

DreamAudio: Customized Text-to-Audio Generation with Diffusion Models

Yi Yuan, Xubo Liu, Haohe Liu, Xiyuan Kang, Zhuo Chen
Yuxuan Wang, Mark D. Plumbley, Wenwu Wang

Abstract—With the development of large-scale diffusion-based and language-modeling-based generative models, impressive progress has been achieved in text-to-audio generation. Despite producing high-quality outputs, existing text-to-audio models mainly aim to generate semantically aligned sound and fall short of controlling fine-grained acoustic characteristics of specific sounds. As a result, users who need specific sound content may find it difficult to generate the desired audio clips. In this paper, we present DreamAudio for customized text-to-audio generation (CTTA). Specifically, we introduce a new framework that is designed to enable the model to identify auditory information from user-provided reference concepts for audio generation. Given a few reference audio samples containing personalized audio events, our system can generate new audio samples that include these specific events. In addition, two types of datasets are developed for training and testing the proposed systems. The experiments show that DreamAudio generates audio samples that are highly consistent with the customized audio features and aligned well with the input text prompts. Furthermore, DreamAudio offers comparable performance in general text-to-audio tasks. We also provide a human-involved dataset containing audio events from real-world CTTA cases as the benchmark for customized generation tasks. Demos are available at https://yyua8222.github.io/DreamAudio_demo/.

Index Terms—audio generation, diffusion model, retrieval argumentation, customized generation, AIGC

I. INTRODUCTION

Audio generation, as a crucial technology for enabling artificial intelligence generated content (AIGC) [1], has gained significant interest from the research community. In recent years, conditional audio generation has become a popular paradigm, where music, speech, or general sound effects are generated based on a variety of conditions, such as text [2], images [3], [4], and videos [5], [6]. This opened up new opportunities for a range of potential applications, including audio generation for movies [7], games [8], and audiobooks [9].

Facilitated by advances in diffusion-based generative models [10]–[12] and large-scale audio-language datasets [13]–[16], several models have been developed for audio generation with text prompts as conditions, such as AudioLDM [2],

Yi Yuan, Xubo Liu, Haohe Liu, Xiyuan Kang, and Wenwu Wang are with the School of Computer Science and Electronic Engineering, University of Surrey, Guildford, UK. Email: {yi.yuan, xubo.liu, haohe.liu, xk00063, m.plumbley, w.wang}@surrey.ac.uk.

Mark D. Plumbley is with Department of Informatics, King’s College London, London, UK. Email: mark.plumbley@kcl.ac.uk.

Zhuo Chen and Yuxuan Wang are with the Seed Group, ByteDance Inc. Email: {zhuo.chen1, wangyuping, wangyuxuan.11}@bytedance.com.

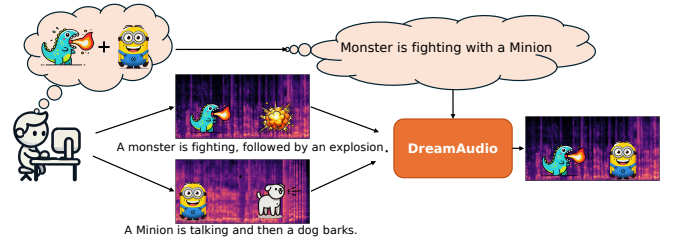


Fig. 1. An illustration of *DreamAudio* for audio generation with customized content of “monster fighting” and “Minion talking”. The system takes both the text prompt and user-provided audio-caption pairs as the reference concepts, and generate audio content consistent with the description “Monster is fighting with a Minion”.

AudioGen [17], DiffSound [18], TANGO [19], Make-an-Audio2 [20], Re-AudioLDM [21], and AudioLDM2 [22]. Building on the “semantic prior” learned from large collections of datasets [23] to associate textual concepts with audio features, these models have shown strong capabilities in generating audio samples of high quality, fidelity and diversity. For example, when provided with the text prompt *dog barking*, the model leverages this particular “semantic prior” to generate various audio clips depicting *dog barking* across varying species, emotions, and durations.

Despite significant progress in diffusion-based methods [24]–[28], current text-to-audio generation systems often lack the flexibility to customize content based on personalized intentions. This can cause problems in real-world multimedia production, which often requires the generation of audio samples to be tailored for specific features. For example, it can be challenging for current TTA models to generate sounds that are consistent with the text prompt “*a monster is fighting with a Minion*”. This is because audio events such as “monster fight” and “Minion talk” are unique and rare, or with a specific timbre, which can hardly be found in any current training data. Hence, users often need to go through multiple rounds of trial and error to produce the desired output and may encounter difficulties in achieving the optimal result [29].

To address this challenge, several works have explored the ideas for generating rare or unseen audio events, especially for few-shot or zero-shot scenarios. One of the pioneering works is the Re-AudioLDM [21], which applies retrieval-based techniques to improve the performance for the generation of rare audio events using audio-caption pairs retrieved from external data. Similarly, AudioBox TTA-RAG [30] introduced a retrieval embedding module to improve the performance for few-shot and zero-shot audio generation tasks. Although these previous approaches effectively improve the semantic

accuracy in generating infrequent or unseen audio events, they are still unable to provide explicit control over the sound effects generated, as specified by users.

In this paper, we propose a new TTA task, namely, customized text-to-audio generation (CTTA), where the audio content produced by the generation system can be controlled and customized by users. For instance, the system is capable to generate the *dog barking* sound from a specific dog with unique timber, or the sound for a specific *monster fighting* which is not included in the training dataset. We refer to these featured audio events as user reference concepts. To achieve this, we introduce *DreamAudio*, a latent diffusion-based system with flow matching for the CTTA task. Inspired by ControlNet [31], we propose a multi-reference customization (MRC) structure for the generator module to identify the reference concepts and control the generated content. More specifically, we design a new group of encoder blocks to extract features from the user-provided reference concepts. The system is trained to fuse such customized features for the generation of the audio output. By incorporating multiple cross-attention modules that link the target text prompt with the corresponding reference text, our proposed approach establishes a pipeline capable of generating user-preferred audio samples without requiring concept-specific fine-tuning during inference. In contrast to tuning-based customization methods, which necessitate model updates for each new reference even after the model has been fully trained, *DreamAudio* directly extracts and integrates features from the new references within a single forward pass. Figure 1 shows an example of CTTA, where the model is enabled to generate the audio sample with customized “monster fighting” and “Minion talking”, in terms of the provided reference concepts.

To facilitate model development and evaluation, we create two datasets with different formats by concatenating and overlapping different audio events, called Customized-Concatenation and Customized-Overlay. In addition, we collect several special audio events and then manually design various customized cases, creating a small-scale dataset that more closely reflects real-world multimedia production scenarios and serves as a benchmark for CTTA. Experiments conducted on these datasets demonstrate that our method effectively handles customized audio generation and establishes a strong baseline for this new CTTA task.

Our contributions can be summarized as follows.

- We propose a novel audio generation model, *DreamAudio*, that is capable of performing content-customized audio generation with text prompts as condition.
- We propose a multi-reference customization (MRC) structure, which allows the features from the reference audio to be fused with input prompts for text-to-audio generation.
- We develop two new datasets for model training and evaluation, and establish a new benchmark for content-customized text-to-audio generation.
- Our experiments show that *DreamAudio* significantly enhances the ability of TTA models for customized audio generation and establishes a strong baseline for this new CTTA task.

II. RELATED WORK

A. Diffusion Models

Diffusion-based models [11], [32] have demonstrated improved performance in generative tasks for image [25], [33]–[35], audio [36]–[38], and video [39], [40]. In the realm of audio synthesis, researchers initially followed the design used in image generation, and adapted it for audio generation based on mel-spectrogram [41], [42] and waveform [43]–[45]. Due to the involvement of high-dimensional data [46], [47], the representations for waveform and spectrogram can be difficult to train and slow to infer. To address this limitation, recent diffusion-based models work in a latent space through encoding pipelines [27]. Specifically, current state-of-the-art (SOTA) systems follow an encoder-decoder framework [48], trained to generate the target features in a latent space, which are then decoded into waveforms through vocoders [49]–[51].

B. Conditional Audio Generation

Controllable generation has emerged as an important area in audio synthesis, enabling users to guide generative models with specific constraints or conditions to produce desired outcomes [31]. Text-driven audio generation has gained significant attention. AudioGen [17] employs a conditional language modeling pipeline that generates audio waveforms directly. AudioLDM [2] employs the Contrastive Language-Audio Pre-training (CLAP) [52] model to generate the embeddings of audio and text. These embeddings are then used as conditions to guide the training and inference of the latent diffusion model for audio generation. Specifically, AudioLDM is trained to generate feature representations of target audio within the latent space, followed by a variational autoencoder (VAE) decoder to reconstruct the spectrogram from the latent representation. Make-an-Audio [20] develops a pseudo-prompt enhancement strategy to generate extra audio-caption pairs with large-scale compositions to alleviate the data scarcity problem. Tango [19] uses a structure similar to AudioLDM, but replaces the CLAP model with Flan-T5 [53].

Due to the diversity of the training dataset [13], [14], [54], these previous methods allow the model to generate highly diverse samples for the same prompt. However, these models are not designed for CTTA tasks and often struggle to generate content in terms of user preferences [21], which is the problem to be addressed in this paper.

C. Audio Generation with Flow Matching

Flow matching has recently emerged as an efficient alternative to diffusion-based generative models. By learning a deterministic flow path between noise and target distribution [55], flow matching significantly reduces the number of inference steps required for high-quality synthesis, leading to improved efficiency and enhanced performance. FlashAudio [56] is the first audio generation model to employ flow matching, providing fast and high-fidelity results. LAFMA [57] further exploits flow-matching models and reduces the number of inference steps to ten without compromising performance. More recently, TangoFlux [58] and Stable Audio [59] have applied

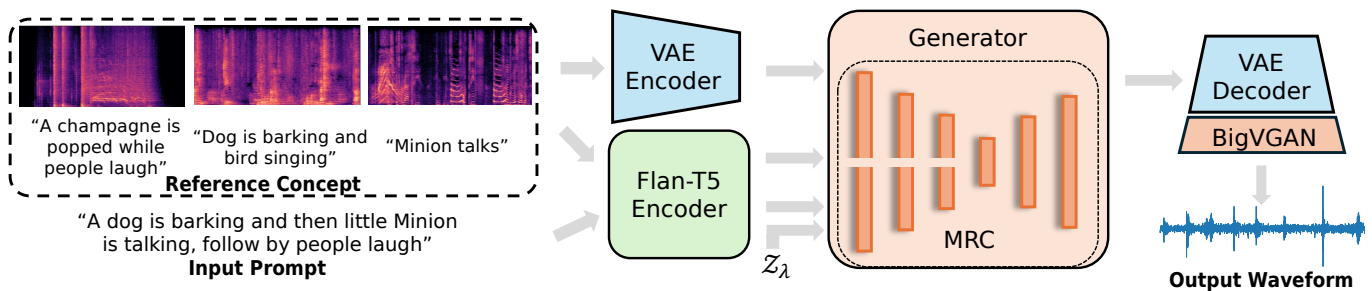


Fig. 2. The inference pipeline of the *DreamAudio*. The input prompt and reference concept are encoded in two paralleled paths through the Flan-T5 Encoder and the reference audio feature is encoded by the VAE Encoder. Along with the noisy data z_λ , four inputs are forwarded to the generator with the MRC structure to generate the denoised data, followed by the VAE decoder and vocoder to reconstruct the final output waveform.

flow matching to large-scale text-to-audio models, achieving SOTA performance on several audio-generation benchmarks. These studies demonstrate that flow matching provides an efficient and scalable framework for audio synthesis, enabling faster sampling and strong performance. Our work builds upon this framework and applies flow matching as the core backbone.

