repertoires [8] because recording these at high quality without bleeding is expensive. One solution is to synthesize the signals [8–10], however, these datasets may not be fully realistic and may produce domain mismatch. On the other hand, multi-track datasets with source bleeding, where the track corresponding to a source is contaminated by the leakage from other sources, are easier to build, since these may be compiled through a less complex process, and can be recorded in live performances. We observe large multi-track datasets with bleeding for diverse domains in the literature [11–14], therefore, dedicated MSS systems to be trained with these would be beneficial. In fact, MSS in the presence of bleeding has recently gained interest: a dedicated leaderboard for this problem – albeit in a slightly different context than here – has been included in the Music Demixing Challenge 2023 [15].

In this work, we propose to address the MSS problem for a repertoire that lacks clean isolated tracks: Carnatic Music (CM). The computational analysis of CM has received growing attention in recent years [16]. MSS is a useful pre-processing step in many computational research pipelines on CM. However, researchers use the available models in the literature, typically the pip-installable version of Spleeter [3] – some examples being [17–22] –, which is trained on a large private dataset, presumably including few or no CM examples. Despite not having information on that latter matter, we make the assumption because the 4/5-stem Spleeter models target an instrument arrangement not applicable to CM (vocals, bass, drum, piano, and other), and CM is rarely recorded stem-by-stem in a studio. The domain mismatch between repertories here may hinder the generalization given the unseen instruments and playing/singing techniques. That may also produce a negative effect on the analysis of the separated sources, as well as on further processes such as melody estimation or pattern recognition. Existing works focusing on CM have pointed out the domain mismatch problem for related tasks currently lead by data-driven models [23].

We propose an MSS model to be trained using the Saraga dataset [11] which is, to the best of our knowledge,

[1] https://github.com/MTG/carnatic-separation-ismir23

the largest open dataset for the computational analysis of Indian Art Music (IAM). Saraga comprises multi-track audio data recorded in live performances and is larger in size ($\approx$ 36h) than the rest of the real-audio MSS datasets in the literature. However, in all multi-track audio signals in Saraga, there is bleeding from the rest of the sources. Our goal is to use the real-world data with bleeding in Saraga to train an MSS model for this domain, while proposing a strategy to output clean isolated signals, even though no bleeding-free signals are available for development.

To achieve so, we train a *cold diffusion* model, followed by unsupervised clustering on the resulting output. Cold diffusion has shown promising results in recovering data samples from a given distribution that have been iteratively perturbed, in T steps, using a deterministic signal [24–26]. We apply said process to iteratively convert the amplitude spectrogram of a mixture to that of a target source with bleeding. This yields a separation as good as the target source with bleeding in the training data. To address this issue, we take advantage of all the intermediary cold diffusion steps to further improve the output. In doing so, we rely on the fact that the energy of the target source, which is predominant, will evolve differently throughout the transformation than the energy of the source bleeding. Note that this process is cumbersome in a single-step non-diffusion separation system, given the overlapping between vocal and accompaniment at various time-frequency bins.

The key contribution of this study is an MSS system that can be developed solely using data with bleeding. With regards to that, solely the mixture and the target source containing bleeding from other instruments are required. Given its relevance to the repertoire, we focus on separating the singing voice. Also, the proposed model is adaptable: the user may choose to be more restrictive with interferences – at the expense of loss of vocal quality – or vice versa. We put special emphasis on being able to characterize the ubiquitous instruments in CM, to reliably remove the interferences from the singing voice. In a computational musicology context, that would improve the musicologically-relevant research done on the separated vocal signal.

2. METHOD

Our separation pipeline assumes the existence of m, the audio signal of the mixture, and the target source with bleeding s^b which is contained in the mixture, while we may not have the remaining sources at hand. In our case, s^b is the singing voice with source bleeding. We present a two-step method to estimate the isolated source $\hat{s}$ by only having m and s^b during training, and solely m during inference.

(1) Cold diffusion process: we aim at running a cold diffusion process to recursively convert the magnitude spectrogram of a mixture M into the magnitude spectrogram of the singing voice with bleeding $\hat{S}^b$.

(2) Unsupervised mask estimation: Note that step **(1)** can only yield estimations as good as the source with bleeding S^b used as ground truth. Toward refining these estimations, we build the final estimation mask by clustering the frequency bins using the entire cold diffusion process to understand how the energy of each bin is evolving during the transformation.

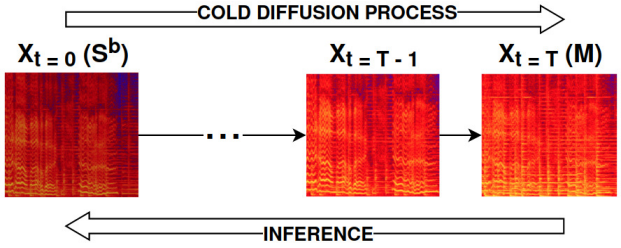

Figure 1. The **spectrogram cold diffusion** transforms, in T steps, a mixture M into a target source with bleeding S^b.

