Latent Autoregressive Source Separation

Emilian Postolache[*1], **Giorgio Mariani**[*1], **Michele Mancusi**[*1], **Andrea Santilli**[1],
Luca Cosmo[†2,3], **Emanuele Rodolà**[†1]

[1] Sapienza University of Rome, Italy
[2] Ca' Foscari University of Venice, Italy
[3] University of Lugano, Switzerland
postolache@di.uniroma1.it, mariani@di.uniroma1.it, mancusi@di.uniroma1.it

Abstract

Autoregressive models have achieved impressive results over a wide range of domains in terms of generation quality and downstream task performance. In the continuous domain, a key factor behind this success is the usage of quantized latent spaces (e.g., obtained via VQ-VAE autoencoders), which allow for dimensionality reduction and faster inference times. However, using existing pre-trained models to perform new non-trivial tasks is difficult since it requires additional fine-tuning or extensive training to elicit prompting. This paper introduces *LASS* as a way to perform vector-quantized *Latent Autoregressive Source Separation* (i.e., de-mixing an input signal into its constituent sources) without requiring additional gradient-based optimization or modifications of existing models. Our separation method relies on the Bayesian formulation in which the autoregressive models are the priors, and a discrete (non-parametric) likelihood function is constructed by performing frequency counts over latent sums of addend tokens. We test our method on images and audio with several sampling strategies (e.g., ancestral, beam search) showing competitive results with existing approaches in terms of separation quality while offering at the same time significant speedups in terms of inference time and scalability to higher dimensional data.

Introduction

Autoregressive models have achieved impressive results in a plethora of domains ranging from natural language (Brown et al. 2020) to densely-valued domains such as audio (Dhariwal et al. 2020) and vision (Razavi, van den Oord, and Vinyals 2019; Esser, Rombach, and Ommer 2021), including multimodal joint spaces (Ramesh et al. 2021; Yu et al. 2022). In the dense setting, it is typical to train autoregressive models over discrete latent representations obtained through the quantization of continuous data, possibly using VQ-VAE autoencoders (van den Oord, Vinyals, and Kavukcuoglu 2017). This way, generating higher resolution samples while simultaneously reducing inference time is possible. Additionally, the learned latent representations are useful for downstream tasks (Castellon, Donahue, and Liang 2021). However, in order to perform new non-trivial tasks, the standard practice is to fine-tune the model or, in alternative, elicit prompting by scaling training (Wei et al. 2021; Sanh et al. 2022). The former is usually the default option, but it requires additional optimization steps or modifications to the model. The latter is challenging on non-trivial tasks, especially in domains different from natural language (Yang et al. 2022; Hertz et al. 2022).

This paper aims to tackle one of such tasks, namely *source separation*, leveraging existing vector-quantized autoregressive models without requiring any gradient-based optimization or architectural modifications. The task of separating two or more sources from a mixture signal has recently received much attention following the success of deep learning, especially in the audio domain, ranging from speech (Dovrat, Nachmani, and Wolf 2021), music (Défossez 2021), and universal source separation (Wisdom et al. 2021; Postolache et al. 2022). Although not as prominent as its audio counterpart, image source separation has been addressed in literature (Halperin, Ephrat, and Hoshen 2019). Most successful approaches use explicit supervision to achieve notable results (Luo and Mesgarani 2019; Défossez et al. 2019), or leverage large-scale unsupervised regression (Wisdom et al. 2020).

We propose a generative approach to perform source separation via autoregressive prior distributions trained on a latent VQ-VAE domain (when class information is used, the approach is weakly supervised; otherwise, it is unsupervised). A non-parametric sparse likelihood function is learned by counting the occurrences of latent mixed tokens with respect to the sources' tokens, obtained by mapping the data-domain sum signals and the relative addends via the VQ-VAE. This module is not invasive, neither for the VQ-VAE nor for the autoregressive priors, given that the representation space of the VQ-VAE does not change while learning the likelihood function. Finally, the likelihood function is combined with the estimations of the autoregressive priors at inference time via the Bayes formula, resulting in a posterior distribution. The separations are obtained from the posterior distributions via standard discrete samplers (e.g., ancestral, beam search). We call our method *LASS (Latent Autoregressive Source Separation)*.

Our contributions are summarized as follows:

- We introduce *LASS* as a Bayesian inference method for source separation that can leverage existing pre-trained

[*] Equal contribution. Listing order is random.
[†] Shared last authorship.