D. Customized Generation

Current methods for customized content generation are developed dominantly for images [60]–[66]. Customized image generation could be categorized into tuning-based customization [23], [67]–[69], and tuning-free customization [62], [70]–[74]. In tuning-based methods, such as DreamBooth [23], the generation models are fine-tuned with the embeddings of specific subjects to control the generated contents. In tuning-free methods, such as Freecustom [73], customization is achieved by integrating the target feature with the generated feature during the inference stages to minimize the fine-tuning processes. Our DreamAudio resembles tuning-free customization methods. While the model itself is trained from scratch to learn general customization capabilities, it does not require any additional adaptation or parameter updates when encountering unseen user-provided reference concepts during inference.

1) *Speech and Music Generation*: Several customized systems have been explored in speech and music generation. For speech generation, ViT-TTS [75] introduces a visual-text encoder that extracts additional visual scene information from reference images to improve text to speech (TTS) generation performance. F5-TTS [76] incorporates reference speech as conditioning signals to control speaking style, prosody, and voice characteristics. In the music domain, Plitsis et al. [77] adapted user-specific concepts from DreamBooth [23] and fine-tune the text embeddings for personalized music style control. Unlike CTTA systems, these existing methods mainly adjust global acoustic attributes such as speaker style, prosody, timbre, genre, or overall musical texture. Their conditioning signals depend on complete speech or full musical phrases, whereas CTTA requires fine-grained, event-level controllability, making it fundamentally different from these style-based generation tasks.

2) *General Audio Generation*: FreeAudio [78] and TG-Diff [79] are two works designed to customize the timing of

sound events in general audio generation. These two models both applied tuning-free customization strategies. More specifically, TG-Diff achieves temporal controllability by learning per-second embeddings, while FreeAudio achieves customization by applying a strategy for decoupling and aggregating attention tokens to fill-in the time-window tokens. However, both systems focus on timing control in text-to-audio generation, which differs from our goal on controlling the acoustic audio events. In addition, these two methods focus on training the time-related embeddings and tokens to match the timing window. Such requirement does not involve zero-shot scenarios, as most temporal patterns can be covered by massive training data with different temporal features.

E. Retrieval-Based Generation

Another task similar to CTTA is retrieval-based TTA, which was developed recently to improve the performance of TTA models in the few-shot and zero-shot cases. An example is Re-AudioLDM [21], which incorporates external audio features into the generation process using retrieval-augmented approaches [80], [81]. This system demonstrates more stable performance on low-occurrence audio events in TTA generation. Building on this concept, AudioBox TTA-RAG [30] introduces a retrieval-information embedding module to enhance the capability in zero-shot generation. Although these systems can improve the TTA generation performance in the few-shot and zero-shot cases, the quality of the audio samples is limited by the database for audio retrieval. Furthermore, these models lack the capability to generate specific content for customization tasks, and research for CTTA tasks remains to be explored.

In this paper, we design a novel CTTA system, which allows the generated audio content to be tailored and controlled in terms of user preference derived from the reference concepts. The ability to achieve customization not only enhances the creative process, but also improves controllability in the event level in TTA systems. Hence, we aim to address a crucial practical challenge for deploying TTA systems in real-world applications.

III. PROPOSED METHOD

A. Overview of the Proposed Model

Given only a set of $k = 1, \dots, K$ audio-language reference pairs $\{(\mathbf{a}_1, \mathbf{t}_1), (\mathbf{a}_2, \mathbf{t}_2), \dots, (\mathbf{a}_K, \mathbf{t}_K)\}$, where each reference audio clip \mathbf{a}_k is associated with a reference caption \mathbf{t}_k , the CTTA task aims to generate the target audio \mathbf{a} conditioned on the input textual prompt c and the reference concepts, derived from pairs of reference audios $\{\mathbf{a}_1, \dots, \mathbf{a}_K\}$ with their corresponding captions $\{\mathbf{t}_1, \dots, \mathbf{t}_K\}$.

To address this task, we propose the DreamAudio system, as illustrated in Figure 2. The system is composed of several modules, including the text encoder (e.g. Flan-T5 [53]) to obtain the embeddings of the input text prompt and reference captions, and the audio encoder (e.g. the variational auto-encoder (VAE) as built in [2]) to obtain the embeddings of the reference audios, which are followed by the feature generator built on rectified flow matching (RFM) [82]. To customize the audio features with reference concepts, a multi-reference customization (MRC) structure is designed for the feature generator. The customized audio embeddings are then converted to spectrograms using a VAE decoder, which are then turned into waveforms using a vocoder (e.g. BigVGAN [51]).

In the remainder of this section, we first discuss the calculation of text and audio embeddings. Then, we present our framework for diffusion-based feature generator, starting with the preliminaries of the module training strategy, followed by the details of the proposed MRC architecture. Finally, we introduce the process for reconstructing the target audio.

B. Text and Audio Embeddings

1) *Text Embedding*: For both the target text prompts and the captions of the reference audio clips, we use the pre-trained Flan-T5 [53] as the text encoder to extract the text feature. Compared to contrastive language pretraining models, such as CLIP [83] and CLAP [52], the Flan-T5 encoder captures both semantic meaning [19] and temporal structures [84] from textual prompts, showing excellent performance in extracting semantic information for text-to-audio generation tasks [19], [21]. Denoting the text prompt as c , the text embedding C is obtained as:

$$C = f_{T5}(c) \quad (1)$$

where $f_{T5}(\cdot)$ is the Flan-T5 text encoder [53].

For each caption of the reference audio concepts, $\mathbf{t}_k, k = 1, \dots, K$, we first use the same Flan-T5 model to compute the embedding as $e_k = f_{T5}(\mathbf{t}_k)$. We then concatenate the embeddings of the reference captions as the reference embedding input E , as follows:

$$E = [e_1, e_2, \dots, e_K]. \quad (2)$$

This enables the model to interact with all the text embeddings of the reference concepts.

2) *Audio Embedding*: We first follow the baseline models [2] by applying a pre-trained VAE encoder $f_{VAE}(\cdot)$ to encode the reference audio clip from the mel-spectrogram into intermediate representations within the latent space. Taking \mathbf{a}_λ as the mel-spectrogram of the original waveform \mathbf{a} , the latent

feature for audio is obtained as $\mathbf{z}_\lambda = f_{VAE}(\mathbf{a}_\lambda)$. Similarly, the latent representation for the mel-spectrogram of referenced audio \mathbf{a}_k is obtained from the VAE encoder [2], as follows:

$$\mathbf{r}_k = f_{VAE}(\mathbf{a}_k) \quad (3)$$

The embeddings $\mathbf{r}_k, k = 1, \dots, K$, are then concatenated into \mathbf{R} , shown as input in Figure 3:

$$\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_K] \quad (4)$$

C. Audio Feature Generation

1) *Rectified Flow Matching*: Traditional diffusion-based generative models, such as DDPMs [11], generate samples through iterative denoising from a Gaussian prior, requiring hundreds of steps to gradually turn noise into the target data. However, such approaches are highly dependent on noise schedules and can be slow in sampling. RFM [82], on the other hand, addresses the sampling issue and increases stability by learning a continuous flow between noise and data distributions, as demonstrated in audio related tasks [55], [85].

Unlike DDPMs that operate on discrete time steps $t \in \{0, 1, \dots, N\}$, RFM models introduce a continuous flow variable $\lambda \in [0, 1]$ that smoothly interpolates between Gaussian noise $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and the target data representation \mathbf{z}_1 . By replacing discrete time steps with a continuous flow parameter λ , RFM enables smooth noise-to-data interpolation and a stable, schedule-free training objective based on a constant velocity field \mathbf{v} . The noisy data \mathbf{z}_λ at the flow parameter of λ is defined by a linear equation:

$$\mathbf{z}_\lambda = (1 - \beta\lambda)\mathbf{z}_0 + \lambda\mathbf{z}_1 \quad (5)$$

where $\beta = (1 - \sigma)$ is a small positive constant (e.g., $\sigma = 1 \times 10^{-5}$) introduced to avoid degeneracy and to improve numerical stability. This formulation allows the model to define a continuous transformation path between the noise and the target.

The model aims to estimate this velocity field \mathbf{v} , which defines the direction and magnitude of the continuous transformation that transports samples from the noise distribution to the data distribution. To train the model, a neural network $\mu(\cdot)$, corresponding to the proposed MRC module, is optimized to predict the velocity field $\mathbf{v} = \mathbf{z}_1 - \beta\mathbf{z}_0$. Unlike denoising targets in general diffusion models, this velocity field is independent of λ , which provides a consistent supervision signal in all flow positions sampled and improves training stability. The network is trained to minimize the following objective:

$$L_{RFM}(\theta) = \mathbb{E}_{\lambda, \mathbf{z}_1, \mathbf{z}_0} \|\mu(\mathbf{z}_\lambda, \mathbf{R}, \lambda, \mathbf{E}, C) - \mathbf{v}\|^2 \quad (6)$$

Here, the network is conditioned on multiple modalities, including the reference audio features \mathbf{R} , the text embedding E , prompt C and the continuous flow parameter λ , thus enabling fine-grained control over the generation.