2.1 Feature extraction

2.1.1 Spectrogram cold diffusion

We propose an approach inspired by diffusion models, a class of generative models that define a Markov chain of T steps to iteratively convert samples from a given data distribution into Gaussian noise while learning to conduct the reverse process [27]. The model learns to generate a sample of the given input data distribution from a random sample of noise. Recently, deterministic signals have been successfully used in place of Gaussian noise for the diffusion process [24–26], a technique known as *cold diffusion*.

In [25], the authors apply a transformative cold diffusion process for SVS, using the mixture as the perturbation signal to gradually convert a singing voice to the corresponding mixture, while learning to conduct the reverse process, yielding improved separations for the evaluated model. The process operates in the waveform domain. Here, we propose an updated version of the cold diffusion paradigm in [25] to apply it in time-frequency domain. The cold diffusion process begins at X_0 which is the target data point at inference, in our case S^b, and ends at X_T, in our case M. Let α_t be the perturbation schedule to control the amount of perturbation added at each step and therefore determining the intermediate states of the variable X_t, being t the cold diffusion step. We define α_t as a 1D vector of linearly spaced values from 1 to 0, and of length T. We compute any step in the cold diffusion process as:

$$q_t(X_t|M, X_0) = \alpha_t X_0 + (1 - \sqrt{\alpha_t})M \qquad (1)$$

The process is depicted in Figure 1. In other words, the proposed cold diffusion process gradually converts the amplitude spectrogram of the singing voice with source bleeding into the corresponding mixture, while at inference we aim at reverting said transformation. The use of cold diffusion is motivated by the successful attempts to iteratively transform two-dimensional signals using a diffusion process through a U-Net [24], a well-known network for source separation [2, 3]. Moreover, we can train the model in a supervised fashion, which may lead to more consistent performance than unsupervised procedures. Approaches to

extract features from the spectrogams for clustering [28] are a problem on its own, which in this context may be hindered by source bleeding and the musical and spectral characteristics of CM instruments, e.g. violin or tanpura.

Note that given Eq. 1, the singing voice stays predominant, while the accompaniment gradually increases – in the direction of the cold diffusion process – or decreases – in the direction of the inference process–. This equation is also aimed at amplifying the energy difference throughout the steps between the singing voice and accompaniment frequency bins, and that explains why X_0 and M have different trajectories assigned. The weighting $(1 - \sqrt{\alpha_t})$ applied to the mixture M ensures larger steps at the start of the inference process, while more fine-grained estimations are performed at the latter steps [27], aiming at obtaining more refined separation outputs. Note also that the expected inference input of a singing voice extraction model – in our case corresponding to X_T – is a mixture. Given the expressions in Eq. 1, the perturbation M ensures that $X_T = M$, otherwise the said condition is not given.

2.1.2 Reverse process

The reverse process iteratively removes the deterministic perturbation, aiming at reaching S^b receiving the corresponding mixture M as input. We directly chain the model estimations, so that the model input at a particular step t is the raw prediction of the model at the previous step $t + 1$ (note that the reverse process begins from step T to reach step 1). Therefore, given a trained model D with parameters θ, the reverse process can be defined as follows:

$$r_t(\hat{X}_{t-1}|X_t) = D_\theta(X_t, t) \tag{2}$$

This process is iteratively performed for $t = [T, T - 1..., 1]$, using M as input corresponding to X_T.

2.1.3 Training algorithm

We aim at training a model that learns a mask K_t for each diffusion step t so that $X_t * K_t = \hat{X}_{t-1}$. For each t, we predict a different mask that transforms the signal into the next step in the reverse process until we reach $\hat{X}_0$, which ideally is as close as possible to S^b. Given Eq. 1, we effectively optimize the model using the following objective [27]:

$$L(\theta) = \|X_{t-1} - D_\theta(q_t(X_t|M, X_0), t)\|^2 \tag{3}$$

where X_{t-1} is the known next step in the reverse process computed following $q_t(X_t|M, X_0)$, whereas the model D_θ predicts the next step $\hat{X}_{t-1}$ based on $q_t(X_t|M, X_0)$, the step t, the mixture M, and the input of the cold diffusion process X_0, corresponding to S^b.

