WEP
Warwick
Evans
Publishing

Transactions on Engineering and Technology Research
ISSN: 2960-1797, eISSN: 2960-2246 | Volume 1
AMMEE 2023

An Overview of Visual Sound Synthesis Generation Tasks Based on Deep Learning Networks

Hongyu Gao

Taiyuan University of technology, School of software, Taiyuan, 030000, China

sthx128@gmail.com

Abstract. Visual sound synthesis (which refers to the process of recreating, as realistically as possible, the sound produced by the movements and actions of objects within a video, given specific conditions such as video content and accompanying text) is an important part of the composition of high-quality films at present. Most traditional methods of sound synthesis are based on the artificial creation of simulated props for sound effects synthesis, which is achieved by using various existing props and constructed scenes. However, traditional methods cannot meet specific conditions for sound effect synthesis and require large amounts of participant, material resources and time. It can take nearly ten hours to simulate realistic sound effects in a minute-long video. In this paper, we systematically summarize and consolidate current advances in deep learning in the field of visual sound synthesis, based on existing related papers. We focus on the exploration and development history of deep learning models for the task of visual sound synthesis, and classify detailed research methods and related dataset information based on their development characteristics. By analyzing the technical differences among various model approaches, we can summarize potential research directions in the field, thereby further promoting the rapid development and practical implementation of deep learning models in the video domain.

Keywords: AI generated content; Video onomatopoeia synthesis; Automatic sound synthesis.

1. Introduction

The name "Foley" originates from Jack Foley, who, in the 1930s, pioneered the use of footsteps, prop sounds, and more to add environmental audio effects to films. He laid the foundation for the artificial synthesis of sound effects, and in honor of his remarkable contribution, the name "Foley" has since been used to represent the profession of sound artists who recreate sounds. Today, Foley artists are professionals in the field of film production and audio post-production, and their primary role is to create lifelike environmental sound effects for films, television shows, animations, and other visual productions.

Sound design, or Foley sound design, refers to the artificial creation or enhancement of sounds used to enhance the audio processing of films, music, and other content. It originated in the early days of film production when specific sound effects needed to be created for particular video segments. This process is known as Foley sound design and can be applied to various fields, including film, games, and more, to generate special effects, environmental sounds, and other audio elements. Foley sound artists are essential because some sounds in films and videos are often challenging to capture during filming or may have poor original recording quality. Therefore, these sounds need to be added and improved during post-production. In the early days, Foley sound artists used various props and equipment to create specific sound effects. For instance, they might use different types of shoes to simulate footsteps on various surfaces or create collision sounds by manipulating objects. This method primarily aimed at reproducing specific situations seen in early films and sufficed for their content. However, as films and video content became more sophisticated, the demand for authentic and detailed audio effects increased. Foley sound production also began to use more sophisticated recording equipment and finely crafted sets. In the late 20th century, the introduction of digital technology allowed Foley sound artists to have more precise control over and edit sound effects, enhancing the quality and adjustability of audio effects.

Artificial Intelligence-Generated Content (AIGC) is an emerging field in the realm of artificial intelligence that leverages AI technologies to produce tailored outputs to meet specific needs. As of the current state of development, AIGC has evolved into various categories of multimodal content generation, including but not limited to text synthesis mode, video synthesis mode, and audio synthesis mode. This paper primarily focuses on the advancements of AIGC in the domain of video sound synthesis, often referred to as Foley sound design tasks. In recent years, with the rapid growth of artificial intelligence, the quest for more efficient and convenient audio generation methods has become a primary focus for AI in this field. In contrast to the traditional approach of creating specific sound effects in dedicated Foley recording studios, AI trained on vast datasets of real-world sounds can analyze specific video segments, select suitable generative models, and produce a wide range of sound effects as needed, tailored to the requirements of the video.

Sound synthesis technology achieved through deep learning networks can be applied to various domains such as film, gaming, and more. This technology enables the automation of generating special effects, ambient sounds, and other audio elements. One crucial application is the conditional generation of specified sound effects, achieving the automation of sound design for videos. In recent years, a variety of well-known core neural network models have been applied to Foley sound design tasks, including Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), Generative Adversarial Networks (GAN), and the Diffusion model. Existing deep learning models typically perform tasks related to generation, sequence modeling, classification, and regression. In these tasks, researchers typically input video or image data and use recurrent networks and generative adversarial networks to produce the required video or image content. In some classic generative models, such as CNN, convolutional operations are often employed to extract feature maps from specific image layers. These feature maps capture the visual characteristics of frames in the video, which are then used as input for RNN to model the temporal information within the video frames. GANs are used for sound effect generation by training on an adversarial learning process within the latent space, resulting in the creation of Foley sound effects that meet specific requirements. The Diffusion model, as an emerging research tool, has also been introduced to Foley research. Its core concept involves using diffusion processes to generate multimodal data, such as audio and video. During training, the model takes a pair of audio and video as input and learns to establish the relationship between these two modalities. During the generation process, it can produce one modality from another or jointly generate video, aligning the video with the audio.

In this paper, the main contributions are as follows:

(1) We systematically reviewed and analyzed nearly twenty papers published from 2016 to 2023 related to Foley sound synthesis under video conditions. We structured the development landscape, summarized core methods, and conducted comparisons among them.

(2) We curated recent publicly available datasets related to Foley sound synthesis, reorganized and categorized the data for ease of training, performed statistical analyses of each dataset, and provided high-speed download links.

(3) By consolidating recent models and datasets relevant to Foley sound synthesis, we summarized and synthesized the current state of the field. We offered new research directions and explored potential areas for future investigation in this task.