During inference, an ordinary differential equation (ODE) of the form $\frac{d\mathbf{z}_\lambda}{d\lambda} = \mu(\mathbf{z}_\lambda, \mathbf{R}, \lambda, \mathbf{E}, C)$ is solved using the numerical ODE solver [82], starting from Gaussian noise \mathbf{z}_0 and integrating from $\lambda = 0$ to $\lambda = 1$ to obtain the final

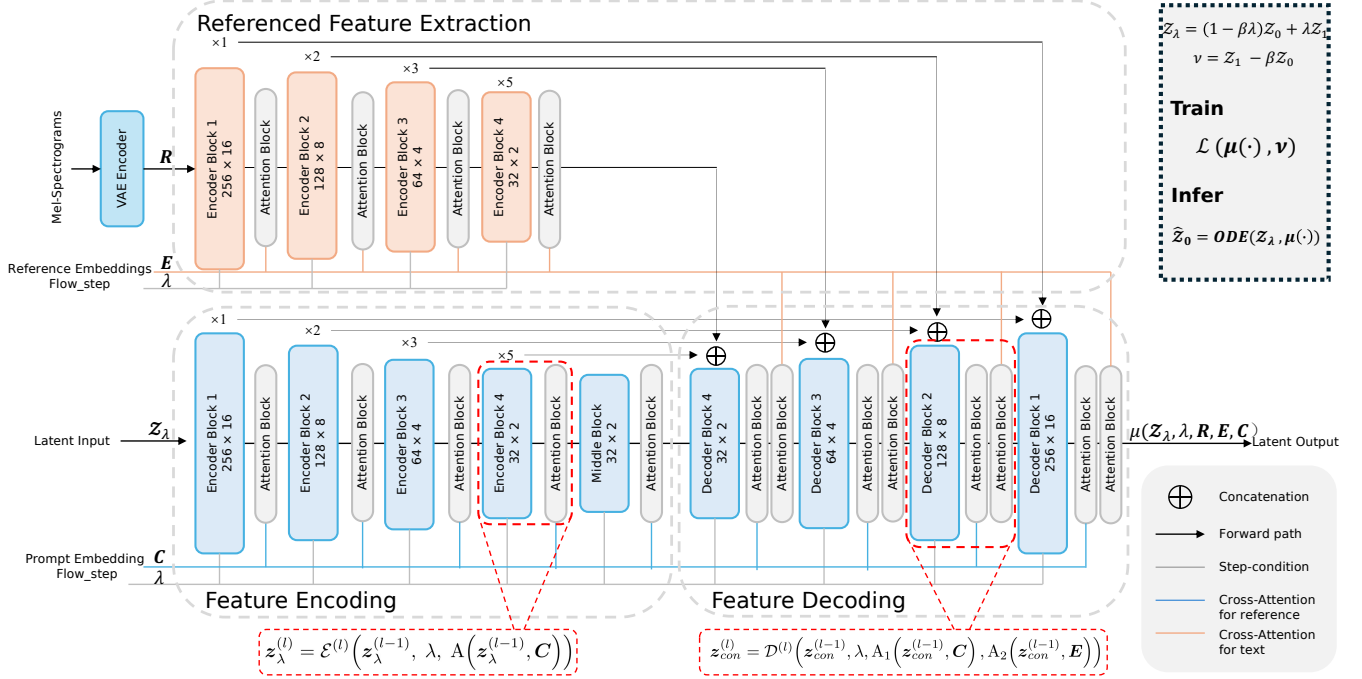


Fig. 3. The details of the MRC module, which takes the reference feature R and E , prompt feature C and the current noisy data z_λ as inputs to generate the dynamics for denoised data z_1 on step λ . The output $\mu(\cdot)$ can then be used for both training and inference.

sample z_1 . Compared to the hundreds or even thousands of iterations required in diffusion-based sampling [10], this process typically requires fewer than 50 iterations to synthesize high-quality audio.

2) *Multi-Reference Customization*: To produce customized audio features that incorporate referenced concepts without requiring fine-tuning, we designed the MRC structure. As shown in Figure 3, the MRC module merges textual and acoustic inputs, drawn from the input prompt and referenced concepts, and predicts the necessary dynamics to yield the denoised output z_1 .

The design draws inspiration from ControlNet [31], which adds a conditional encoder to steer the decoder. However, as illustrated in Figure 3, MRC adopts a U-Net backbone with dedicated encoder blocks in the down-sampling stage to process multiple referenced concepts. These new encoder blocks are externally linked to corresponding decoder blocks during up-sampling, enabling feature integration at multiple scales. Unlike ControlNet, which typically processes spatially aligned visual conditions (e.g., edges or depth maps), MRC handles multiple audio-text references with an external encoder. These references are fed into the generation pipeline via two separate mechanisms: extracted audio latent features are fused directly with the input of each decoder block for feature-level guidance, while text features are directed to cross-attention layers to ensure semantic alignment.

Rather than merely copying information from the references, MRC learns to identify relevant audio-event characteristics from multiple sources and blend them into the target generation. Specifically, the down-sampling stage comprises two parallel encoder paths: one for the generated feature vector and the other for the referenced feature vector. Their outputs

are concatenated before being passed to a unified up-sampling decoder, ultimately producing the final latent representation.

Feature Encoding Path. The encoder layers take the noisy latent representation z_λ and the text prompt embedding C as input. Formally, for each encoding block $\mathcal{E}^{(l)}(\cdot)$, the latent feature is updated along with the flow parameter λ as

$$z_\lambda^{(l)} = \mathcal{E}^{(l)}\left(z_\lambda^{(l-1)}, \lambda, A\left(z_\lambda^{(l-1)}, C\right)\right) \quad (7)$$

where $z_\lambda^{(0)} = z_\lambda$ denotes the input noisy latent feature, $\mathcal{E}^{(l)}(\cdot)$ represents the l -th encoding block, and $A(\cdot)$ denotes the attention operation that injects text conditioning information C into the latent representation.

Reference Feature Extraction Path. The encoder layers for reference feature extraction, on the other hand, take the latent representation of the reference audio concept R and the corresponding textual embedding E . Applying the same equation (7), this approach allows *DreamAudio* to identify and extract the audio feature for the customized content based on the reference audio and their corresponding captions. It is noted that the two groups of encoder blocks share the same structure, e.g., convolutional layers with a cross-attention module for text embedding, but are applied with independent weights for different downsampling purposes, respectively.

Feature Decoding Path. We feed all the features and conditions through a group of decoder blocks, designed as the feature decoding path. Specifically, during the up-sampling stages, the latent vector from the feature encoding path and the feature extraction path are concatenated with the audio feature from each scaling level in downsampling stages via skipped connections, before being passed into each decoder block. Different from single text embedding conditions in the feature encoding path, both the target prompt C and reference captions E

are given as conditions through two cross-attention blocks respectively. The updated latent state for the l -th decoding block $\mathcal{D}^{(l)}$ is formed as:

$$\mathbf{z}_{con}^{(l)} = \mathcal{D}^{(l)}\left(\mathbf{z}_{con}^{(l-1)}, \lambda, A_1\left(\mathbf{z}_{con}^{(l-1)}, \mathbf{C}\right), A_2\left(\mathbf{z}_{con}^{(l-1)}, \mathbf{E}\right)\right) \quad (8)$$

where $\mathbf{z}_{con}^{(l-1)} = [z_{\lambda}^{(l-1)}; \mathbf{R}^{(l-1)}]$ is the concatenation of the latent state from both the feature encoding path and the reference feature extraction path. $A_1(\cdot)$ and $A_2(\cdot)$ are the two blocks that calculate the cross-attention for the prompt embedding and the reference embedding, respectively.

D. Audio Feature Reconstruction

As shown in Figure 2, *DreamAudio* leverages both the VAE decoder and a generative adversarial network (GAN)-based vocoder for reconstructing the audio feature from latent space into the target waveform. Followed by the RFM model, the generated audio representation is first decoded into the mel-spectrogram by the VAE decoder. In the next stage, a vocoder is applied to convert the audio feature into the waveform as the final output, and we train the vocoder using the SOTA structure BigVGAN [51] in the proposed system.

E. Customized Data Processing

In the early training stages, we observed that providing excessively detailed audio features caused the model to become overly reliant on these given representations, thereby limiting its ability to independently learn and generate new audio characteristics. To mitigate this, augmentation strategies are implemented to enhance the generalisation ability beyond the provided references. First, we randomly mask the contents of the customised audio event for the target audio. Second, we randomly remove the customised content during training. We train the *DreamAudio* model with a 10% content masking and a 40% content dropping rate. These augmentation strategies, i.e. masking and dropping, are applied to both reference audio and reference text to improve the model’s robustness. Experiments on the effectiveness of these masking and dropping metrics are given in Section VI-D4.

IV. DATASETS AND EVALUATION BENCHMARK

Previous text-to-audio generation models [2], [16], [19] mainly work on public audio-language datasets [13]–[15]. These datasets only provide audio clips and their corresponding captions. In customized generation tasks, additional reference concepts, including reference audio clips and corresponding captions are also required. To this end, we construct the customized training and evaluation datasets based on four commonly used audio-language datasets. The construction of these datasets enables *DreamAudio* to extract the customized feature from specific user contents. In addition, we manually collect a small-scale dataset with real-world customized scenarios as the first benchmark for CTTA. In the following sections, we first introduce the datasets collected as the base dataset to form the customized datasets. Then, we discuss the strategies for developing the training and testing data, followed by the introduction of the benchmark dataset. All evaluation benchmarks used for the system are presented in the end.

A. General Datasets

1) *AudioCaps*: AudioCaps (AC) [54] is one of the largest audio datasets with hand-crafted captions. As a subset of the AudioSet [13], AudioCaps contains 52,905 10-second audio clips, and each clip is matched with a human-annotated caption. Taking from original videos on YouTube, AudioCaps contains various classes of audio, including music, Foley sounds [61], and human speech. Each audio clip has a single caption in the training set and five captions in the test set, where we only apply the first caption of each waveform to the testing data. We follow the official dataset split, and build the training set of 49,502 audio clips and the testing set of 928 clips.

2) *WavCaps*: WavCaps (WC) [14] is a machine-labeled dataset with audio captions generated through Large Language Models (LLMs). WavCaps contains 403,050 audio clips collected from various datasets such as AudioSet [13] and FreeSound [86], providing various durations of audio clips ranging from 0.1s seconds to 68 seconds. To align the length of the input and target audio clips, we only collect the clips shorter than 10 seconds, resulting in a group of 163,818 audio clips. Then, we randomly selected 5% for testing, forming a set of similar size to the AudioCaps test set.

3) *UrbannSound8K*: UrbanSound8K (UB8K) [87] contains 8,732 labeled audio clips and each clip is shorter than 4 seconds. The dataset is divided into 10 different classes, including urban noise, background sound sources, and natural sound sources. We randomly selected 732 clips for model testing, similar to the size of the AudioCaps test set, and the remaining 8000 clips for model training.

4) *ESC-50*: ESC-50 (ESC) [88] has 2,000 5-second audio recordings natural sounds and domestic sounds. These samples are evenly categorized into 50 distinct classes. To match a similar data scale to AudioCaps, we randomly chose 400 audio clips for model testing and 1,600 clips for model training.

B. Customized Datasets

With the four general audio datasets mentioned above, we design three task-specific data generation pipelines to construct customized datasets for training and evaluation. Unlike standard audio augmentation techniques, these pipelines are specifically designed for the CTTA task to simulate event-level customization. An overview of each customized dataset is presented in Table I, and the corresponding data generation pipelines are illustrated in Figure 4.