We employ a U-Net to learn the reverse process, which has been shown useful for the problem of MSS [2]. We use a U-Net with 7 levels of depth and 4 residual blocks at each level. Both frequency and time dimensions are encoded and then expanded by a factor of 2. The last layer is a sigmoid in order to output the mask K_t of values $\in [0, 1]$, which is multiplied by the input X_t to get $\hat{X}_{t-1}$. We estimate masks instead of spectrograms to obtain a more consistent and linear evolution of the bins energy. Estimating spectrograms may lead to unstable removal of accompaniment, which adds complexity to the proposed approach. To inform the network about the current diffusion step t in the reverse process, we encode it using a 16-dimension sinusoidal positional vector [29]. Said embedding is processed through two dense layers of 64 units. Next, the embedded t is projected to the corresponding channel size at each level of depth of the U-Net and added to the input of each residual block. We inject the time-step embedding to all residual blocks in the encoder, decoder, and bottleneck.

2.1.4 Inference

In standard diffusion models, the output of the last step is considered to be the cleanest signal. We argue that we can achieve better separation by studying how the time-frequency bins evolve throughout the inference process.

We run inference using the trained D_θ to automatically convert an input M to a predicted $\hat{S}^b$ while capturing and stacking all intermediate representations, in order, in a feature matrix $\hat{X}_{T,...,0}$. These features are sized $I \times J \times T$, where I is time size, J number of frequency bins, and T is the number of cold diffusion steps, and represent how the iterative transformation of the magnitude spectrogram of M changes over the cold diffusion steps until reaching predicted $\hat{S}^b$. We normalize the features by dividing all $\hat{X}^{(i,j)}_{T,...,0}$ – being (i, j) the coordinates of a given frequency bin in $\hat{X}_{T,...,0}$ – by $max(\hat{X}^{(i,j)}_{T,...,0})$. Therefore, the energy vectors are studied on the same scale.

2.2 Unsupervised mask estimation

The final mask estimation is performed on top of the cold diffusion feature matrix $\hat{X}_{T,...,0}$, as seen in Figure 2. Existing works use diffusion models to generate features or embeddings for downstream tasks [30], however, to our best knowledge, this is the first attempt to use an entire diffusion process rather than relying only on the output signal.

Note that in the proposed cold diffusion paradigm, we iteratively convert the accompaniment into bleeding – much lower in presence but not removed –, while preserving the cleanest possible voice. Therefore, the energy of the time-frequency bins $\hat{X}^{(i,j)}_{T,...,0}$ across the diffusion steps fluctuates less for the voice than for the accompaniment, which is iteratively lowered by the model. To this end, we propose to cluster the frequency bins based on the evolution of these in $\hat{X}_{T,...,0}$. Clustering techniques have been previously used in a separation context [28, 31, 32], aiming at grouping the components belonging to the same source.

We use K-means clustering to automatically create groups of frequency bins associated with sources, given the computed features $\hat{X}_{T,...,0}$. For example, if a binary separation mask is desired, one may use two clusters and multiply by 0 the clustered bins belonging to the accompaniment, while leaving the rest unchanged. For a soft mask, we consider more than two clusters, and the bins classified in the middle clusters may be shared between the singing voice and other sources, as seen in Figure 2, where we use three clusters. In our case, ideally, the cold diffusion process iteratively reduces the energy of accompaniment bins