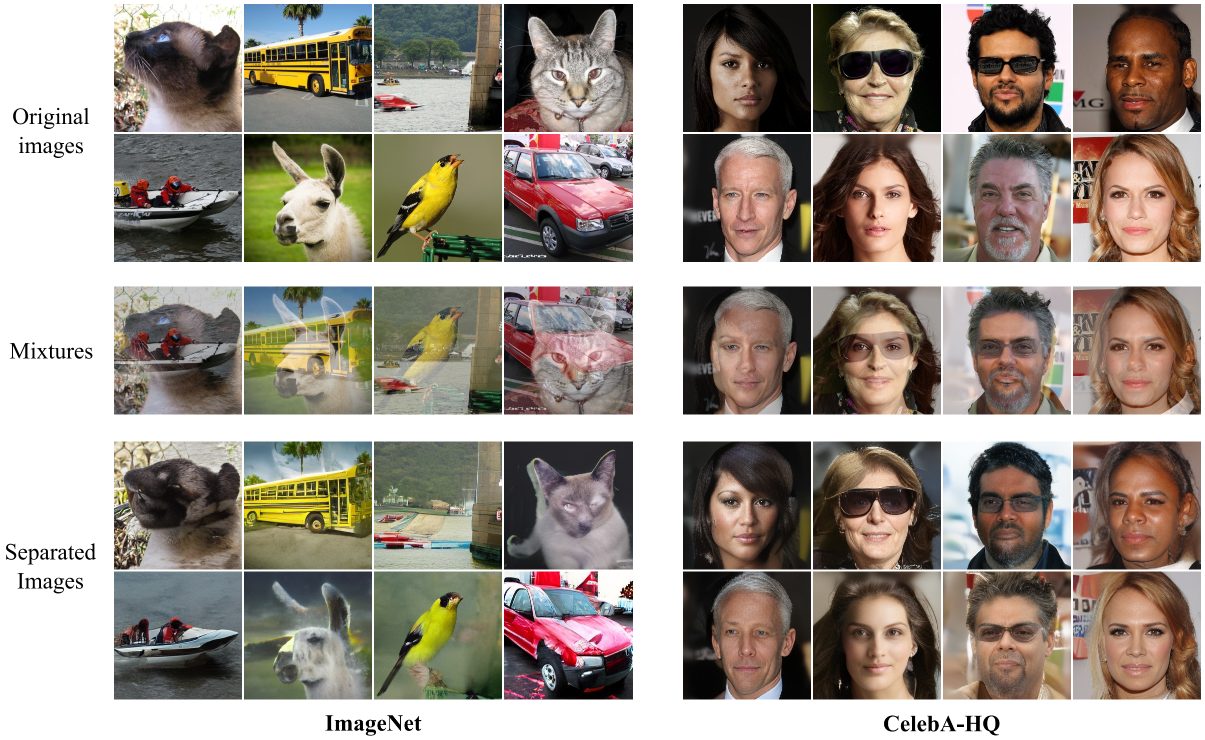

Figure 1: 256x256 separations obtained with *LASS* using pre-trained autoregressive models. Left: class-conditional ImageNet. Right: unconditional CelebA-HQ.

autoregressive models in quantized latent domains.

- We experiment with *LASS* in the image domain and showcase competitive results at a significantly smaller cost in inference time with respect to competitors on MNIST and CelebA (32×32). We also showcase qualitative results on ImageNet (256×256) and CelebA-HQ (256×256), thanks to the scalability of *LASS* to pre-trained models. To the best of our knowledge, this is the first method to scale generative source separation to higher resolution images.
- We experiment with *LASS* in the music source separation task on the Slakh2100 dataset. *LASS* obtains performance comparable to state-of-the-art supervised models, with a significantly smaller cost in inference and training time with respect to generative competitors.

Related Work

The problem of source separation has been classically tackled in an unsupervised fashion under the umbrella term of *blind source separation* (Comon 1994; Hyvärinen and Oja 2000; Huang et al. 2012; Smaragdis et al. 2014). In this setting, there is no information regarding the sources to be separated from a mixture signal. As such, these methods rely on broad mathematical priors such as source independence (Hyvärinen and Oja 2000) or repetition (Rafii and Pardo 2012) to perform separation. With the advent of deep learning, most prominent methods for source separation can be classified as regression-based or generative-based methods.

Regression-Based Source Separation

In this setting, a mixture is fed to a parametric model (i.e., a neural network) that outputs the separated sources. Training is typically performed in a supervised manner by matching the estimated separations with the ground truth sources with a regression loss (e.g., $\mathcal{L}_1$ or $\mathcal{L}_2$) (Gusó et al. 2022). Supervised regression has been applied to image source separation (Halperin, Ephrat, and Hoshen 2019), but it has been mainly investigated in the audio domain, where two approaches are prevalent: the mask-based approach and the waveform approach. In the mask-based approach, the model performs separation by applying estimated masks on mixtures, typically in the STFT domain (Roweis 2000; Uhlich, Giron, and Mitsufuji 2015; Huang et al. 2014; Nugraha, Liutkus, and Vincent 2016; Liu and Yang 2018; Takahashi, Goswami, and Mitsufuji 2018). In the waveform approach, the model outputs the estimated sources directly in the time domain to overcome phase estimation, which is required when transforming the signal from the STFT domain to the waveform domain (Lluís, Pons, and Serra 2019; Luo and Mesgarani 2019; Défossez et al. 2019).

Generative Source Separation

Following the success of deep generative models (Goodfellow et al. 2014; Kingma and Welling 2014; Ho, Jain, and Abbeel 2020; Song et al. 2021), a new class of generative source separation methods is gaining prominence. This setting emphasizes the exploitation of broad generative models (especially pre-trained ones) to solve the separation task without needing a specialized architecture (as with regression-based models).

Following early work on deep generative separation based on GANs (Subakan and Smaragdis 2018; Kong et al. 2019; Narayanaswamy et al. 2020), Jayaram and Thickstun (2020) propose the generative separation method BASIS in the image setting using score-based models (Song and Ermon 2019) (BASIS-NCSN) and a noise-annealed version of flow-based models (BASIS-Glow). The inference procedure is performed in the image domain through Langevin dynamics (Parisi 1981), obtaining good quantitative and qualitative results. The authors extend the Langevin dynamics inference procedure to autoregressive models by re-training them with a noise schedule, introducing the Parallel and Flexible (PnF) method (Jayaram and Thickstun 2021). Although innovative, mainly when used for tasks such as inpainting, this method cannot use pre-trained autoregressive models directly, requiring fine-tuning under different noise levels. Further, working directly on the data domain, it exhibits a high inference time and scales with difficulty to higher resolutions. In this paper, we extend this line of research by proposing a separation procedure for latent autoregressive models that does not involve re-training, is scalable to arbitrary pre-trained checkpoints and is compatible with standard discrete samplers.

Background

This section briefly introduces vector-quantized autoencoders (VQ-VAE) and autoregressive models, since they are core components of the separation procedure used in *LASS*.

VQ-VAE

A data point $\mathbf{x} \in \mathbb{R}^N$ (N is the total length of the data point, e.g., the length of the audio sequence or the number of pixel channels in an image) can be mapped to a discrete latent domain via a VQ-VAE (van den Oord, Vinyals, and Kavukcuoglu 2017). First an encoder $E_\theta : \mathbb{R}^N \to \mathbb{R}^{S \times C}$ maps $\mathbf{x}$ to $E_\theta(\mathbf{x}) = (\mathbf{h}_1, \ldots, \mathbf{h}_S)$, where C denotes the number of latent channels and S the length of the latent sequence. A bottleneck block $B : \mathbb{R}^{S \times C} \to [K]^S$ casts the encoding into a discrete sequence $\mathbf{z} = (z_1, \ldots, z_S)$ by mapping each $\mathbf{h}_s$ into the index (also called token) $z_s = B(\mathbf{h}_s)$ of the nearest neighbor $\mathbf{e}_{z_s}$ contained in an (ordered) set $\mathcal{C} = \{\mathbf{e}_k\}_{k=1}^K$ of learned vectors in $\mathbb{R}^C$ (called codes). A decoder $D_\psi : [K]^S \to \mathbb{R}^N$ maps the latent sequence back into the data domain, obtaining a reconstruction $\hat{\mathbf{x}} = D_\psi(\mathbf{z})$. VQ-GAN (Esser, Rombach, and Ommer 2021) is an enhanced version of the VQ-VAE, where the training loss is augmented with a discriminator and a perceptual loss, that improve reconstruction quality while increasing the compression rate of the autoencoder. We refer the reader to (van den Oord, Vinyals, and Kavukcuoglu 2017) and (Esser, Rombach, and Ommer 2021) for more details on VQ-VAE and VQ-GAN. In the remainder of the article, we will refer to both models as VQ-VAE and make distinctions when necessary.