1) *Customized-Concatenation*: We develop the dataset by concatenating audio events. In detail, we collect audio clips shorter than 5 seconds from the database and then randomly form a group of audio clips whose lengths sum up to 10 seconds. The target audio is then generated by concatenating these short audio clips. Next, we construct the referenced audio by grouping the short audio clips into three sets, each set is concatenated into a single clip as the reference audio. For the reference captions, we first convert the labels from US8K and ESC-50 into short captions by simply adding verbs and subjects. Then, we generate sentences by connecting the

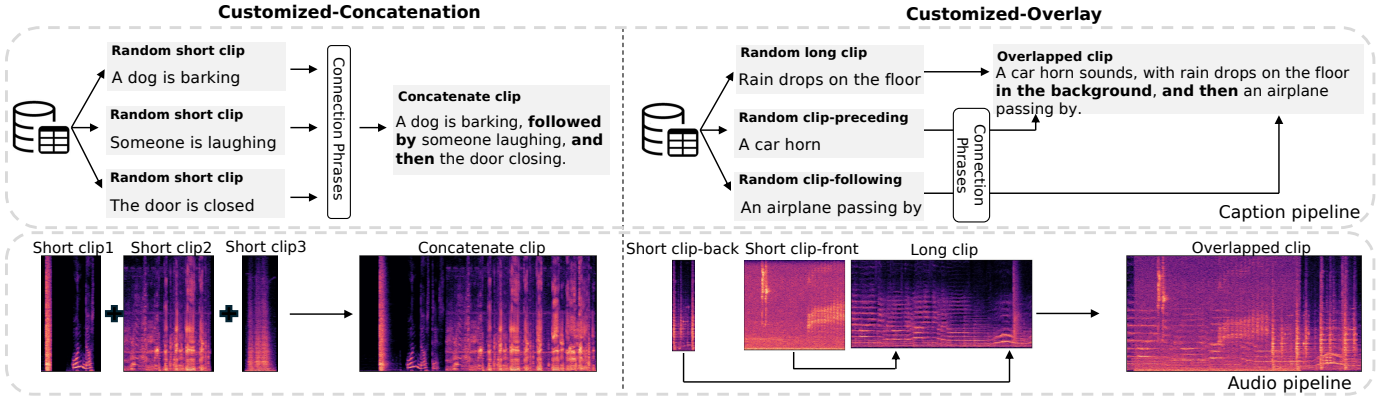


Fig. 4. The generation pipeline of the customized datasets, with the Customized-Concatenation on the left and Customized-Overlay on the right. All the clips are selected randomly from the base dataset and both the concatenation clips and overlapped clips are fixed into 10-seconds.

TABLE I

THE SETUP OF FOUR DATASETS USED FOR TRAINING AND TESTING OF THE PROPOSED DREAMAUDIO. THE REFERENCE-TYPE INDICATES THE STRATEGY FOR GENERATING THE REFERENCE CONCEPT AND THE REFERENCE-NUM REPRESENTS THE NUMBER OF AUDIO EVENTS FOR CUSTOMIZATION THROUGH THE REFERENCE CONCEPT.

Dataset	General-Dataset	Customized-Content	Train-Num	Test-Num	Reference-Type	Referenced-Num
AudioCaps-General	AudioCaps	✗	49,502	928	Retrieval-based	✗
Customized-Overlay	WC+UB8K+ESC	✓	146,481	200	Overlap-audio	1 – 3
Customized-Concatenation	AC+WC+UB8K+ESC	✓	92,299	200	Concate-audio	1 – 5
Customized-Fantasy	Online-Collect	✓	–	25	Concate-audio	3

caption of each short audio clip with a connection phrase randomly selected from our connection list. For the Customized-Concatenation dataset, we collect 92,300 audio-text pairs for model training and 200 pairs for model testing.

2) *Customized-Overlay*: We found that the simple concatenating strategy led to several limitations. First, not all audio events can be connected smoothly, which may leave a large blank space in the synthesized waveforms. Second, audio created solely by concatenation often sounds unnatural, failing to accurately reproduce user-provided reference contents, and thereby not reflecting realistic scenarios. To guide the model with more realistic cases, we develop a second version of the customized dataset, called Customized-Overlay dataset. Specifically, for each audio sample, we take a 10-second audio clip as the base waveform and randomly select two shorter clips (less than 5 seconds) as the “preceding” and “following” audio clips. The target audio clip is then generated by adding the “preceding” and “following” audio events into the base waveform. Then, we remix the waveform of two extended clips and the base clips under a random signal to noise ratio (SNR) between -15 and 15 dB. The captioning strategy is similar to the approach we apply in the Customized-Concatenation dataset. We generated 146,481 pairs for training and 200 pairs for testing in the Customized-Overlay dataset.

3) *Customized-Fantasy*: We manually created a small-scale dataset as a benchmark for this new task, named as Customized-Fantasy. All the data clips are collected from online sources, including many unique and special events extracted from games or movies to simulate the user cases for real-world audio-production scenarios. For example, customization in *laser guns*, *monster roars* and *Minion speaks*.

In addition, human interactions are involved to ensure that the captions are meaningful and present semantic coherence. In total, we collected 60 different events from Pixabay¹, a website which provides royalty-free audio resources under the Pixabay License suitable for research purposes. Together, we manually developed 25 different scenarios to simulate real-world customized generation.

4) *AudioCaps-General*: To maintain the capability of the system in more robust situations, e.g., prompting on audio events without the reference concepts. We apply the general AudioCaps dataset with empty reference concepts. For instance, the model is given with the target audio clip and the target caption, along with three empty waveform and empty text as the reference concept. The same training-testing split is applied for training and evaluation of the model.

C. Evaluation Metrics

Following the evaluation protocol of baseline audio generation models [22], we use three different metrics for performance evaluation, including the Fréchet Audio Distance (FAD), Kullback-Leibler (KL) divergence and Contrastive Language-Pretraining (CLAP) related score. In addition, we follow the evaluation pipelines from the customized image generations [23], [73] and incorporate subjective assessments that measure the consistency and the fidelity between generated and target audio, including CLAP audio score (CLAP_A), AudioBox Aesthetics, and subjective evaluations. Specifically, audio-text latent similarity based metrics including FAD, KL and CLAP-score are applied for evaluating

¹<https://pixabay.com/>

TABLE II

THE SETUP OF THE PRIMARY EXPERIMENTS WE PERFORMED, WITH DDPM FOR USING DDPM-BASED TRAINING STRATEGIES, UNET FOR BASIC DIFFUSION-BASED STRUCTURE, AND DiT FOR DIFFUSION-TRANSFORMER BASED BACKBONE. *DreamAudio* WITH DUAL-ENCODER AND SINGLE-ENCODER INDICATES WHETHER THE TWO GROUPS OF ENCODER BLOCKS SHARE THE SAME WEIGHTS.

Model	Param(train)	Diffusion	MRC
DreamAudio-DDPM	760M	DDPM	Dual-Encoder
DreamAudio-UNet	760M	RFM	X
DreamAudio-DiT-B	920M	RFM	Dual-Encoder
DreamAudio-DiT-L	1.3B	RFM	Dual-Encoder
DreamAudio-SEncoder	815M	RFM	Single-Encoder
DreamAudio	891M	RFM	Dual-Encoder
DreamAudio-L	1.1B	RFM	Dual-Encoder

the performance on Customized-Concatenation, Customized-Overlay and general AudioCaps testing sets, subjective evaluations and AudioBox Aesthetics are used for the evaluations on Customized-Fantasy.

1) *FAD*: FAD first computes the multivariate Gaussian of two groups of embedding values from a pre-trained VG-Gish [89] feature extraction model. Then, it computes the Frechet distance between the Gaussian mean and variance of two sets of high-dimensional features. A lower FAD score indicates that the generated audio group presents a closer data distribution to the target audio group based on audio features.

2) *KL Divergence*: KL measures the logarithmic difference between the probabilities assigned by the distributions of two audio clips across all possible events. We apply the audio classification model PANNs [90] for feature extraction and calculate the KL score between each target audio sample and their corresponding outputs. A lower KL represents a smaller distance in distributions, suggesting that the generated outputs are more similar to the target clips.

3) *CLAP*: The CLAP score calculates the cosine similarity between the text embedding and the audio embedding provided by the CLAP model [52]. The CLAP model learns projectors to align the audio and text embeddings into a joint space and presents paired audio and language data with similar features within the latent space. Given both generated audio embedding \mathbf{e}_a and text embedding \mathbf{e}_t , the score is calculated as:

$$\text{CLAP}(\mathbf{e}_a, \mathbf{e}_t) = \frac{\mathbf{e}_a \cdot \mathbf{e}_t}{\max(\|\mathbf{e}_a\| \|\mathbf{e}_t\|, \epsilon)}, \quad (9)$$

where ϵ is a small value to avoid zero division. The CLAP score illustrates the correspondence between the generated audio and text prompt, and a higher score shows better performance on the semantic level.

4) *CLAP_A*: Instead of calculating the similarity between audio and text embeddings, the CLAP_A score is specially designed for the CTTA task to compare the customized concepts between the targets and generated audio by calculating the cosine similarity score between the target audio embedding \mathbf{e}_a and the generated audio embedding $\mathbf{e}_{\hat{a}}$,

$$\text{CLAP}_A(\mathbf{e}_a, \mathbf{e}_{\hat{a}}) = \frac{\mathbf{e}_a \cdot \mathbf{e}_{\hat{a}}}{\max(\|\mathbf{e}_a\| \|\mathbf{e}_{\hat{a}}\|, \epsilon)}. \quad (10)$$

A higher score means that the generated output is more similar to the target audio within the embedding space.

5) *AudioBox Aesthetics*: AudioBox Aesthetics is a recently proposed audio quality assessment tool that has demonstrated a strong correlation with subjective evaluations by human listeners [91]. In detail, this metric applies a transformer-based model for extracting and assessing the quality of audio clips without reference clips. In this paper, we choose the production quality (PQ) and content usefulness (CU) for the task. In detail, PQ is an objective aspect of the overall quality and CU is a subjective axis to evaluate the likelihood of audio sample for content creation.

6) *Subjective Evaluation*: We adopt the evaluation metrics used in the baseline models [22], incorporating both Overall Impression (OVL) and Audio-Text Relation (REL) for subjective assessments. Detailed scoring instructions, including illustrative examples, are provided to the raters to ensure clarity. For OVL, we follow the approach outlined in the baseline models [22] by asking raters: *How would you rate the overall quality of this audio sample?*. Responses are scored on a Likert scale, ranging from 5 for “excellent” to 1 for “bad”. For REL, we adapt the metric to better suit the customized generation tasks by asking: *Does the generated audio successfully present the target audio events customized by the reference audio samples?* The subjective evaluation was conducted with ten human raters recruited through an open advertisement within the Department of Electrical and Electronic Engineering at the University of Surrey. All participants were PhD students and were not involved in the development of the proposed model or the writing of this paper, and none of the authors participated in the evaluation. To encourage diverse perspectives and reduce potential bias, the participant pool included six raters with backgrounds in audio related research and four raters from unrelated fields such as computer vision and robotics, providing complementary non-expert assessments.

V. EXPERIMENTAL SETTING

A. Model Architecture Details

For the text encoder, we use a pre-trained Flan-T5 model [53], resulting in a 1024 dimension feature sequence with a fixed length of 50 for every caption. We use the pre-trained VAE model from AudioLDM [2] which provides a compression ratio of 4 and results in a latent vector of 256 dimension in temporal and 16 dimension in frequency for a ten-second mel spectrogram. For the vocoder, we apply BigVGAN [51] and perform the self-supervised pre-training on audio clips sampled at 16 kHz. Table II summarizes our experiments. We perform the experiments with two sizes of the latent diffusion model, *DreamAudio* and *DreamAudio-L*, with the hidden dimension size of $n_{\text{hidden}} = 96$ and $n_{\text{hidden}} = 128$, respectively. In addition, we evaluated several variants of the proposed system. The *DreamAudio* trained under a DDPM framework is denoted as *DreamAudio-DDPM*. The version using a standard UNet backbone but without the MRC module is named *DreamAudio-UNet*. In this system, rather than using the dual-path MRC structure, all three reference audio latent features are concatenated together with the latent of the noisy input at the input layer, extending the input dimension to align with the MRC process. The system with shared-weight

TABLE III
COMPARISON OF MODEL PERFORMANCES ON THE CUSTOMIZED-BASED EVALUATION SET. AC IS SHORT FOR AUDIOCAP DATASET, CM IS FOR CUSTOMIZED-CONCATENATION AND CE IS FOR CUSTOMIZED-OVERLAY.