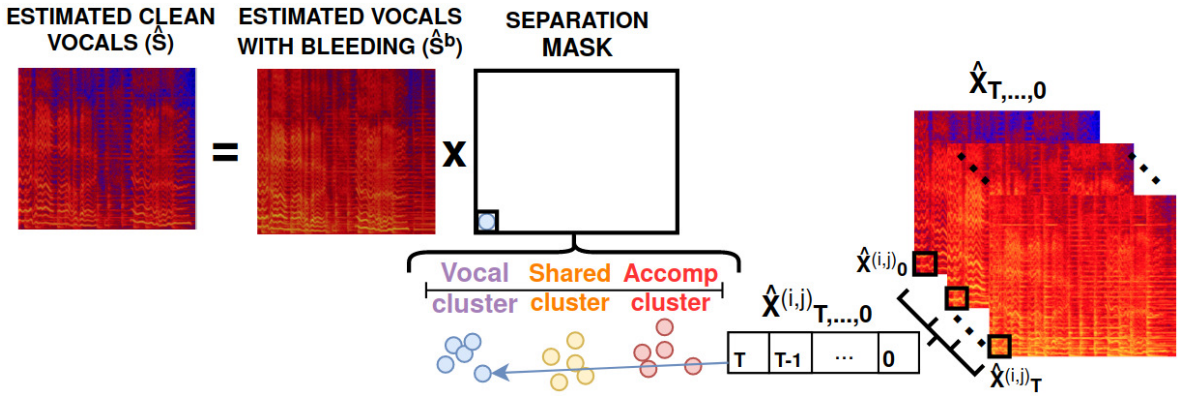

Figure 2. The **unsupervised mask estimation** step clusters the frequency bins given vector $\hat{X}_{T,\ldots,0}$, which stores the evolution from mixture M to the predicted source with bleeding $\hat{S}^b$. The example in this figure uses $C = 3$, being C the number of clusters. The cluster with centroid with lower value is considered the *accompaniment cluster* and assigned 0 in the mask and removed, while the furthest cluster to that is the *vocal cluster* and assigned value 1, so values are left untouched. The bins in the *shared clusters* (we have more than one shared cluster for $C > 3$) are weighted given w^F. Therefore, the user can navigate, given parameter F, through the interference/artifacts trade-off. Using larger C (i.e. considering more clusters) delivers a more granular masking.

while preserving the singing voice. Therefore, the features per bin $\hat{X}^{(i,j)}_{T,\ldots,0}$ have higher values for those corresponding to the singing voice. In this case, the centroid of the closest cluster to the voice centroid has the largest L1 norm. We can then sort the clusters by the L1 norm of the centroid.

Having the clusters ordered, a weight $\in [0, 1]$ must be assigned to each cluster to create the soft mask. Rather than normalizing the cluster centroids, we discover that it is desirable to assign a balanced weighting to the clusters. Thus, we define w, a 1D array linearly spaced values $\in [0, 1]$ of length C, which is the number of clusters. Note that 0 and 1 are both included to directly give a weight of 0 to the *accompaniment cluster* and 1 to the *vocal cluster*, which are the two furthest clusters. Now let F be an integer representing weight factor that is used to control how restrictive we want to be with the intermediate clusters. Given w and F, we compute the final cluster weight array as w^F. For an $F > 1$, we are being more restrictive, especially with the clusters closer to the accompaniment one, and the bigger we set F, the more restrictive we are. When evaluating the clustering, we experiment with various parameter configurations. However, C and F may also be given by the users to control the trade-off between interference and artifacts depending on their needs. Intuitively, the more clusters are considered and removed, we obtain an output with less interference from other sources, at the expense of a loss of quality from the target source.

To take advantage of the first separation run given by the cold diffusion process, we multiply the final mask with the last step of the inference process $\hat{X}_0$, or $\hat{S}^b$. Preliminary results confirmed that this is beneficial over masking the input mixture, and it does not imply added computational expense since $\hat{X}_0$ is contained in the features $\hat{X}_{T,\ldots,0}$.

Note that the use of the cold diffusion process allows the development of differentiable operations for estimating the final separation mask in the context of bleeding. We observed that clustering is not feasible when using a one-step prediction, e.g. two spectrograms do not yield enough information to study the energy change between a vocal and an accompaniment frequency bin.

3. EXPERIMENTS

3.1 Experimental setup

We perform our experimentation using $q_t(X_t|M, X_0)$ with $T = 8$. Generative diffusion typically uses larger T, e.g. 1000 [27]. Using large values for T in this context produces two consecutive steps in the process practically identical, and the optimization of the model becomes extremely complex. We compute the STFT of m and s^b with window size 1024 and hop 256, at a sampling rate of 22050Hz. We use ADAM optimizer with a learning rate of 2^{-4} and batch size of 8, and we run the training process for 1M steps.

The larger in time the input mixture spectrograms are, the more bins to cluster for the final mask estimation. While using an oversized spectrogram may lead to a complex clustering problem given the variations in playing intensity, few points may hinder the estimation of the clusters. Given the improvisatory nature of CM, we propose to use chunks of 3 seconds in order to be robust to the recurrent changes in intensity and dynamics of the performers.

Since we operate on magnitude spectrograms, we require the phase information to reconstruct the estimated audio signals. Here we reuse the phase from m, which is not ideal but it is fast and broadly used in the MSS [2].

3.1.1 Objective evaluation

We evaluate the models on a real-audio test set we record for the purpose of this work. It includes $\approx 2h$ of music and two different singers (male and female). Bleeding-free tracks for violin, mridangam, and tanpura are also available. We split the tracks into chunks of 30s, slightly mod-

ifying the mixing parameters to enrich the diversity in the dataset. The tracks are mixed with the assistance of an audio engineer. The testing set is made available for reproducibility and further MSS research.

MSS is commonly evaluated objectively using the `BSS_Eval` metrics [33]: (1) SDR: overall quality, (2) SIR: intrusiveness of the other sources in the estimated source, and (3) SAR: quality of the estimated source. For particular music genres SDR may not correlate with perceptual quality [34–37]. Thus, we run a subjective evaluation in which we contrast the two dimensions captured by the objective evaluation: interference removal (SIR) and signal quality (SAR). This is a common experimental setup in perceptual evaluation of MSS [38]. Therefore, we put more emphasis on these metrics on our objective evaluation as well, rather than comparing solely SDR. Note that our method allows for selecting the desired level of interference at the expense of signal artifacts. Therefore, we aim at covering two scenarios: creative tasks e.g. practicing or mixing, and analysis tasks, e.g. melody estimation.