Autoregressive Models

An autoregressive model defines a probability distribution over a discrete domain $[K]^S$ (in our case, the latent domain of the VQ-VAE). The joint probability of a sequence $\mathbf{z} = (z_1, \ldots, z_S)$ is decomposed via the chain rule:

$$p_\phi(\mathbf{z}) = \prod_{s=1}^{S} p_\phi(z_s|\mathbf{z}_{<s}),$$

where $p_\phi(\cdot)$ is a learned parametric model, generally a neural network such as CNNs (van den Oord et al. 2016; Salimans et al. 2017) or Transformers (Vaswani et al. 2017). At inference time, samples can be obtained depending on the choice of a sampling procedure. Generally, ancestral sampling is used, where at each step, the token z_s is drawn stochastically from the conditional $p_\phi(z_s|\mathbf{z}_{<s})$, possibly employing top-k (Kool, van Hoof, and Welling 2020) filtering to increase the diversity of the generated data (Holtzman et al. 2020). When the goal is instead to maximize the probability of the whole sequence (w.r.t. all the sequences), heuristics like beam search are used (Reddy et al. 1977). Beam search maintains B possible hypotheses (beams) $\mathbf{z}^1, \ldots, \mathbf{z}^B$ in parallel during inference. At each step s, it computes the conditional distributions $p_\phi(z_s^b|\mathbf{z}^b_{<s})$ for each beam and selects the B new hypotheses that maximize the joint distributions $p_\phi(\mathbf{z}^b_{<s})p_\phi(z_s|\mathbf{z}^b_{<s})$.

Method

Let $\mathbf{x} = (\mathbf{x}^1, \mathbf{x}^2) \in \mathbb{R}^{2 \times N}$ denote two sources distributed according to $p_{\text{data}} = (p^1_{\text{data}}, p^2_{\text{data}})$ and $\mathbf{y} = (\mathbf{x}^1 + \mathbf{x}^2)/2$ an observable mixture. The goal of generative source separation is to estimate the sources $\mathbf{x}$ given the mixture $\mathbf{y}$, using the Bayesian posterior (assuming independent sources):

$$p(\mathbf{x}^1, \mathbf{x}^2|\mathbf{y}) \propto p^1_{\text{data}}(\mathbf{x}^1) p^2_{\text{data}}(\mathbf{x}^2) p(\mathbf{y}|\mathbf{x}^1, \mathbf{x}^2). \qquad (1)$$

Working directly with Eq. (1) in the continuous data domain is inefficient. To overcome this problem, we first model p_{data} with autoregressive models in the latent space of a VQ-VAE. By changing the domain, we subsequentially redefine the likelihood function $p(\mathbf{y}|\mathbf{x}^1, \mathbf{x}^2)$ such that no gradient-based optimization or model re-training is required. We address the first issue in the following subsection and the second in the subsequent one. We then describe how to perform inference using *LASS* to separate data and propose a post-inference refinement procedure.

Latent Autoregressive Source Separation

This paper explores the case in which p_{data} is estimated by a unique autoregressive model p_ϕ for all the sources (unsupervised[1]) and the case in which we have two independent ones, $p_\phi = (p_{\phi_1}, p_{\phi_2})$, for each of the two sources (weakly supervised), either in terms of class-conditioned or independently trained models. We will focus on this latter case in the following, since the former can be generalized setting $p_{\phi_1} = p_{\phi_2}$.

[1] Not to be confused with the unsupervised blind setting, i.e., in our unsupervised setting we have access to sources but we do not have class labels.

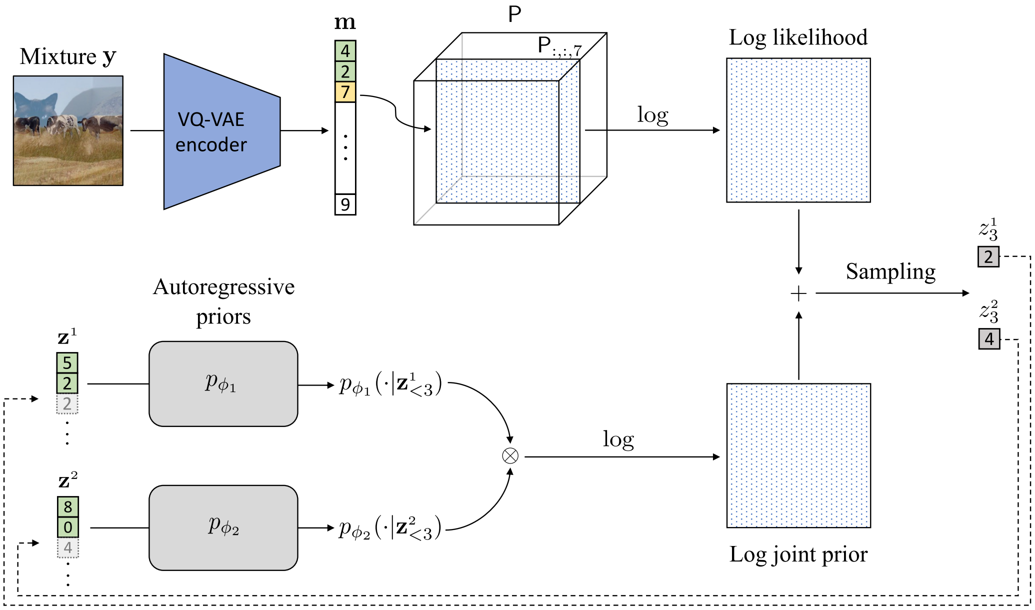

Figure 2: Schematic of the *LASS* separation procedure. The picture shows the separation procedure at $s = 3$ and is repeated until $s = S$. At the end of inference, we obtain $\mathbf{x}^1$ and $\mathbf{x}^2$ decoding $\mathbf{z}^1$ and $\mathbf{z}^2$ via the VQ-VAE decoder (not depicted in the picture). We refer the reader to Algorithm 1 for more details.