Model	Dataset	Customized-Concatenation				Customized-Overlay				Customized-Fantasy			
		FAD ↓	KL ↓	CLAP ↑	CLAP _A ↑	FAD ↓	KL ↓	CLAP ↑	CLAP _A ↑	PQ ↑	CU ↑	OVL ↑	REL ↑
Re-AudioLDM	AudioCaps	3.05	3.24	39.4	48.7	2.96	3.09	44.3	47.8	5.80	5.05	3.15	3.26
DreamAudio-DDPM	AC+CM+CE	0.94	1.09	52.3	79.8	0.99	1.32	46.7	83.3	6.18	5.45	3.70	3.53
DreamAudio-UNet	AC+CM+CE	1.59	1.73	48.1	73.5	1.03	0.87	41.8	77.8	5.59	4.78	3.41	3.39
DreamAudio-DiT-B	AC+CM+CE	0.84	1.18	50.5	78.9	0.94	1.05	41.9	79.3	6.22	5.39	3.55	3.51
DreamAudio-DiT-L	AC+CM+CE	0.49	0.95	52.1	84.9	0.81	0.90	43.5	82.1	6.42	5.85	3.79	3.85
DreamAudio-SEncoder	AC+CM+CE	0.79	1.05	51.4	85.9	0.88	0.92	41.5	81.3	6.12	5.31	3.51	3.69
DreamAudio	AC+CM+CE	0.50	0.90	52.0	86.3	0.78	0.82	42.0	81.7	6.31	5.56	3.74	3.91
DreamAudio-L	AC+CM+CE	0.46	0.92	52.5	87.7	0.73	0.67	42.4	83.9	6.37	5.65	3.89	4.17

MRC encoders is denoted as *DreamAudio-SEncoder*, where the encoder blocks used for referenced feature extraction and target feature encoding share the same encoder. This encoder is simply run multiple times for the target latent and each referenced latent during training and inference. Beyond the UNet-based architectures, we also experimented with two models that adopt a diffusion-transformer (DiT) backbone [92]. In *DreamAudio-DiT-B*, we employ 12 transformer layers, while in *DreamAudio-DiT-L*, we use 24 transformer layers. Both models incorporate skip connections, and first half of the layers are used as the encoders while the second half is applied as the decoders.

B. Training and Inference Setup

Despite the pre-trained text encoder and VAE, we train the model and vocoder separately. We first train the vocoder on a large-scale audio dataset for 1M steps and then freeze the module while we train the proposed *DreamAudio*. We collected 288,283 clips for model training and 1328 clips for model evaluation by combining Customized-Concatenation, Customized-Overlay, and general AudioCaps datasets mentioned in Section IV-B. For each audio clip, we provide three-pairs of reference concepts as the external data. The system is trained on a single NVIDIA A100 80GB GPU for 2 millions steps. Following similar strategies from generative networks, we apply the Classifier Free Guidance (CFG) for RFM with a value of 2.0 for both *DreamAudio* and *DreamAudio-L*. We utilize the AdamW [93] optimizer with a learning rate of 5×10^{-5} for training and apply a linear warming up for 10,000 steps.

VI. RESULTS AND ANALYSIS

We evaluated DreamAudio on both customized and general text-to-audio generation. The following section first discusses the performance of our system on the CTTA tasks and then explores the capability of general TTA tasks.

A. Customized Text-to-Audio Generation

All the models in Table III are evaluated on both Customized-Concatenation and Customized-Overlay datasets. We compare our method with Re-AudioLDM [21] as this is the only work related to customized and zero-shot tasks. The result for AudioBox-TTA-RAG [30] is not included

because the authors did not provide any model or checkpoint. For Re-AudioLDM, we apply the system with the retrieval number of 3 as the retrieval-based information. As shown in Table III, the proposed *DreamAudio* significantly outperforms the previous systems across most of the metrics. The baseline system, Re-AudioLDM, achieves a FAD score of 3.05 and 2.96 for Customized-Concatenation and Customized-Overlay, respectively. Our systems, on the other hand, illustrate a substantially enhanced score of 0.46 for the concatenated testing set and 0.73 for the overlapped testing set. *DreamAudio-L* also achieves the best KL divergence score of 0.92 and 0.67 on the two datasets, respectively.

For the CLAP score, our system demonstrates the best performance on the Customized-Concatenation dataset and achieves comparable results on the Customized-Overlay dataset. Unlike other objective metrics that directly compare target and generated audio samples, the CLAP score measures semantic alignment rather than precisely evaluating the presence of customized audio content. In this case, it is less effective in evaluating the CTTA task. In contrast, the CLAP_A score evaluates the similarity between the features from each generated audio clip and its corresponding target audio clip, making it a more accurate metric for reflecting the correctness of customized content. *DreamAudio-L* achieves a CLAP_A score of 87.7 on the Customized-Concatenation dataset and 83.9 on the Customized-Overlay dataset, significantly surpassing all previous models. For the Customized-Fantasy testing set, *DreamAudio* achieves the best performance among production quality, content usefulness and human evaluation, where *DreamAudio-L* achieves better results on OVL of 3.89 and REL of 4.17. The significant enhancement of these metrics illustrates that DreamAudio can generate specific audio features related to the reference concepts and be faithful to text prompts.

B. General Text-to-Audio Generation

In this section, we evaluate the capability of *DreamAudio* on general text-to-audio generation tasks. We compare the performance on audio generation with several SOTA systems, including AudioLDM [2], AudioGen [17], Make-an-Audio [20], Stable Audio Open [59], Tango2 [19], AudioLDM2 [22], TangeFlux [58], and Re-AudioLDM [21]. All the checkpoints are downloaded from open-sourced GitHub or HuggingFace to reproduce the results. For plain text-to-audio

TABLE IV

COMPARISON OF MODEL PERFORMANCES ON THE AUDIOCAPS EVALUATION SET. AC IS SHORT FOR AUDIOCAPS DATASET, CC IS FOR CUSTOMIZED-CONCATENATION AND CO IS FOR CUSTOMIZED-OVERLAY. THE DATASET MARKED WITH * INDICATES FINE-TUNING AND † MEANS TRAINING FROM SCRATCH ON AUDIOCAPS.

Model	Dataset	FAD ↓	KL ↓	CLAP ↑	CLAP _A ↑
AudioLDM	AC+AS+2 others	5.25	1.90	42.1	53.5
AudioGen	AC+AS+8 others	2.87	1.52	46.4	60.2
Make-an-Audio	AC+AS+13 others	2.39	1.64	45.4	59.8
Stable Audio Open	FreeSound	4.05	2.11	44.9	55.2
Tango	AudioCaps	2.24	1.04	51.7	66.2
AudioLDM2	AC+AS+6 others	2.56	1.75	45.8	55.5
TangoFlux	WavCaps+2 others	2.33	1.09	50.4	59.5
Re-AudioLDM	AudioCaps	1.85	1.46	49.9	62.0
DreamAudio	AC+CC+CO	4.25	2.48	34.9	43.6
DreamAudio	AudioCaps*	1.92	1.51	47.5	58.8
DreamAudio	AudioCaps†	1.90	1.50	50.8	62.5

tasks without any reference inputs, DreamAudio is provided with empty reference audio clips and empty reference text token. As shown in Table IV, DreamAudio trained on reference-based data does not achieve SOTA performance when directly applied to general text-to-audio generation tasks. We believe the main reason for this is that DreamAudio is architecturally designed for customized generation with reference-based conditions. The MRC module provides an additional pathway for extracting and leveraging external audio/text reference signals. When reference data are available during training stages, the model learns to rely on these external concepts for capturing acoustic feature and fine-grained attributes. This makes the model less dependent on the text conditions while generating audio. Consequently, when operating on general tasks without reference concepts, the external pathway of MRC is un-used. Hence, the performance is limited since the model has been guided by the conditions from both reference concepts and text prompts, rather than purely text prompts.

After fine-tuning DreamAudio on AudioCaps, the model recovers strong performance and achieves comparable results to most non-retrieval based systems. Furthermore, when DreamAudio is trained from scratch with only AudioCaps and no reference-based data, the model exhibits even better performance, demonstrating that the architecture itself remains fully capable of high-quality general text-to-audio generation. These findings indicate that the system can be applied to general text-to-audio generation tasks by reducing the conditioning of the reference concepts.

TABLE V

EXPERIMENTAL RESULTS FOR CUSTOMIZED GENERATION WITH FOUR REFERENCE CONCEPTS, WHERE THE MODELS ARE OBTAINED BY FINE-TUNING THE MODEL TRAINED WITH 3 REFERENCES. THE STEP NUMBER WITH * INDICATES THAT THE MODEL IS FULLY-TRAINED WITH 3 REFERENCES AND WITHOUT ANY FURTHER FINE-TUNING.

Reference Num	Steps	FAD ↓	KL ↓	CLAP ↑	CLAP _A ↑
3	0*	0.50	0.92	52.5	87.7
4	0*	14.31	6.01	34.9	43.6
4	10,000	3.86	3.01	45.6	69.5
4	50,000	0.79	1.15	49.2	77.5

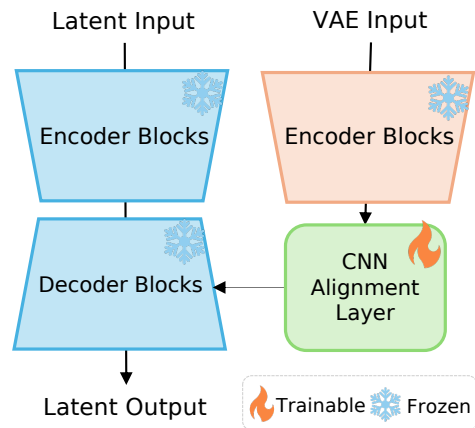


Fig. 5. The details of the MRC UNet network for reference length fine-tuning. All the existing blocks are frozen and only the introduced CNN alignment layer is trained during this stage.

C. Customized Generation on Various Number of References

Our DreamAudio is primarily trained and evaluated with three-pairs of reference concepts. In this section, we investigate the capability of *DreamAudio* to perform customized generation with four pairs of reference concepts. The current backbone adopts a CNN-based U-Net architecture, where all intermediate latent feature maps within the decoder must maintain fixed spatial dimensions to allow their concatenation through skip connections. In this case, the referenced VAE latent features need to be projected into a fixed-size representation, e.g. 96 in the temporal dimension and 6 in the frequency dimension for three reference pairs. When the system is given more than three-pairs of reference concepts, the dimension of reference latent feature provided by the feature extraction path does not match with the designed input. To address this issue, we introduce a lightweight CNN alignment layer that maps the reference VAE inputs with variable temporal lengths into this fixed-size latent space, as shown in Figure 5.