In a first experiment we compare our system using three different configurations with a baseline U-Net model trained with raw Saraga as regular MSS models are. A second experiment is intended to compare our system with: **(1)** our cold diffusion model skipping the unsupervised clustering mask estimation, and **(2)** Spleeter [3], a widely used model in the literature, also in computational analysis works for CM. We include this comparison considering that Spleeter is trained using a much larger dataset with an unknown distribution. To the best of our knowledge, no Carnatic-specific separation models are available in the literature. In the latter experiment we report the absolute SIR and SAR difference of our models w.r.t. the alternatives aiming at providing an intuitive comparison in terms of interference removal and vocal signal quality. We compute the global MSS metrics [15] for all testing samples using the latest `museval` version [39], and we compute the median to be robust to extreme cases in the testing set.

3.1.2 Perceptual evaluation

Despite the efforts to enhance the variety within our testing set, it is restricted in size and all recordings are obtained from the same source. This is added to the fact that the objective metrics in [40] may not always correlate with the perceptual quality of MSS estimations [34]. For these reasons, we run a perceptual test with subjects including samples from the non-multi-track recordings in Saraga ($\approx$ 17h) – which were not included in the training set for our models, and ensuring there are no overlapping artists – and from the private collection of the Dunya database [41]. We first randomly sample 50 recordings from the said data collections, and we extract the singing voice from a randomly selected 30s chunk for each recording. Using the mixture as reference, we manually collect 6 examples from the batch of separations ensuring that the test includes different audio qualities, gender balance, and tonic diversity.

We design an online survey based on the MUSHRA framework [42]. We request the participants to rate, from 1 to 5, the vocal quality and the intrusiveness of other sources *separately*. The participants are shown the mixture as a reference and two stimuli: our system with $C = 5$ and $F = 4$, and Spleeter. The test includes a tutorial stage with examples – these are not shown during the actual test and are not passed through any of the evaluated models – to make sure the participants have the difference between distortion and intrusiveness from other sources clear. We randomize the order of the stimuli at each example, to prevent the order from having an impact on the ratings. The proposed subjective evaluation follows closely the ITU-T P.835. We include a short survey in the test to collect information on the expertise of the subjects on MSS and CM.

For each testing example, we compute the mean and standard deviation of all rankings. We finally report the mean and standard deviation over the 6 excerpts. The deviation serves as an indicator of the sparsity of the opinions.

	C	***F***	**SDR**	**SIR**	**SAR**
Baseline	-	-	**6.10**	10.71	**8.16**
Ours	3	1	5.88	13.69	6.72
Ours	4	2	5.12	14.94	5.57
Ours	5	3	4.56	**15.84**	4.68

Table 1. Comparison of the baseline with three configurations for our system. C is no. of clusters and F the weight factor. Results are given in dB.

3.2 Results

3.2.1 Objective results

We first compare, on our testing set, our system with $T = 8$ and three different cluster configurations with the baseline U-Net separation model. Results are shown in Table 1. The baseline system is more prone to leak other sources in the estimated vocals given the source bleeding in the training data, while it better preserves the quality of the target source. On the other hand, our system further eliminates the CM instrumentation from the input signal. However, additional masking comes with a drawback and especially in the case of CM where all instruments are pitched and tuned in the same tonic. That produces an important overlap, especially between vocals and violin. Therefore, by removing more interference, we are penalizing the quality of the singing voice.

Related to the latter observation, we confirm the adaptability of our system. The more clusters we consider and remove, we achieve better interference removal at the expense of a loss of vocal quality. However, as seen in Table 1, this is translated into worse SDR values. In the perceptual evaluation we study how these metrics correlate with the perceived quality of the estimations.

In Table 2 we report the difference in SIR and SAR (denoted, respectively, SIRd and SARd), first between two versions of our system (with and without clustering), and second between our system and Spleeter. Using roughly all tested configurations, our system is able to outperform the

	No clustering		Spleeter [3]	
	SIR	**SAR**	**SIR**	**SAR**
	9.39	10.28	14.21	10.95
Comparison of our system with *T=8*				
Config		**vs. No clustering**		**vs. Spleeter** [3]
C	***F***	**SIRd**	**SARd**	**SIRd**	**SARd**
2	1	+4.70	**-3.43**	+0.14	**-4.09**
3	1	+4.31	-3.56	+0.52	-4.22
3	2	+5.46	-4.49	+0.64	-5.16
4	2	+5.55	-4.71	+0.72	-5.38
4	3	+6.41	-5.53	+1.60	-6.20
5	2	+5.61	-4.69	+0.78	-5.35
5	3	+6.45	-5.59	+1.63	-6.26
5	4	**+7.14**	-6.25	**+2.32**	-6.91