We denote the latent sources and mixtures, respectively, with $\mathbf{z} = (\mathbf{z}^1, \mathbf{z}^2) = B(E_\theta(\mathbf{x}))$ and $\mathbf{m} = B(E_\theta(\mathbf{y}))$. The posterior distribution in Eq. (1) can be locally expressed in the latent domain as:

$$p(\mathbf{z}_s|\mathbf{z}_{<s}, \mathbf{m}_{\leq s}) \propto p_\phi(\mathbf{z}_s|\mathbf{z}_{<s})p(\mathbf{m}_{\leq s}|\mathbf{z}_{\leq s}), \qquad (2)$$

for all $s = 1, \ldots, S$. The first factor is the (joint) Bayesian prior, modeled with autoregressive distributions. The second factor is the likelihood function, which quantifies the likelihood of the sequences $\mathbf{z}^1_{\leq s}, \mathbf{z}^2_{\leq s}$ to combine into $\mathbf{m}_{\leq s}$.

Since each code in the convolutional VQ-VAE describes a local portion of the data, and given that the mixing operation is point-wise in the data domain, the mixing relation between latent codes is local also in the latent domain. As such, we can drop the dependency on the previous context inside the likelihood function in Eq. (2), approximating it as:

$$p(\mathbf{m}_{\leq s}|\mathbf{z}_{\leq s}) \approx p(m_s|\mathbf{z}_s). \qquad (3)$$

Notice that not depending on the global context and thus on the specific position in the sequence, we can drop the position index s:

$$p(m_s|\mathbf{z}_s) = p(m_s|z_s^1, z_s^2) = p(m|z^1, z^2). \qquad (4)$$

The following subsection describes how *LASS* models the likelihood function.

Discrete Likelihoods for Source Separation

Previous works in generative source separation (Jayaram and Thickstun 2020, 2021) model likelihood functions directly in the data domain, typically employing a σ-isotropic Gaussian term:

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}|(\mathbf{x}^1 + \mathbf{x}^2)/2, \sigma^2\mathbf{I}).$$

In our setting, we cannot combine z_s^1 and z_s^2 (or the associate dense codes $\mathbf{e}_{z_s^1}$ and $\mathbf{e}_{z_s^2}$) with the canonical sum operation, given that the VQ-VAE does not impose an explicit arithmetic structure on the latent space.

To cope with this, we model the likelihood function in Eq. (4) using discrete conditionals, represented with rank-3 tensors[2] $\mathsf{P} \in \mathbb{R}^{K\times K\times K}$:

$$p(\cdot\,|z^1, z^2) = \mathsf{P}_{z^1,z^2,:}.$$

In order to learn P, we perform frequency counts on latent mixed tokens given the latent sources' tokens, by iterating over a dataset X. We first initialize a null integer tensor $\mathsf{F}^0 \in \mathbb{N}^{K\times K\times K}$. Iterating over $\mathbf{x}^1, \mathbf{x}^2 \in X$, we compute $\mathbf{y} = (\mathbf{x}^1 + \mathbf{x}^2)/2$, then obtain the latent sequences $\mathbf{z}^1 = B(E_\theta(\mathbf{x}^1)), \mathbf{z}^2 = B(E_\theta(\mathbf{x}^2))$ and $\mathbf{m} = B(E_\theta(\mathbf{y}))$. For each entry $(z_s^1, z_s^2, m_s) \in (\mathbf{z}^1, \mathbf{z}^2, \mathbf{m})$, at step t, we simply increment the previous count by one:

$$\mathsf{F}^t_{z_s^1,z_s^2,m_s} = \mathsf{F}^{t-1}_{z_s^1,z_s^2,m_s} + 1\,,$$
$$\mathsf{F}^t_{z_s^2,z_s^1,m_s} = \mathsf{F}^{t-1}_{z_s^2,z_s^1,m_s} + 1\,.$$

We permute the order of the addends in order to enforce the commutative property of the sum. After performing the statistics, we can define P as:

$$\mathsf{P}_{z^1,z^2,:} = \frac{1}{\sum_{k=1}^K \mathsf{F}_{z^1,z^2,k}} \mathsf{F}_{z^1,z^2,:}.$$

At inference time, the likelihood function (parametric in z^1 and z^2, with m fixed) can be obtained by slicing the tensor

[2] We follow the notation for tensors as in Goodfellow, Bengio, and Courville (2016).

Algorithm 1: *LASS* inference

```
Input: $\mathbf{y}$
Output: $\mathbf{x}^1, \mathbf{x}^2$
 1: $\mathbf{m} \leftarrow B(E_\theta(\mathbf{y}))$
 2: $\mathbf{z}^1 \leftarrow []$
 3: $\mathbf{z}^2 \leftarrow []$
 4: for $s = 1$ to $S$ do
 5:     prior $\leftarrow \log(p_{\phi_1}(\cdot\,|\mathbf{z}^1) \otimes p_{\phi_2}(\cdot\,|\mathbf{z}^2))$
 6:     likelihood $\leftarrow \log(\mathsf{P}_{:,:,m_s})$
 7:     posterior $\leftarrow$ prior $+\ \lambda$ likelihood
 8:     $(z_s^1, z_s^2) \leftarrow \texttt{Sampler}($posterior$)$
 9:     $\mathbf{z}^1 \leftarrow \texttt{concat}(\mathbf{z}^1, z_s^1)$
10:     $\mathbf{z}^2 \leftarrow \texttt{concat}(\mathbf{z}^2, z_s^2)$
11: end for
12: $\mathbf{x}^1 \leftarrow D_\psi(\mathbf{z}^1)$
13: $\mathbf{x}^2 \leftarrow D_\psi(\mathbf{z}^2)$
14: return $\mathbf{x}^1, \mathbf{x}^2$
```

along m, namely:

$$p(m|\cdot,\cdot) = \mathsf{P}_{:,:,m}.$$

At first glance, modeling the conditional distributions without parameters could seem memory inefficient, with a complexity of $O(K^3)$. In practice, the tensor P is *highly sparse*. We showcase this in Table 1 for all our experiments, where the density of P is defined as the percentage of nonzero elements in P.

Employing discrete likelihood functions for source separation in the latent domain of a VQ-VAE is a flexible approach; there is no need to change the VQ-VAE representation, the non-parametric learning procedure does not depend on hyperparameters, and the autoregressive priors do not require re-training.

Inference Procedure

Given an observable mixture $\mathbf{y}$, the autoregressive priors p_{ϕ_1}, p_{ϕ_2} and the learned likelihood tensor P, it is possible to perform inference and estimate $\mathbf{x}^1, \mathbf{x}^2$, as described in Algorithm 1 and depicted in Figure 2.