During fine-tuning, all backbone encoder-decoder blocks are frozen and only the CNN alignment layer is trained to adapt to the scenarios with different numbers of reference concepts. The performance results of *DreamAudio* trained or fine-tuned with four-pairs referenced concepts are shown in Table V. The first row shows the performance of a fully-trained DreamAudio model using 3 reference concepts. In contrast, a model using 4 references without any fine-tuning (0 steps) fails, as its untrained CNN alignment layer cannot properly reshape or compress the four latent inputs into a compatible, fixed-size representation. This results in meaningless outputs and a sharp drop in performance. However, after just 10,000 fine-tuning steps, the alignment layer learns to effectively project the four references into a unified latent space, adapting quickly to the additional input. Performance continues to improve with extended training, achieving a competitive FAD score of 0.79 after 50,000 steps. These results demonstrate that DreamAudio can readily generalize to different numbers of reference concepts by integrating a simple, trainable CNN alignment layer.

TABLE VI

ABLATION STUDIES ON DATA AUGMENTATION (MASKING, DROPPING) AND THE EFFECTIVENESS OF DIFFERENT PROPORTIONS OF TRAINING DATA. THE GENERAL TASK IS EVALUATED ON THE AUDIOCAPS TESTING SET AND CTTA IS EVALUATED ON THE CUSTOMIZED-CONCATENATION AND CUSTOMIZED-OVERLAY TESTING SETS. FOR DATA PROCESSING, AC IS SHORT FOR AUDIOCAPS, AND CC AND CO ARE SHORT FOR CUSTOMIZED-CONCATENATION AND CUSTOMIZED-OVERLAY. THE NUMBER INDICATES THE QUANTITY OF EACH DATASET FOR TRAINING. THE RESULT WITH † INDICATES THE EXPERIMENTS WITHOUT REFERENCE TEXT CONCEPTS.

Model	Training Data-Mixing				General Text-to-Audio Task				Customized Text-to-Audio Task				
	AC	CC	CO	Masking	Dropping	FAD ↓	KL ↓	CLAP ↑	CLAP _A ↑	FAD ↓	KL ↓	CLAP ↑	CLAP _A ↑
DreamAudio	49K	92K	146K	10%	40%	4.25	2.48	34.9	43.6	0.64	0.86	47.0	84.0
				0%	0%	5.28	2.65	31.2	40.4	0.51	0.56	47.7	88.1
				10%	10%	4.28	2.25	35.2	44.4	0.56	0.71	47.2	84.3
				50%	90%	3.81	2.15	38.1	46.9	1.32	1.68	42.5	74.3
				—	100%†	3.69	2.18	40.3	47.5	3.11	2.15	39.6	66.8
	24K	92K	146K			5.58	2.69	30.1	39.2	0.64	0.88	47.2	84.9
	49K	24K	24K			2.85	1.82	41.5	51.1	2.11	1.96	41.9	68.5
	49K	92K	24K	10%	40%	5.14	2.68	33.9	42.1	0.48	0.61	47.8	86.5
	49K	24K	146K			4.03	2.11	36.8	44.6	0.88	1.01	44.5	80.2

D. Ablation Studies

In order to validate our design choice of the proposed system, a series of ablation experiments were conducted on each proposed technique within *DreamAudio*.

1) *Effectiveness of RFM*: As shown in Table III, we compare the outputs of *DreamAudio* with different training approaches on various datasets. Experiments on RFM-based systems present enhanced performance on customized tasks, i.e., Customized-Concatenation and Customized-Overlay. On the other hand, we observe that *DreamAudio-DDPM* trained with diffusion-based techniques offers better results in the semantic feature than *DreamAudio* by achieving better CLAP scores. These experimental results demonstrate two key findings: a) RFM-based models show promising performance in generating customized and personalized content; b) DDPM-based approaches guide the models toward more robust performance by preserving the capability of generating features based on semantic information.

2) *Effectiveness of UNet Backbone*: We also compare our UNet architecture with diffusion-transformer (DiT) models, as DiT-based backbones have been shown to achieve superior performance in large-scale diffusion systems [92]. As reported in Table III, under a similar number of trainable parameters, *DreamAudio-DiT-B* does not achieve better performance than the UNet-based *DreamAudio*. By increasing the model size to approximately 1.3B, *DreamAudio-DiT-L* provides better results, however, it results in significantly longer training and inference time. These observations are consistent with previous findings that DiT architectures tend to outperform UNet models only at considerably larger scales (e.g., DiT-XL) [92]. Based on the current results, we adopt the UNet backbone in *DreamAudio* as a practical trade-off between performance and computational efficiency.

3) *Effectiveness of MRC*: We conducted various experiments with and without the external feature extraction path from the MRC module in Table III. For *DreamAudio-UNet*, the input of the generator module is a concatenation of general latent input and three VAE latent features of the reference audios. The results show that *DreamAudio* with only basic UNet structures exhibits significant performance

degradation in all metrics. In addition, the comparison between *DreamAudio-SEncoder* and *DreamAudio* indicates that the use of two groups of encoder blocks for encoding and feature extraction helps improve the overall performance.

4) *Effectiveness of Data Masking and Dropping*: We also conducted experiments on reference concept masking, dropping, and the impact of different data proportions. As shown in Table VI, the masking and dropping ratios significantly influence the model’s performance on customized tasks. When provided with more precise reference concepts, the model learns more effectively. However, based on the KL divergence and CLAP_A scores, we observe that the generated results contain more similar information to the target audio. This may be because the model is led to pay more attention to extraction and concatenation rather than generation, which may explain the reason of performance drop in general TTA generation tasks. This indicates that without masking and dropping, the model’s performance is degraded. Furthermore, in the data distribution comparison, we found that an excessive amount of customized-concatenation data makes the model overly reliant on the content of the reference concept. In contrast, Customized-Overlay data helps maintain a balance between leveraging the reference concept and the text prompts.

Training with AudioCaps data improves the robustness of the model in general TTA scenarios. However, an excessive amount of AudioCaps data causes the model to shift focus away from the features provided by the reference concept, leading to degraded performance on the CTTA task. Lastly, the experiment with 100% dropping represents the system trained without any reference text concepts while maintaining access to the reference audio clips, evaluating the ability of the model to perform customization based solely on the reference audio clips. The result shows a decrease in customization performance, which illustrates the importance of reference text in providing semantic grounding and helping the model exploit the reference audio features. In addition, as shown in Table IV, *DreamAudio* offers improved performance on general text-to-audio tasks on the AudioCaps benchmark, giving lower FAD and higher CLAP scores, as compared to AudioLDM and Stable Audio Open. This suggests that without the constraints of the reference text, *DreamAudio* tends to reduce its reliance

on reference concepts and learn to generate audio based primarily on the target prompt.

E. Limitation

Despite giving the satisfying performance, our method still has certain limitations.

1) *Fixed Reference Format*: The requirement for reference concept on both customized audio and related captions can be challenging to satisfy in real-world applications. Preparing suitable captions for referenced audio samples can lead to extra workloads and pose a practical barrier to general users.

2) *Fixed Audio Length*: Our current architecture is primarily developed for 10-second audio, aligned with those in training data. Although the model can be extended to generate longer clips (e.g., 30 seconds or more) by adjusting the latent feature dimensions, we observe a noticeable performance degradation when samples significantly exceed the 10-second training window. To improve the model’s generalization across varying audio lengths, incorporating more diverse datasets with variable durations for training and evaluation remains an important aspect of future work.

3) *The Number of References*: The current model architecture limits the number of reference concept pairs to three, restricting the system’s flexibility when handling extensive or diverse customization. Ablation studies show that *DreamAudio* can be easily fine-tuned to handle a varying number of reference concepts. However, the performance of the system can be affected by the availability of appropriate training data.

4) *Artificial Training Data*: Due to the lack of real-world data for customized tasks, the audio generated by the proposed model can sometimes sound unnatural. In addition, in many cases, the reference audio naturally corresponds to a subset of the target audio events, which can make the task appear closer to stitching or inpainting in terms of the sound generated. Addressing this limitation requires the development of more diverse and high-quality datasets tailored to specific tasks, allowing models to capture detailed patterns and improve their generalization to practical applications.

VII. CONCLUSION AND FUTURE WORKS

In this paper, we have introduced *DreamAudio*, a model for customised text-to-audio generation (CTTA). Given target prompts and reference audio-caption pairs as input, *DreamAudio* demonstrates satisfying performance in generating audio clips with specified audio content in the CTTA task. In addition, *DreamAudio* delivers competitive results in the general TTA task, establishing a robust foundation for a wide range of audio-related applications. To further validate the real-world applicability of our method, we created the first benchmark for CTTA, which includes zero-shot audio events to simulate real-world scenarios. Our evaluation demonstrates that *DreamAudio* captures these user-specific audio events, highlighting its ability to generate meaningful and semantically coherent content. In addition, we studied the adaptability of the proposed method, which can effectively handle a varying number of audio-caption pairs as reference concepts. Future work will focus on improving the flexibility of incorporating

reference audio-text pairs, changing the number of references, and developing diverse and high-quality real-world data. We will integrate additional modalities such as images and video, and extend its use in other customized audio tasks, such as audio separation, style transfer, and editing.

ACKNOWLEDGMENTS

This research was partly supported by a research scholarship from the China Scholarship Council (CSC), an internship from ByteDance, funding from British Broadcasting Corporation Research and Development (BBC R&D), Research England “Games and Innovation Nexus”, Engineering and Physical Sciences Research Council (EPSRC) Grant EP/T019751/1 “AI for Sound”, and a PhD scholarship from the Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey. For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) license to any Author Accepted Manuscript version arising. The authors wish to thank the associate editor and the reviewers for their helpful comments to further improve this work.

REFERENCES

- [1] Y. Cao, S. Li, Y. Liu, Z. Yan, Y. Dai, P. S. Yu, and L. Sun, “A comprehensive survey of AI-generated content: A history of generative AI from GAN to ChatGPT,” *arXiv:2303.04226*, 2023.
- [2] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, “AudioLDM: Text-to-Audio generation with latent diffusion models,” in *Proceedings of the International Conference on Machine Learning*, 2023, pp. 21 450–21 474.
- [3] K. Sung-Bin, A. Senocak, H. Ha, A. Owens, and T.-H. Oh, “Sound to visual scene generation by audio-to-visual latent alignment,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 6430–6440.
- [4] R. Sheffer and Y. Adi, “I hear your true colors: Image guided audio generation,” in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023.
- [5] X. Mei, V. Nagaraja, G. Le Lan, Z. Ni, E. Chang, Y. Shi, and V. Chandra, “FoleyGen: Visually-guided audio generation,” in *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing*, 2024.
- [6] V. Iashin and E. Rahtu, “Taming visually guided sound generation,” in *Proceedings of British Machine Vision Conference*, 2021.
- [7] F. Kreuk, G. Synnaeve, A. Polyak, U. Singer, A. Défossez, J. Copet, D. Parikh, Y. Taigman, and Y. Adi, “AudioGen: Textually guided audio generation,” *Proceedings of International Conference on Learning Representations*, 2022.
- [8] S. Forsgren and H. Martiros, “Riffusion: Stable diffusion for real-time music generation,” 2022.[Online]. Available: <https://riffusion.com/about>.
- [9] X. Liu, Z. Zhu, H. Liu, Y. Yuan, M. Cui, Q. Huang, J. Liang, Y. Cao, Q. Kong, M. D. Plumbley, and W. Wang, “WavJourney: Compositional audio creation with large language models,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 33, pp. 2830–2844, 2025.
- [10] J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” in *Proceedings of International Conference on Learning Representations*, 2020.
- [11] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 6840–6851.
- [12] R. Valle, R. Badlani, Z. Kong, S.-g. Lee, A. Goel, S. Kim, J. F. Santos, S. Dai, S. Gururani, A. Aljafari *et al.*, “Fugatto I: Foundational generative audio transformer opus 1,” in *Proceedings of the International Conference on Learning Representations*.
- [13] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “AudioSet: An ontology and human-labeled dataset for audio events,” in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 776–780.