Table 2. SIR and SAR difference of our full system with (1) our system with no unsup. mask estimation and (2) Spleeter. Results given in dB, + indicate that we improve. On top, we provide the absolute metrics of the alternatives for reference. C is no. of clusters and F weight factor.

alternatives in terms of interference removal, better characterizing and cleaning the Carnatic accompaniment from the singing voice, suggesting that we are taking advantage of the in-domain data despite the bleeding. Note also the SIR improvement – more than 4dB in the worst case – that the unsupervised masking provides on top of the last step of the cold diffusion model, which can only estimate, at most, the vocals with bleeding. That is the problem of using data with bleeding for training supervised MSS systems.

However, our system tends to perform worse in signal quality. This may be given by frequency components of the singing voice that are being removed while performing the unsupervised mask estimation, especially those living in the bins shared with other sources. On the other hand, Spleeter maintains a more complete singing voice despite being more prone to interference. We perceptually note that our estimations are *dryer*, while Spleeter is able to better capture components such as reverb and high-frequency details. This may be explained by the much larger training dataset comprising several different vocal styles and effects. In our case, given the proposed schedule and diffusion steps, these components may be partially living on the shared clusters and therefore negatively affected as we use a more restrictive parametrization.

Note the small difference in SAR between our system with no clustering-based masking and Spleeter. Said observation suggests that the cold diffusion process preserves the vocal quality roughly as Spleeter achieves so. That may also explain why masking the last cold diffusion step $\hat{X}_0$ provides an improved output over masking the mixture M.

3.2.2 Perceptual results

We run the MUSHRA test on 25 subjects. From the population, $\approx 44\%$ of the subjects have mid-to-high expertise in MSS, while $\approx 48\%$ have listened CM at least once.

	Mean Opinion Scores (MOS)	
	Vocal quality	**Vocal isolation**
Ours (*C=5*, *F=4*)	2.80 ± 0.29	**3.72** ± 0.31
Spleeter [3]	**3.73** ± 0.17	1.97 ± 0.19

Table 3. Comparison between our system and Spleeter [3] on a perceptual test. Min=1 / Max=5, the higher the better.

The results of the MUSHRA test (see Table 3) on the intrusiveness of other sources – or how well the vocals are isolated – present a notable correlation with the SIRd in Table 2, suggesting that our model is able to better eliminate the Carnatic instruments from the separated singing voice. Another relevant aspect that we observe is that while Spleeter is still leading on source quality, the scores are more balanced between both models than what the SARd metrics in Table 2 suggest. That may be an indicator that the singing voice components erroneously removed by our model – which notably penalize metrics-wise – are not notably perceivable to the naked ear. All deviations of participant rankings per example are < 1, suggesting that generally there is a disagreement of 1 point at most. Additionally, we run the Wilcoxon signed-rank test for paired data on each example, observing for all cases a p-value < 0.05, indicating that the subject ratings were not given randomly.

4. CONCLUSIONS

We present a system that uses an entire cold diffusion process as features to perform singing voice separation when no isolated ground-truth sources are available, and we solely have the mixture and the target source with bleeding at hand for training. The cold diffusion process, which iteratively transforms a mixture into the target source with bleeding, allows for unsupervised clustering to build the final separation masks. We run our approach on the Saraga dataset, a large Carnatic collection of multi-track audio with bleeding. Despite being trained solely using these data, our model is able to better eliminate the Carnatic instruments from the singing voice than Spleeter, the most commonly used model in computational research for this repertoire, which is trained on a much larger private dataset of clean signals. Albeit the source separation metrics suggest that our system performs worse in terms of vocal distortion, perceptual tests on a dedicated test set suggest that the proposed system trained with noisy and considerably fewer data than Spleeter is competitive with the said system. This will allow to scale up our system since new in-domain data with bleeding are easier to compile than clean data, especially for under-represented music cultures.

As further research, we propose to investigate different schedules, while exploring more sophisticated clustering techniques, aiming at improving source distortion. We also aim at running the proposed pipeline for the other available instrument tracks in Saraga: violin and mridangam.

5. ACKNOWLEDGEMENTS

This work was carried out under the projects Musical AI - PID2019-111403GB-I00/AEI/10.13039/501100011033 funded by the Spanish Ministerio de Ciencia, Innovación y Universidades (MCIU) and the Agencia Estatal de Investigación (AEI). We would also like to acknowledge the 25 subjects that took the perceptual test and Xavier Lizarraga for the assitance on mixing the testing set.