We start by mapping $\mathbf{y}$ to the latent domain obtaining $\mathbf{m} = B(E_\theta(\mathbf{y}))$ and initializing the estimates $\mathbf{z}^1, \mathbf{z}^2$ with the empty sequences. The algorithm iterates over $s = 1, \ldots, S$.

At each step, the joint prior (a $K \times K$ matrix) is computed (Line 5) by taking the outer product of the two distributions predicted by the autoregressive models conditioned over the past context. We use the logarithms of the distributions for numerical stability. The log-likelihood function is computed next (Line 6), applying the logarithm on $\mathsf{P}_{:,:,m_s}$. In our experiments, we can apply different scaling factors λ to the log-likelihood to balance it to the priors. The two matrices are then combined to form the posterior on Line 7.

Finally (Lines 8-10), different techniques can be employed to sample the best candidate tokens (z_s^1, z_s^2) from the posterior. In our experiments, we used ancestral sampling (with and without top-k filtering) and beam search. After the inference loop ends, the estimated sequences are mapped back to the data domain with the decoder of the VQ-VAE (Lines 12-13), obtaining $\mathbf{x}^1$ and $\mathbf{x}^2$.

Dataset	K	Density (%)
MNIST	256	1.49×10^0
CelebA	512	6.06×10^0
CelebA-HQ	1024	3.80×10^{-1}
ImageNet	16384	3.90×10^{-3}
Slakh (Drum + Bass)	2048	7.60×10^{-2}

Table 1: Statistics on likelihood functions over different datasets. K is the number of VQ-VAE (or VQ-GAN) latent codes. Density is the percentage of nonzero elements in the likelihood function.

Post-inference Refinement The quality of the separated images is limited by the quality of the images obtained via the VQ-VAE decoder. To enhance the separations we can adopt an additional refinement step by iteratively optimizing the VQ-VAE latent representations of the samples:

$$\mathbf{e}_{t+1}^1 = \mathbf{e}_t^1 + \alpha \nabla_{\mathbf{e}_t^1} \| D_\psi(\mathbf{e}_t^1) + D_\psi(\mathbf{e}_t^2) - 2\mathbf{y} \|_2 \quad (5)$$

$$\mathbf{e}_{t+1}^2 = \mathbf{e}_t^2 + \alpha \nabla_{\mathbf{e}_t^2} \| D_\psi(\mathbf{e}_t^1) + D_\psi(\mathbf{e}_t^2) - 2\mathbf{y} \|_2 \quad (6)$$

for $t = 1, \ldots, T-1$ and $\mathbf{e}_1^1 = E_\theta(\mathbf{x}^1)$, $\mathbf{e}_1^2 = E_\theta(\mathbf{x}^2)$. In simple words, we optimize for dense latent embeddings such that their decodings better sum to the mixture, initializing them to the output of Algorithm 1. We found this strategy particularly helpful on the MNIST datset, where we assess the quality of the separation through a pixel-wise metric (PSNR) and the VQ-VAE tends to produce smooth images.

Experiments

We perform quantitative and qualitative experiments on various datasets to demonstrate the efficacy and scalability of *LASS*. In the image domain, we evaluate on MNIST (Lecun et al. 1998) and CelebA (32×32) (Liu et al. 2015) and present qualitative results on the higher resolution datasets CelebA-HQ (256×256) (Karras et al. 2018) and ImageNet (256×256) (Deng et al. 2009). In the audio domain, we test on Slakh2100 (Manilow et al. 2019), a large dataset for music source separation suitable for generative modeling. We conducted all our experiments on a single Nvidia RTX 3090 GPU with 24 GB of VRAM. Implementation details for all the models are listed on the companion website[3].

Image Source Separation

We choose the Transformer architecture (Vaswani et al. 2017) as the autoregressive backbone for all image source separation experiments. With MNIST and CelebA, we first train a VQ-VAE, then train the autoregressive Transformer on its latent space. We use $K = 256$ codes on MNIST and $K = 512$ on CelebA, given that CelebA presents more variability, requiring more information to reconstruct data. On CelebA-HQ and ImageNet, we leverage pre-trained VQ-GANs (Esser, Rombach, and Ommer 2021) alongside the pre-trained Transformers published by the authors[4]

[3] `github.com/gladia-research-group/latent-autoregressive-source-separation`

[4] `github.com/CompVis/taming-transformers`

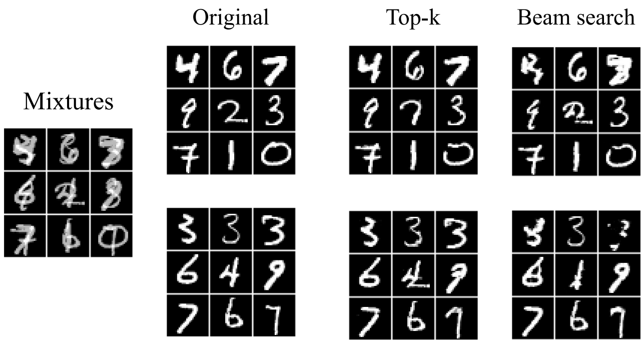

Figure 3: Results on MNIST with top-k sampling ($k = 32$) over a random batch of examples. Top-k sampling produces more defined digits, in agreement with the results in Table 3.

Separation Method	MNIST (PSNR)	CelebA (FID)
Average	14.9	15.19
NMF	9.4	-
S-D	18.5	-
BASIS Glow	22.7	-
BASIS NCSN	29.3	7.55
***LASS* (Ours)**	24.2	8.96

Table 2: Comparison with other methods on MNIST and CelebA test set. Results are reported in PSNR (higher is better) and FID (lower is better).

(`celebahq_transformer` checkpoint for CelebA-HQ and `cin_transformer` for ImageNet). Given the flexibility of *LASS* , they are employed inside the separation algorithm without modifications. On CelebA-HQ the VQ-GAN has $K = 1024$ codes, while on ImageNet has $K = 16384$. As a first step, in all image-based experiments we learn the P tensor using the procedure presented in the section "Method". As shown in Table 1, CelebA presents the lowest sparsity (highest density) while ImageNet has the highest. In all cases, density is below 7%, and the inference procedure is not affected by memory issues.

Quantitative Results To assess the quality of image separations produced by *LASS*, we compare our method with different baselines on MNIST and CelebA.