- [14] X. Mei, C. Meng, H. Liu, Q. Kong, T. Ko, C. Zhao, M. D. Plumbley, Y. Zou, and W. Wang, "WavCaps: A ChatGPT-assisted weakly-labelled audio captioning dataset for audio-language multimodal research," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 3339–3354, 2024.
- [15] L. Sun, X. Xu, M. Wu, and W. Xie, "A large-scale dataset for audio-language representation learning," *arXiv:2309.11500*, 2023.
- [16] Y. Yuan, D. Jia, X. Zhuang, Y. Chen, Z. Liu, Z. Chen, Y. Wang, Y. Wang, X. Liu, X. Kang *et al.*, "Sound-VECaps: Improving audio generation with visual enhanced captions," in *Proceedings of the NeurIPS Workshop*, 2024.
- [17] F. Kreuk, G. Synnaeve, A. Polyak, U. Singer, A. Défossez, J. Copet, D. Parikh, Y. Taigman, and Y. Adi, "AudioGen: textually guided audio generation," in *Proceedings of International Conference on Learning Representations*, 2023.
- [18] D. Yang, J. Yu, H. Wang, W. Wang, C. Weng, Y. Zou, and D. Yu, "Diffsound: Discrete diffusion model for text-to-sound generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.
- [19] D. Ghosal, N. Majumder, A. Mehrish, and S. Poria, "Text-to-audio generation using instruction guided latent diffusion model," in *Proceedings of ACM International Conference on Multimedia*, 2023, pp. 3590–3598.
- [20] J. Huang, Y. Ren, R. Huang, D. Yang, Z. Ye, C. Zhang, J. Liu, X. Yin, Z. Ma, and Z. Zhao, "Make-An-Audio 2: Temporal-enhanced text-to-audio generation," *arXiv:2305.18474*, 2023.
- [21] Y. Yuan, H. Liu, X. Liu, Q. Huang, M. D. Plumbley, and W. Wang, "Retrieval-augmented text-to-audio generation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal*, 2024, pp. 581–585.
- [22] H. Liu, Y. Yuan, X. Liu, X. Mei, Q. Kong, Q. Tian, Y. Wang, W. Wang, Y. Wang, and M. D. Plumbley, "AudioLDM 2: Learning holistic audio generation with self-supervised pretraining," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.
- [23] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 22 500–22 510.
- [24] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," in *Proceedings of International Conference on Machine Learning*, 2021, pp. 8821–8831.
- [25] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with CLIP latents," *arXiv:2204.06125*, 2022.
- [26] J. Betker, G. Goh, L. Jing, T. Brooks, J. Wang, L. Li, L. Ouyang, J. Zhuang, J. Lee, Y. Guo *et al.*, "Improving image generation with better captions," *Computer Science*, vol. 2, no. 3, 2023.
- [27] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.
- [28] P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saimi, Y. Levi, D. Lorenz, A. Sauer, F. Boesel *et al.*, "Scaling rectified flow transformers for high-resolution image synthesis," in *Proceedings of International Conference on Machine Learning*, 2024.
- [29] K. Saito, D. Kim, T. Shibuya, C.-H. Lai, Z. Zhong, Y. Takida, and Y. Mitsufuji, "SoundCTM: Uniting score-based and consistency models for text-to-sound generation," *arXiv:2405.18503*, 2024.
- [30] M. Yang, B. Shi, M. Le, W.-N. Hsu, and A. Tjandra, "AudioBox TTA-RAG: Improving zero-shot and few-shot text-to-audio with retrieval-augmented generation," *arXiv:2411.05141*, 2024.
- [31] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *Proceedings of IEEE International Conference on Computer Vision*, 2023.
- [32] Y. Song, J. Sohl-Dickstein, D. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *Proceedings of International Conference on Learning Representations*, 2021.
- [33] P. Dhariwal and A. Nichol, "Diffusion models beat GANs on image synthesis," in *Advances in Neural Information Processing Systems*, 2021.
- [34] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi, "Photorealistic text-to-image diffusion models with deep language understanding," *arXiv:2205.11487*, 2022.
- [35] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4713–4726, 2022.
- [36] N. Chen, Y. Zhang, H. Zen, R. Weiss, M. Norouzi, and W. Chan, "WaveGrad: Estimating gradients for waveform generation," in *Proceedings of International Conference on Learning Representations*, 2021.
- [37] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "DiffWave: A versatile diffusion model for audio synthesis," in *Proceedings of International Conference on Learning Representations*, 2021.
- [38] H. Liu, R. Huang, Y. Liu, H. Cao, J. Wang, X. Cheng, S. Zheng, and Z. Zhao, "AudioLCM: Efficient and high-quality text-to-audio generation with minimal inference steps," in *Proceedings of the ACM International Conference on Multimedia*, 2024, pp. 7008–7017.
- [39] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni *et al.*, "Make-A-Video: Text-to-video generation without text-video data," in *Proceedings of International Conference on Learning Representations*, 2022.
- [40] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, and T. Salimans, "ImageGen Video: High definition video generation with diffusion models," *arXiv:2210.02303*, 2022.
- [41] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, "GradTTS: A diffusion probabilistic model for text-to-speech," in *Proceedings of International Conference on Machine Learning*, 2021, pp. 8599–8608.
- [42] Z. Chen, Y. Wu, Y. Leng, J. Chen, H. Liu, X. Tan, Y. Cui, K. Wang, L. He, S. Zhao, J. Bian, and D. Mandic, "ResGrad: Residual denoising diffusion probabilistic models for text to speech," *arXiv preprint:2212.14518*, 2022.
- [43] M. Lam, J. Wang, R. Huang, D. Su, and D. Yu, "Bilateral denoising diffusion models," in *Proceedings of International Conference on Learning Representations*, 2022.
- [44] S. Lee, H. Kim, C. Shin, X. Tan, C. Liu, Q. Meng, T. Qin, W. Chen, S. Yoon, and T. Liu, "PriorGrad: Improving conditional denoising diffusion models with data-driven adaptive prior," in *Proceedings of International Conference on Learning Representations*, 2022.
- [45] Z. Chen, X. Tan, K. Wang, S. Pan, D. Mandic, L. He, and S. Zhao, "InferGrad: Improving diffusion models for vocoder by considering inference in training," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022.
- [46] Q. Kong, Y. Xu, T. Iqbal, Y. Cao, W. Wang, and M. D. Plumbley, "Acoustic scene generation with conditional SampleRNN," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 925–929.
- [47] X. Liu, T. Iqbal, J. Zhao, Q. Huang, M. Plumbley, and W. Wang, "Conditional sound generation using neural discrete time-frequency representation learning," *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing*, 2021.
- [48] Y. Yuan, H. Liu, J. Liang, X. Liu, M. D. Plumbley, and W. Wang, "Leveraging pre-trained AudioLDM for sound generation: A benchmark study," in *Proceedings of European Association for Signal Processing*, 2023.
- [49] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: generative adversarial networks for efficient and high fidelity speech synthesis," in *Proceedings of the International Conference on Neural Information Processing Systems*, 2020, pp. 17 022–17 033.
- [50] K. Kumar, R. Kumar, T. De Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. De Brebisson, Y. Bengio, and A. C. Courville, "MelGAN: Generative adversarial networks for conditional waveform synthesis," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [51] S.-g. Lee, W. Ping, B. Ginsburg, B. Catanzaro, and S. Yoon, "BigVGAN: A universal neural vocoder with large-scale training," in *Proceedings of International Conference on Learning Representations*, 2022.
- [52] Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov, "Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023.
- [53] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.
- [54] C. D. Kim, B. Kim, H. Lee, and G. Kim, "AudioCaps: Generating captions for audios in the wild," in *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics*, 2019, pp. 119–132.
- [55] Y. Yuan, X. Liu, H. Liu, M. D. Plumbley, and W. Wang, "FlowSep: Language-queried sound separation with rectified flow matching," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2025.