6. REFERENCES

[1] Y. Luo and J. Yu, "Music source separation with band-split RNN," 2022. [Online]. Available: http://arxiv.org/abs/2209.15174

[2] A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde, "Singing voice separation with deep U-Net convolutional networks," in *Proc. of the 18th Int. Society for Music Information Retrieval Conf. (ISMIR),* Suzhou, China, 2017, pp. 745–751.

[3] R. Hennequin, A. Khlif, F. Voituret, and M. Moussallam, "Spleeter: a fast and efficient music source separation tool with pre-trained models," *Journal of Open Source Software*, pp. 1–4, 2020.

[4] D. Stoller, S. Ewert, and S. Dixon, "Wave-U-Net: A multi-scale neural network for end-to-end audio source separation," in *Proc. of the 19th Int. Society for Music Information Retrieval Conf. (ISMIR),* Paris, France, 2018, pp. 334–340.

[5] A. Défossez, N. Usunier, L. Bottou, and F. Bach, "Demucs: Deep Extractor for Music Sources with extra unlabeled data remixed," 2019. [Online]. Available: http://arxiv.org/abs/1909.01174

[6] A. Défossez, "Hybrid spectrogram and waveform source separation," 2021. [Online]. Available: http://arxiv.org/abs/2111.03600

[7] M. Kim, W. Choi, J. Chung, D. Lee, and S. Jung, "KUIELab-MDX-Net: a two-stream neural network for music demixing," 2021. [Online]. Available: http://arxiv.org/abs/2111.12203

[8] M. Miron, J. Janer, and E. Gómez, "Generating data to train convolutional neural networks for classical music source separation," in *Proc. of the 14th Sound and Music Computing Conf.,* Espoo, Finland, 2017, pp. 227–233.

[9] 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. of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2019.

[10] S. Sarkar, E. Benetos, and M. Sandler, "EnsembleSet: a new high quality synthesised dataset for chamber ensemble separation," in *Proc. of the 23rd Int. Conf. on Music Information Retrieval (ISMIR),* Bengaluru, India, 2022.

[11] A. Srinivasamurthy, S. Gulati, R. C. Repetto, and X. Serra, "Saraga: Open Datasets for Research on Indian Art Music," *Empirical Musicology Review*, 2020.

[12] E. Gómez, M. Grachten, A. Hanjalic, J. Janer, S. Jordà, C. F. Julià, C. Liem, A. Martorell, M. Schedl, and G. Widmer, "PHENICX: Performances as Highly Enriched and Interactive Concert Experiences," in *Proc. of the 10th Sound and Music Computing Conf. (SMC),* Stockholm, Sweden, 2013.

[13] O. Mayor, Q. Llimona, M. Marchini, P. Papiotis, and E. M. Gómez, "repoVizz: a framework for remote storage, browsing, annotation, and exchange of multimodal data," in *Proc. of the ACM Int. Conf. on Multimedia (MM'13)*, 2013.

[14] T. Prätzlich, M. Müller, B. W. Bohl, and J. Veit, "Freischütz Digital: Demos of audio-related contributions," in *Demos and Late Breaking News of the Int. Society for Music Information Retrieval Conf. (ISMIR),* Málaga, Spain, 2015.

[15] Y. Mitsufuji, G. Fabbro, S. Uhlich, F.-R. Stöter, A. Défossez, M. Kim, W. Choi, C.-Y. Yu, and K.-W. Cheuk, "Music Demixing Challenge 2021," *Frontiers in Signal Processing*, vol. 1, 2022.

[16] G. Tzanetakis, "Computational ethnomusicology: A music information retrieval perspective," in *Proc. of the 40th Int. Computer Music Conf.,* Athens, Greece, 2014.

[17] T. Nuttall, G. Plaja-Roglans, L. Pearson, and X. Serra, "The matrix profile for motif discovery in audio-an example application in Carnatic music," in *Proc. of the 15th Int. Symposium on Computer Music Multidisciplinary Research (CMMR),* Tokyo, Japan.

[18] M. Clayton, P. Rao, N. Shikarpur, S. Roychowdhury, and J. Li, "Raga classification from vocal performances using multimodal analysis," in *Proc. of the 23rd Int. Conf. on Music Information Retrieval (ISMIR),* Bengaluru, India, 2022.

[19] S. John, M. Sinith, S. R.S., and L. P.P., "Classification of indian classical carnatic music based on raga using deep learning," *IEEE Recent Advances in Intelligent Computational Systems (RAICS)*, 2020.

[20] N. Shikarpur, A. Keskar, and P. Rao, "Computational analysis of melodic mode switching in raga performance," in *Proc. of the 22th Int. Society for Music Information Retrieval Conf. (ISMIR),* Online, 2021, pp. 657–664.