On MNIST, we compare *LASS* with results reported for the two generative separation methods "BASIS NCSN" (score-based) and "BASIS Glow" (noise-annealed flow-based) from (Jayaram and Thickstun 2020), the GAN-based "S-D" method (Kong et al. 2019), the fully supervised version of Neural Egg "NES" and the "Average" baseline, where separations are obtained directly from the mixture $\mathbf{x}^1 = \mathbf{x}^2 = \mathbf{y}/2$. In all these cases, the evaluation metric is the PSNR (Peak Signal to Noise Ration) (Horé and Ziou 2010). We follow the experimental procedure of (Jayaram and Thickstun 2020) on MNIST and perform separation on a set of 6,000 mixtures obtained by combining 12,000 test sources. In order to choose the best sampler for this dataset, we validate the set of samplers in Table 3 on 1,000 mixtures constructed from the test split. We find that stochastic samplers perform best (PSNR > 20 dB) while MAP methods do not reach a satisfactory performance. We hypothesize that beam search tends to fall into sub-optimal solutions by performing incorrect choices in early inference over sparse images such as MNIST digits. Top-k sampling with $k = 32$ performs best, so we choose it to perform the evaluation (a qualitative comparison is shown in Figure 3). For each mixture in the test set we sample a candidate batch of 512 separations, select the separation whose sum better matches the mixture (w.r.t. the $\mathcal{L}_2$ distance), and finally perform the refinement procedure in Eqs. (5), (6) with $T = 500$ and $\alpha = 0.1$. Evaluation metrics on this experiment are shown in Table 2, while inference time is reported in Table 4. Our method achieves higher metrics than "NMF", "S-D" and "BASIS Glow" and is faster than "BASIS NCSN", thanks to the latent quantization. The higher PSNR achieved by the later method can be attributed to the fact that, in their case, the underlying generative models perform sampling directly in the image domain; in our case, the VQ-VAE compression can hinder the metrics.

Sampling Method	MNIST (PSNR)	Slakh (SDR)
Greedy	17.36 ± 5.90	1.23 ± 2.33
Beam Search	16.96 ± 5.78	5.01 ± 2.39
Ancestral Sampl.	24.03 ± 6.37	4.23 ± 2.29
Top-k ($k = 16$)	23.74 ± 6.55	3.13 ± 2.53
Top-k ($k = 32$)	24.23 ± 6.23	2.93 ± 2.20
Top-k ($k = 64$)	23.85 ± 6.13	3.24 ± 3.29

Table 3: Performance of *LASS* with different sampling methods. On MNIST, the reported score is PSNR (dB) (higher is better), while on Slakh is SDR (dB) (higher is better). When stochastic samplers are used (ancestral or top-k), the selected solution in the batch is the one whose sum minimizes the $\mathcal{L}_2$ distance to the input mixture.

We compare our method to "BASIS NCSN", using the pre-trained NCSN model (Song and Ermon 2019) on CelebA. In this case, we evaluate against the FID metric (Heusel et al. 2017) instead of PSNR, given that for datasets that feature more variability than MNIST, source separation can be an underdetermined task (Jayaram and Thickstun 2020): semantically good separations can receive a low PSNR score since the generative models may alter features such as color and cues (an effect amplified by a VQ-GAN decoder). The FID metric better quantifies if the separations belong to the distribution of the sources. We test on 10,000 mixtures computed from pair of images in the validation split using a top-k sampler with $k = 32$. We scale the likelihood term by multiplying it by $\lambda = 3$. It is a known fact in the literature that score-based models outperform autoregressive models on FID metrics (Dockhorn, Vahdat, and Kreis 2021) on different datasets, yet our method paired with an autoregressive model shows competitive results with respect to the score-based "BASIS NCSN".

Qualitative Results To demonstrate the flexibility of *LASS* in using existing models without any modification, we

	Method	Time
MNIST	***LASS* (Ours)**	4.49 s ± 0.27 s
	BASIS NCSN	53.34 s ± 0.51 s
Slakh	***LASS* (Ours)**	1.33 min ± 0.87 s
	PnF	42.29 min ± 1.08 s

Table 4: Inference speed comparisons for computing one separation. To estimate variance, we repeat inference 10 times on MINST and 3 times on Slakh. We consider 3-second-long mixtures on Slakh.

Separation Method	Avg	Drums	Bass
rPCA	0.82	0.60	1.05
ICA	-1.26	-0.99	-1.53
HPSS	-0.45	-0.56	-0.33
REPET	1.04	0.53	1.54
FT2D	0.95	0.59	1.31
***LASS* (Ours)**	4.86	4.73	4.98
Demucs	5.39	5.42	5.36
Conv-Tasnet	5.47	5.51	5.43

Table 5: Comparison with other source separation methods on Slakh ("Drums" and "Bass" classes). Results are reported in SDR (dB) (higher is better). Lower part of the table shows supervised methods. With "Avg" we refer to the mean between the results over the two classes.

leverage pre-trained checkpoints on CelebA-HQ and ImageNet. In this case, only the likelihood tensor P is learned. We showcase a curated results list in Figure 1 and a more extensive list on the companion website. To the best of our knowledge, our method is the first to scale up to 256×256 resolutions and can be used with more powerful latent autoregressive models without re-training (which is cumbersome for very large models). As such, end-users can perform generative separation without having access to extensive computational resources for training these large models.

Music Source Separation

We perform experiments on the Slakh2100 dataset (Manilow et al. 2019) for the music source separation task. This dataset contains 2100 songs with separated sources belonging to 34 instrument categories, for a total of 145 hours of mixtures. We focus on the "Drums" and "Bass" data classes, with tracks sampled at 22kHz. We use the public checkpoint of Dhariwal et al. (2020) for the VQ-VAE model, taking advantage of its expressivity in modeling audio data over a quantized domain. Given that such a model is trained at 44kHz, we upsample input data linearly, then downsample the output back at 22kHz. For the two autoregressive priors, we train two Transformer models, one for "Drums" and another for "Bass" and learn the likelihood function over the VQ-VAE (statistics are reported in Table 1). We compare *LASS* to a set of unsupervised blind source separation methods -"rPCA" (Huang et al. 2012), "ICA" (Hyvärinen and Oja 2000), "HPSS" (Rafii and Pardo 2012), "FT2D" (Seetharaman, Pishdadian, and Pardo 2017) - and to two supervised baselines Demucs (Défossez et al. 2019) and Conv-Tasnet (Luo and Mesgarani 2019) using the SDR (dB) evaluation metric computed with the `museval` library (Stöter, Liutkus, and Ito 2018). To evaluate the methods, we select 900 music chunks of 3 seconds from the test splits of the "Drums" and "Bass" classes, combining them to form 450 mixtures. The validation dataset is constructed similarly (with different music chunks). As a sampling strategy, we use beam search since it shows the best results on a validation of 50 mixtures (Table 3), using $B = 100$ beams. Evaluation results are reported in Table 5: *LASS* clearly performs better than all the blind unsupervised baselines and is comparable with the results obtained by methods that use supervision. Furthermore, we compare the time performance of *LASS* against the generative source separation method "PnF" (Jayaram and Thickstun 2021) by evaluating the time required to separate a mixture of 3 seconds sampled at 22 kHz (piano vs. voice on "PnF"). Results in Table 4 show that *LASS* is significantly faster, and as such, it can be adopted in more realistic inference scenarios.