- [56] H. Liu, J. Wang, R. Huang, Y. Liu, H. Lu, Z. Zhao, and W. Xue, "FlashAudio: Rectified flow for fast and high-fidelity text-to-audio generation," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2025, pp. 13 694–13 710.
- [57] W. Guan, K. Wang, W. Zhou, Y. Wang, F. Deng, H. Wang, L. Li, Q. Hong, and Y. Qin, "LAFMA: A latent flow matching model for text-to-audio generation," in *Proceedings of Interspeech 2024*, 2024.
- [58] C.-Y. Hung, N. Majumder, Z. Kong, A. Mehrish, A. A. Bagherzadeh, C. Li, R. Valle, B. Catanzaro, and S. Poria, "Tangoflux: Super fast and faithful text to audio generation with flow matching and clap-ranked preference optimization," *arXiv preprint arXiv:2412.21037*, 2024.
- [59] Z. Evans, J. D. Parker, C. Carr, Z. Zukowski, J. Taylor, and J. Pons, "Stable audio open," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2025, pp. 1–5.
- [60] D. Hendrycks, K. Lee, and M. Mazeika, "Using pre-training can improve model robustness and uncertainty," in *Proceedings of International Conference on Machine Learning*, vol. 97, 2019, pp. 2712–2721.
- [61] Y. Yuan, H. Liu, X. Kang, P. Wu, M. D. Plumbley, and W. Wang, "Text-driven Foley sound generation with latent diffusion model," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, 2023, pp. 231–235.
- [62] M. Cao, X. Wang, Z. Qi, Y. Shan, X. Qie, and Y. Zheng, "Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 22 560–22 570.
- [63] X. Chen, L. Huang, Y. Liu, Y. Shen, D. Zhao, and H. Zhao, "Anydoor: Zero-shot object-level image customization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 6593–6602.
- [64] R. Gal, M. Arar, Y. Atzmon, A. H. Bermano, G. Chechik, and D. Cohen-Or, "Encoder-based domain tuning for fast personalization of text-to-image models," *ACM Transactions on Graphics*, vol. 42, no. 4, 2023.
- [65] B. Kawar, S. Zada, O. Lang, O. Tov, H. Chang, T. Dekel, I. Mosseri, and M. Irani, "Imagic: Text-based real image editing with diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 6007–6017.
- [66] Y. Tewel, R. Gal, G. Chechik, and Y. Atzmon, "Key-locked rank one editing for text-to-image personalization," in *Proceedings of ACM SIGGRAPH*, 2023, pp. 1–11.
- [67] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or, "An image is worth one word: Personalizing text-to-image generation using textual inversion," *arXiv:2208.01618*, 2022.
- [68] A. Voynov, Q. Chu, D. Cohen-Or, and K. Aberman, "P+: Extended textual conditioning in text-to-image generation," *arXiv:2303.09522*, 2023.
- [69] Y. Alaluf, E. Richardson, G. Metzger, and D. Cohen-Or, "A neural space-time representation for text-to-image personalization," *ACM Transactions on Graphics*, vol. 42, no. 6, 2023.
- [70] Z. Liu, R. Feng, K. Zhu, Y. Zhang, K. Zheng, Y. Liu, D. Zhao, J. Zhou, and Y. Cao, "Cones: Concept neurons in diffusion models for customized generation," *arXiv:2303.05125*, 2023.
- [71] N. Ruiz, Y. Li, V. Jampani, W. Wei, T. Hou, Y. Pritch, N. Wadhwa, M. Rubinstein, and K. Aberman, "HyperDreamBooth: Hypernetworks for fast personalization of text-to-image models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 6527–6536.
- [72] Y. Wei, Y. Zhang, Z. Ji, J. Bai, L. Zhang, and W. Zuo, "Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 15 943–15 953.
- [73] G. Ding, C. Zhao, W. Wang, Z. Yang, Z. Liu, H. Chen, and C. Shen, "FreeCustom: Tuning-free customized image generation for multi-concept composition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 9089–9098.
- [74] J. Choi, Y. Choi, Y. Kim, J. Kim, and S. Yoon, "Custom-edit: Text-guided image editing with customized diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023.
- [75] H. Liu, R. Huang, X. Lin, W. Xu, M. Zheng, H. Chen, J. He, and Z. Zhao, "ViT-TTS: Visual text-to-speech with scalable diffusion transformer," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2023.
- [76] Y. Chen, Z. Niu, Z. Ma, K. Deng, C. Wang, J. JianZhao, K. Yu, and X. Chen, "F5-TTS: A fairytales that fakes fluent and faithful speech with flow matching," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2025, pp. 6255–6271.
- [77] M. Plitsis, T. Kouzelis, G. Paraskevopoulos, V. Katsouros, and Y. Panagakis, "Investigating personalization methods in text to music generation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2024, pp. 1081–1085.
- [78] Y. Jiang, Z. Chen, Z. Ju, C. Li, W. Dou, and J. Zhu, "FreeAudio: Training-free timing planning for controllable long-form text-to-audio generation," in *Proceedings of the ACM International Conference on Multimedia*, 2025, pp. 9871–9880.
- [79] Z. Evans, C. Carr, J. Taylor, S. H. Hawley, and J. Pons, "Fast timing-conditioned latent audio diffusion," in *Forty-first International Conference on Machine Learning*, 2024.
- [80] S. Sheynin, O. Ashual, A. Polyak, U. Singer, O. Gafni, E. Nachmani, and Y. Taigman, "KNN-Diffusion: Image generation via large-scale retrieval," in *Proceedings of International Conference on Learning Representations*, 2023.
- [81] W. Chen, H. Hu, C. Saharia, and W. W. Cohen, "Re-finalImagen: Retrieval-augmented text-to-image generator," in *Proceedings of International Conference on Learning Representations*, 2023.
- [82] X. Liu, C. Gong *et al.*, "Flow straight and fast: Learning to generate and transfer data with rectified flow," in *International Conference on Learning Representations*.
- [83] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*, 2021, pp. 8748–8763.
- [84] Y. Yuan, Z. Chen, X. Liu, H. Liu, X. Xu, D. Jia, Y. Chen, M. D. Plumbley, and W. Wang, "T-CLAP: Temporal-enhanced contrastive language-audio pretraining," in *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing*, 2024.
- [85] A. Vyas, B. Shi, M. Le, A. Tjandra, Y.-C. Wu, B. Guo, J. Zhang, X. Zhang, R. Adkins, W. Ngan *et al.*, "Audiobox: Unified audio generation with natural language prompts," *arXiv:2312.15821*, 2023.
- [86] F. Font, G. Roma, and X. Serra, "Freesound technical demo," in *Proceedings of the ACM International Conference on Multimedia*, 2013, pp. 411–412.
- [87] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 279–283, 2017.
- [88] K. J. Piczak, "ESC: dataset for environmental sound classification," in *Proceedings of the ACM International Conference on Multimedia*, 2015.
- [89] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image Recognition," *arXiv:1409.1556*, 2014.
- [90] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.
- [91] A. Tjandra, Y.-C. Wu, B. Guo, J. Hoffman, B. Ellis, A. Vyas, B. Shi, S. Chen, M. Le, N. Zacharov *et al.*, "Meta Audiobox Aesthetics: Unified automatic quality assessment for speech, music, and sound," *arXiv:2502.05139*, 2025.
- [92] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4195–4205.
- [93] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proceedings of International Conference on Learning Representations*, 2019.



Yi Yuan received the B.Eng. degree from the University of Sydney, Sydney, NSW, Australia, in 2021, and the M.S. degree in artificial intelligence from the University of Surrey, Guildford, U.K., in 2022. He is currently working toward the Ph.D. degree in vision, speech, and signal processing with the University of Surrey. His research focuses on deep-learning-based audio generation. In 2023, he achieved the top-1 ranking in DCASE Challenge Task 7. He has coauthored more than 20 papers published in leading journals and conferences, including *IEEE/ACM Transactions on Audio Speech and Language Processing*, *IEEE Journal of Selected Topics in Signal Processing*, *CVPR*, *ICML*, *ICASSP*, and *INTERSPEECH*.



Xubo Liu received Ph.D. degree with the Centre for Vision, Speech, and Signal Processing (CVSSP), University of Surrey, Guildford, U.K. in 2025, working on multimodal learning for audio and language, focusing on the understanding, separation, and generation of audio signals in tandem with natural language. He has coauthored more than 40 papers in top conferences such as CVPR, ICML, AAAI, EMNLP, ICASSP, and Interspeech. He organized the “Multimodal Learning for Audio and Language” special session at EUSIPCO 2023, and the “Language-Queried Audio Source Separation” challenge on DCASE 2024. He is also a frequent Reviewer for IEEE/ACM Transactions on Audio Speech and Language Processing, CVPR, EMNLP, ICASSP, Interspeech, and MLSP.



Haohe Liu received the B.Eng. degree from Northwestern Polytechnical University, Xi’an, China, in 2020. He received the Ph.D. degree with the Centre for Vision Speech and Signal Processing, University of Surrey, Guildford, U.K., in 2025. His past research has contributed to audio quality enhancement, audio generation, source separation, and audio recognition. His research interests include developing AudioLDM for text-to-audio generation, generating wide attention in the open-source community. His work as the primary author has been accepted in top journals and conferences such as IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE/ACM Transactions on Audio Speech and Language Processing, IEEE Journal of Selected Topics in Signal Processing, ICML, AAAI, ICASSP, and INTERSPEECH, and recent projects include AudioLDM, VoiceFixer, AudioSR, and NaturalSpeech.



Xiyuan Kang received the B.Eng. degree from Harbin Engineering University, Harbin, China, in 2021, and the M.S. degree in artificial intelligence from the University of Surrey, Guildford, U.K., in 2023. She is currently pursuing the Ph.D. degree in computer science with the University of Surrey. Her research interests include 3D human pose estimation and shape disentanglement. Her work has been published in leading international conferences, including CVPR. She serves as a reviewer for IEEE Transactions on Neural Networks and Learning Systems.



Mark D. Plumbley (S’88-M’90-SM’12-F’15) received the B.A.(Hons.) degree in electrical sciences and the Ph.D. degree in neural networks from University of Cambridge, Cambridge, U.K., in 1984 and 1991, respectively. He is a Professor of Signal Processing and Head of Department of Informatics department at King’s College London, UK. His current research concerns AI, machine learning and signal processing for analysis, recognition and generation of sound. He led the first international data challenge on Detection and Classification of Acoustic Scenes and Events (DCASE) and recently held an Engineering and Physical Sciences Research Council (EPSRC) Fellowship on “AI for Sound”. He currently co-leads the EPSRC-funded Noise Network Plus, and is part of the EPSRC AI Hub in Generative Models. He is a Member of the IEEE Signal Processing Society Technical Committee on Audio and Acoustic Signal Processing, and a Fellow of the IET and IEEE. This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) [grant numbers EP/T019751/1, EP/Y028805/1]. For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.



Wenwu Wang (M’02-SM’11-F’26) was born in Anhui, China. He received the B.Sc., M.E., and the Ph.D. degrees, all in the field of automation, from Harbin Engineering University, China, in 1997, 2000, and 2002, respectively. He then worked with King’s College London, Cardiff University, Tao Group Ltd. (now Antix Labs Ltd.), and Creative Labs, before joining University of Surrey, U.K., in May 2007, where he is currently a Professor in Signal Processing and Machine Learning, and an Associate Head in External Engagement, School of Computer Science and Electronic Engineering, University of Surrey, UK. He is also a Core AI Fellow at the Surrey Institute for People Centred Artificial Intelligence. His current research interests include signal processing, machine learning and perception, artificial intelligence, machine audition (listening), human-AI collaboration, and statistical anomaly detection. He has (co)-authored over 400 papers in these areas. His works have been recognized with various awards, including the Meta Distinguished Faculty Award (2026), Audio Engineering Society Best Technical Paper Award (2025), IEEE Signal Processing Society Young Author Best Paper Award (2022), DCASE Judge’s Award (2020, 2023, and 2024), DCASE Reproducible System Award (2019 and 2020), and LVA/ICA Best Student Paper Award (2018). He has been elected to IEEE Fellow for contributions to audio classification, generation and source separation, since 2026. He is a Senior Area Editor (2025-2027) for IEEE Open Journal of Signal Processing and an Associate Editor (2024-2028) for IEEE Transactions on Multimedia. He was a Senior Area Editor (2019-2023) and Associate Editor (2014-2018) for IEEE Transactions on Signal Processing, and an Associate Editor (2020-2025) for IEEE/ACM Transactions on Audio Speech and Language Processing. He was the elected Chair (2023-2024) of IEEE Signal Processing Society (SPS) Machine Learning for Signal Processing (MLSP) Technical Committee, and a Board Member (2023-2024) of IEEE SPS Technical Directions Board. He is currently the elected Chair (2025-2027) of the EURASIP Technical Area Committee on Acoustic Speech and Music Signal Processing, a Technical Directions Board Representative (2026-2028) and Executive Sub-Committee Member (2026) of the IEEE SPS Conference Board, and an elected Member (2021-2026) of the IEEE SPS Signal Processing Theory and Methods Technical Committee. He was on the organization committee of IEEE ICASSP 2019 and 2024, INTERSPEECH 2022, IEEE MLSP 2013 and 2024, and IEEE SSP 2009. He is a Technical Program Co-Chair of IEEE MLSP 2025. He has been a keynote or plenary speaker at about 30 international conferences and workshops.

Zhuo Chen biography not available at the time of publication.

Yuxuan Wang biography not available at the time of publication.