[21] R. M.A., V. T.P., and P. Rao, "Structural segmentation of dhrupad vocal bandish audio based on tempo," in *Proc. of the 21th Int. Society for Music Information Retrieval Conf. (ISMIR),* Montréal, Canada, 2020, pp. 678–684.

[22] D. P. Shah, N. M. Jagtap, P. T. Talekar, and K. Gawande, "Raga recognition in indian classical music using deep learning," *Artificial Intelligence in Music, Sound, Art and Design*, pp. 248–263, 2021.

[23] G. Plaja-Roglans, T. Nuttall, L. Pearson, X. Serra, and M. Miron, "Repertoire-specific vocal pitch data generation for improved melodic analysis of Carnatic music," *Transactions of the Int. Society for Music Information Retrieval Conf. (TISMIR)*, 2023.

[24] A. Bansal, E. Borgnia, H.-M. Chu, J. S. Li, H. Kazemi, F. Huang, M. Goldblum, J. Geiping, and T. Goldstein, "Cold diffusion: Inverting arbitrary image transforms without noise," 2022. [Online]. Available: http://arxiv.org/abs/2208.09392

[25] G. Plaja-Roglans, M. Miron, and X. Serra, "A diffusion-inspired training strategy for singing voice extraction in the waveform domain," in *Proc. of the 23rd Int. Conf. on Music Information Retrieval (ISMIR),* Bengaluru, India, 2022.

[26] H. Yen, F. G. Germain, G. Wichern, and J. Le Roux, "Cold diffusion for speech enhancement," 2022. [Online]. Available: https://arxiv.org/abs/2211.02527

[27] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. of the 33th Advances in Neural Information Processing Systems (NeurIPS),* Online, 2020, pp. 6840–6851.

[28] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP),* Shangai, China, 2016, pp. 31–35.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. of Advances in Neural Information Processing Systems (NeurIPS),* Long Beach, USA, 2017, pp. 5999–6009.

[30] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. of the Conf. on Computer Vision and Pattern Recognition (CVPR),* New Orleans, Louisiana, USA, 2021.

[31] Y. Luo, Z. Chen, J. R. Hershey, J. Le Roux, and N. Mesgarani, "Deep clustering and conventional networks for music separation: Stronger together," *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP),* New Orleans, USA, pp. 61–65, 2017.

[32] K. Chen, G. Wichern, F. G. Germain, and J. L. Roux, "Pac-hubert: Self-supervised music source separation via primitive auditory clustering and hidden-unit bert," 2023. [Online]. Available: http://arxiv.org/abs/2304.02160

[33] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 4, no. 14, pp. 1462–1469, 2006.

[34] E. Cano, D. Fitzgerald, and K. Brandenburg, "Evaluation of quality of sound source separation algorithms: Human perception vs quantitative metrics," in *Proc. of the 24th European Signal Processing Conf. (EUSIPCO),* Budapest, Hungary. IEEE, 2016, pp. 1758–1762.

[35] U. Gupta, E. Moore, and A. Lerch, "On the perceptual relevance of objective source separation measures for singing voice separation," in *Proc. of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2015.

[36] H. Wierstorf, D. Ward, R. Mason, E. M. Grais, C. Hummersone, and M. D. Plumbley, "Perceptual evaluation of source separation for remixing music," *Journal of the Audio Engineering Society (AES)*, 2017.

[37] E. Gusó, J. Pons, S. Pascual, and J. Serrà, "On loss functions and evaluation metrics for music source separation," *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 306–310, 2022.

[38] A. Défossez, N. Usunier, L. Bottou, and F. Bach, "Music source separation in the waveform domain," 2019. [Online]. Available: http://arxiv.org/abs/1911.13254

[39] F.-R. Stöter, A. Liutkus, D. Samuel, L. Miner, and F. Voituret, "sigsep/sigsep-mus-eval: museval 0.4.0," Feb. 2021.

[40] F. R. Stöter, A. Liutkus, and N. Ito, "The 2018 Signal Separation Evaluation Campaign," *Lecture Notes in Computer Science*, vol. 10891, pp. 293–305, 2018.

[41] A. Porter, M. Sordo, and X. Serra, "Dunya: A system for browsing audio music collections exploiting cultural context," in *Proc. of the 14th Int. Society for Music Information Retrieval Conf. (ISMIR),* Curitiba, Brazil, 2013.

[42] M. Schoeffler, S. Bartoschek, F.-R. Stöter, M. Roess, S. Westphal, B. Edler, and J. Herre, "webMUSHRA — a comprehensive framework for web-based listening tests," *Journal of Open Research Software*, vol. 6, no. 1, 2018.