Limitations

In this paper we limit our analysis to the separation of two sources. Even if this is a common setup especially in image separation (Jayaram and Thickstun 2021; Halperin, Ephrat, and Hoshen 2019), dealing with multiple sources is a possible line of future work. Under our framework, this would require to increase the dimensions of the discrete distributions (both the priors and the likelihood function). To alleviate this problem, techniques such as recursive separation may be employed (Takahashi et al. 2019).

Another limitation of the proposed method is the locality assumption taken in Eq. (3). Different tasks such as colorization and super-resolution would require a larger conditioning context, and newer quantization schemes to aggregate latent codes on global contexts (using self-attention in the encoder and the decoder of the VQ-VAE) (Yu et al. 2021). Adopting a VQ-VAE quantized with respect to the latent channels (Xu et al. 2021) combined with a parametric likelihood function could be a way to solve this limitation, while still maintaining the flexible separation between VQ-VAE, priors, and likelihoods presented in the paper.

Conclusion

In this paper, we proposed *LASS* as a source separation method for latent autoregressive models that does not modify the structure of the priors. We have tested our method on different datasets and have shown results comparable to state-of-the-art methods while being more scalable and faster at inference time. Additionally, we have shown qualitative results at a higher resolution than those proposed by the competitors. We believe our method will benefit from the improved quality of newer autoregressive models, improving both the quantitative metrics and the perceptive results.

Acknowledgments

We thank Marco Fumero for helping to compute the blind unsupervised baseline metrics in the audio setting. This work is supported by the ERC Grant no. 802554 (SPECGEO) and the IRIDE grant from DAIS, Ca' Foscari University of Venice, Italy.

References

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Proc. NeurIPS*, 33: 1877–1901.

Castellon, R.; Donahue, C.; and Liang, P. 2021. Codified audio language modeling learns useful representations for music information retrieval. *arXiv preprint arXiv:2107.05677*.

Comon, P. 1994. Independent Component Analysis, a new concept? *Signal Processing*.

Défossez, A. 2021. Hybrid Spectrogram and Waveform Source Separation. In *Proceedings of the ISMIR 2021 Workshop on Music Source Separation*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *Proc. CVPR*, 248–255.

Dhariwal, P.; Jun, H.; Payne, C.; Kim, J. W.; Radford, A.; and Sutskever, I. 2020. Jukebox: A Generative Model for Music. arXiv:2005.00341.

Dockhorn, T.; Vahdat, A.; and Kreis, K. 2021. Score-Based Generative Modeling with Critically-Damped Langevin Diffusion. *ArXiv*, abs/2112.07068.

Dovrat, S.; Nachmani, E.; and Wolf, L. 2021. Many-Speakers Single Channel Speech Separation with Optimal Permutation Training. In *Interspeech*.

Défossez, A.; Usunier, N.; Bottou, L.; and Bach, F. 2019. Music Source Separation in the Waveform Domain. *arXiv:1911.13254 [cs, eess, stat]*. ArXiv: 1911.13254.

Esser, P.; Rombach, R.; and Ommer, B. 2021. Taming transformers for high-resolution image synthesis. In *Proc. CVPR*, 12873–12883.

Goodfellow, I.; Bengio, Y.; and Courville, A. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Proc. NIPS*, 27.

Gusó, E.; Pons, J.; Pascual, S.; and Serrà, J. 2022. On Loss Functions and Evaluation Metrics for Music Source Separation. In *Proc. ICASSP*, 306–310.

Halperin, T.; Ephrat, A.; and Hoshen, Y. 2019. Neural separation of observed and unobserved distributions. *36th International Conference on Machine Learning, ICML 2019*, 2019-June: 4548–4557.

Hertz, A.; Mokady, R.; Tenenbaum, J.; Aberman, K.; Pritch, Y.; and Cohen-Or, D. 2022. Prompt-to-Prompt Image Editing with Cross Attention Control. *arXiv preprint arXiv:2208.01626*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *Proc. NeurIPS*, volume 30.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Proc. NeurIPS*, 33: 6840–6851.

Holtzman, A.; Buys, J.; Du, L.; Forbes, M.; and Choi, Y. 2020. The Curious Case of Neural Text Degeneration. In *Proc. ICLR*.

Horé, A.; and Ziou, D. 2010. Image Quality Metrics: PSNR vs. SSIM. In *Proc. ICPR*, 2366–2369.

Huang, P.-S.; Chen, S. D.; Smaragdis, P.; and Hasegawa-Johnson, M. 2012. Singing-voice separation from monaural recordings using robust principal component analysis. In *Proc. ICASSP*, 57–60. IEEE.

Huang, P.-S.; Kim, M.; Hasegawa-Johnson, M. A.; and Smaragdis, P. 2014. Singing-Voice Separation from Monaural Recordings using Deep Recurrent Neural Networks. In *Proc. ISMIR*.

Hyvärinen, A.; and Oja, E. 2000. Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5): 411–430.

Jayaram, V.; and Thickstun, J. 2020. Source Separation with Deep Generative Priors. In *Proc. ICML*, PMLR.

Jayaram, V.; and Thickstun, J. 2021. Parallel and flexible sampling from autoregressive models via langevin dynamics. In *Proc. ICML*, 4807–4818. PMLR.

Karras, T.; Aila, T.; Laine, S.; and Lehtinen, J. 2018. Progressive Growing of GANs for Improved Quality, Stability, and Variation. In *Proc. ICLR*.

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. In *Proc. ICLR*.

Kong, Q.; Xu, Y.; Wang, W.; Jackson, P. J. B.; and Plumbley, M. D. 2019. Single-Channel Signal Separation and Deconvolution with Generative Adversarial Networks. In *Proc. IJCAI*, 2747–2753. AAAI Press. ISBN 9780999241141.

Kool, W.; van Hoof, H.; and Welling, M. 2020. Ancestral Gumbel-Top-k Sampling for Sampling Without Replacement. *Journal of Machine Learning Research*, 21(47): 1–36.

Lecun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Liu, J.-Y.; and Yang, Y.-H. 2018. Denoising Auto-encoder with Recurrent Skip Connections and Residual Regression for Music Source Separation. arXiv:1807.01898.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep Learning Face Attributes in the Wild. In *Proc. ICCV*.

Lluís, F.; Pons, J.; and Serra, X. 2019. End-to-End Music Source Separation: Is it Possible in the Waveform Domain? In *INTERSPEECH*, 4619–4623.

Luo, Y.; and Mesgarani, N. 2019. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 27(8): 1256–1266.

Manilow, E.; Wichern, G.; Seetharaman, P.; and Le Roux, J. 2019. 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.

Narayanaswamy, V.; Thiagarajan, J. J.; Anirudh, R.; and Spanias, A. 2020. Unsupervised Audio Source Separation using Generative Priors. arXiv:2005.13769.

Nugraha, A. A.; Liutkus, A.; and Vincent, E. 2016. Multichannel Audio Source Separation With Deep Neural Networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(9): 1652–1664.

Parisi, G. 1981. Correlation functions and computer simulations. *Nuclear Physics B*, 180(3): 378–384.

Postolache, E.; Pons, J.; Pascual, S.; and Serrà, J. 2022. Adversarial Permutation Invariant Training for Universal Sound Separation. *arXiv preprint arXiv:2210.12108*.

Rafii, Z.; and Pardo, B. 2012. Repeating pattern extraction technique (REPET): A simple method for music/voice separation. *IEEE transactions on audio, speech, and language processing*, 21(1): 73–84.

Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; and Sutskever, I. 2021. Zero-shot text-to-image generation. In *Proc. ICML*, 8821–8831. PMLR.

Razavi, A.; van den Oord, A.; and Vinyals, O. 2019. Generating Diverse High-Fidelity Images with VQ-VAE-2. In *Proc. NeurIPS*.

Reddy, D. R.; et al. 1977. Speech understanding systems: A summary of results of the five-year research effort. *Department of Computer Science. Camegie-Mell University, Pittsburgh, PA*, 17: 138.

Roweis, S. T. 2000. One Microphone Source Separation. In *Proc. NIPS*.

Salimans, T.; Karpathy, A.; Chen, X.; and Kingma, D. P. 2017. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*.

Sanh, V.; Webson, A.; Raffel, C.; et al. 2022. Multitask Prompted Training Enables Zero-Shot Task Generalization. In *Proc. ICLR*.

Seetharaman, P.; Pishdadian, F.; and Pardo, B. 2017. Music/voice separation using the 2d fourier transform. In *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 36–40. IEEE.

Smaragdis, P.; Févotte, C.; Mysore, G. J.; Mohammadiha, N.; and Hoffman, M. 2014. Static and Dynamic Source Separation Using Nonnegative Factorizations: A unified view. *IEEE Signal Processing Magazine*, 31(3): 66–75.

Song, Y.; and Ermon, S. 2019. Generative Modeling by Estimating Gradients of the Data Distribution. In *Advances in Neural Information Processing Systems*, 11895–11907.

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2021. Score-Based Generative Modeling through Stochastic Differential Equations. In *Proc. ICLR*.

Stöter, F.-R.; Liutkus, A.; and Ito, N. 2018. The 2018 Signal Separation Evaluation Campaign. In *Proc. LVA/ICA*, 293–305.

Subakan, Y. C.; and Smaragdis, P. 2018. Generative adversarial source separation. In *Proc. ICASSP*, 26–30. IEEE.

Takahashi, N.; Goswami, N.; and Mitsufuji, Y. 2018. Mmdenselstm: An Efficient Combination of Convolutional and Recurrent Neural Networks for Audio Source Separation. In *Proc. IWAENC*, 106–110.

Takahashi, N.; Parthasaarathy, S.; Goswami, N.; and Mitsufuji, Y. 2019. Recursive speech separation for unknown number of speakers. *arXiv preprint arXiv:1904.03065*.

Uhlich, S.; Giron, F.; and Mitsufuji, Y. 2015. Deep neural network based instrument extraction from music. In *Proc. ICASSP*.

van den Oord, A.; Kalchbrenner, N.; Espeholt, L.; Vinyals, O.; Graves, A.; et al. 2016. Conditional image generation with pixelcnn decoders. *Proc. NeurIPS*, 29.

van den Oord, A.; Vinyals, O.; and Kavukcuoglu, K. 2017. Neural Discrete Representation Learning. In *Proc. NeurIPS*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Proc. NeurIPS*, 30.

Wei, J.; Bosma, M.; Zhao, V. Y.; Guu, K.; Yu, A. W.; Lester, B.; Du, N.; Dai, A. M.; and Le, Q. V. 2021. Finetuned Language Models Are Zero-Shot Learners. *CoRR*, abs/2109.01652.

Wisdom, S.; Erdogan, H.; Ellis, D. P. W.; Serizel, R.; Turpault, N.; Fonseca, E.; Salamon, J.; Seetharaman, P.; and Hershey, J. R. 2021. What's all the Fuss about Free Universal Sound Separation Data? In *Proc. ICASSP*, 186–190.

Wisdom, S.; Tzinis, E.; Erdogan, H.; Weiss, R.; Wilson, K.; and Hershey, J. 2020. Unsupervised Sound Separation Using Mixture Invariant Training. In *Proc. NeurIPS*, volume 33, 3846–3857.

Xu, Y.; Song, Y.; Garg, S.; Gong, L.; Shu, R.; Grover, A.; and Ermon, S. 2021. Anytime sampling for autoregressive models via ordered autoencoding. *arXiv preprint arXiv:2102.11495*.

Yang, H.; Lin, J.; Yang, A.; Wang, P.; Zhou, C.; and Yang, H. 2022. Prompt Tuning for Generative Multimodal Pretrained Models. *arXiv preprint arXiv:2208.02532*.

Yu, J.; Li, X.; Koh, J. Y.; Zhang, H.; Pang, R.; Qin, J.; Ku, A.; Xu, Y.; Baldridge, J.; and Wu, Y. 2021. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*.

Yu, J.; Xu, Y.; Koh, J. Y.; Luong, T.; Baid, G.; Wang, Z.; Vasudevan, V.; Ku, A.; Yang, Y.; Ayan, B. K.; et al. 2022. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*.