2. Applications of Foley research

In this major section, the first part, 2.1, provides an overview and introduction to relevant articles on Foley. The second part, 2.2, introduces the basic methods currently used in Foley tasks, and the algorithmic principles behind these methods are briefly explained in section 2.3. Finally, section 2.4 analyzes various derivative models under Foley tasks, while section 2.5 discusses the strengths and weaknesses of each derivative model.

2.1. Literature Review

From 2016 to 2023, a significant number of scholars have conducted in-depth research on the Foley video sound synthesis task (referred to as "Foley" below). The following papers reveal the developments in Foley research over the past seven years. By reviewing and understanding the research directions within these papers, we can gain a better understanding of the trajectory of Foley research.

In 2016, the groundbreaking work by [1], initially presented at CVPR, outlined the specific direction of Foley research, marking the inception of research in the field of automated Foley sound synthesis. The authors proposed that different materials convey distinct sound characteristics, and by modeling and learning these sound features, it becomes possible to generate video sound effects that match the materials. In this paper, the authors introduced the GHD (Greatest Hits dataset), which included a total of 977 videos, comprising 64% indoor scenes and 36% outdoor scenes. The materials used encompassed soft materials found in natural environments, such as grass and leaves, as well as various hard and soft materials found in indoor settings, including metal, plastic, fabric, and plastic bags. This dataset provided source material for training subsequent models and also enhanced the generative performance of the authors' CNN model.

In the same year, a publication in ECCV [2], building upon the previous research, suggested that sound could serve as annotated self-supervision information for training neural networks. The authors argued that sound, as a perceptual signal closely related to visual input, often provides valuable supplementary information, given the strong link between sound and visual effects. They used CNN for image feature extraction and employed RNN to indirectly learn contextual information from video frames. This approach allowed them to capture sound latent space features, perform clustering for categorization, and achieve self-supervised learning, effectively associating visual and audio information.

In 2017, a paper titled [3] introduced the task of cross-modal content generation. In this work, the authors proposed the use of GAN models and constructed two similar reversible networks, S2I (Sound-to-Image) and I2S (Image-to-Sound). Both networks included encoders, generators, and discriminators. By extracting features from each modality using CNNs and employing iterative adversarial training, they aimed to achieve cross-modal content generation. However, this approach was limited to specific scenarios, such as musical instrument performances, and did not support more extensive generative capabilities. The same author published another paper in 2018, [4], with the goal of providing a unified framework for cross-modal generation across multiple domains. Based on this, the authors introduced CMCGAN for cyclical generation. However, a challenge with this approach was the complexity of preprocessing, which required the use of an Auto Encoder for GAN training, adding to the workload.

In 2018, a paper published at the CVPR conference [5] delved into the study of generating audio from a visual perspective. Unlike previous research, this paper focused on the direction of generating audio from images. It employed GAN as the overall framework and introduced Sample RNN as the audio generation framework. Trained on the GHD dataset from [1], the authors achieved a 70% realism rate in audio generation. Subsequently, the same authors presented a paper at the MULA conference [6], where they introduced the POCAN model to address the issue of varying sound quality resulting from different types of sound inputs. This can be seen as another model research endeavor in sound generation and provided a unified approach to handling different types of sound inputs.

In 2019, a paper published at the ICCV conference [7] introduced a novel research direction, namely audio inpainting. In this paper, the VIDAI framework was proposed. It incorporated video information and utilized frame sequence content extracted from videos for contextual reasoning. This approach was used to fill in missing audio segments and generate complete audio by leveraging video information. This paper did not focus solely on soundless videos but introduced an entirely new framework and research direction for subsequent studies.

In 2020, a paper published in IEEE [11] focused on the research direction of generating audio from soundless videos while ensuring alignment with visual signals in both time and content. [11] introduced the REGNET architecture, which, through the proposed audio-forward regularizer, enabled the specific extraction and differentiation of information from the video content. This approach allowed for the separate processing of audio information to achieve alignment between audio and visual signals.

In the same year, a paper in IEEE [12] explored the relationship between audio and visual information, emphasizing the importance of focusing on the content of video frames. The authors achieved temporal synchronization between the generated sound and the video using interpolation techniques and Temporal Relation Networks (TRN) and aimed to provide more realistic sound effects.

In a paper published in SoSE in the same year [14], the authors realized that the applicability of handling only noise-free training data was too limited. They introduced Big GAN and utilized a large amount of unlabeled network information for input training. Additionally, they proposed research on combining Internet of Things (IoT) with cloud computing as an emerging concept. Following a similar research approach in 2021, the same authors combined TRN and GAN to redesign a model and published it in [17]. Their aim was also to address the consistency between video and audio content in Foley content generation.

The latest paper published in 2023, [18], introduces a conditional Foley generation task. It utilizes Transformer and a VQGAN-based generative model to predict a code that represents the input sound example in an autoregressive manner, guided by specific generation requirements. Subsequently, this code is transcribed into a waveform using a Mel GAN vocoder.

Another paper published in CVPR, [19], argues that most previous research only achieved single-modal generation at a time (e.g., CMCGAN in [4]). In contrast, the work presented in this paper is capable of simultaneously generating two modalities. Specifically, the MM-Diffusion framework aims to recover two consistent modalities during a diffusion process, allowing it to handle different modality data simultaneously. The challenge in this work lies in the fact that audio and video are two distinct modalities with different data representations: video is represented as a 3D signal, while audio is a 1D waveform. Furthermore, it requires audio and video to be temporally synchronized in real-world scenarios.To address these challenges, the authors propose the use of a coupled U-Net architecture for joint modeling of audio and video data. This architecture can simultaneously process audio and video data in each denoising diffusion step, enabling the joint generation of multi-modal data.

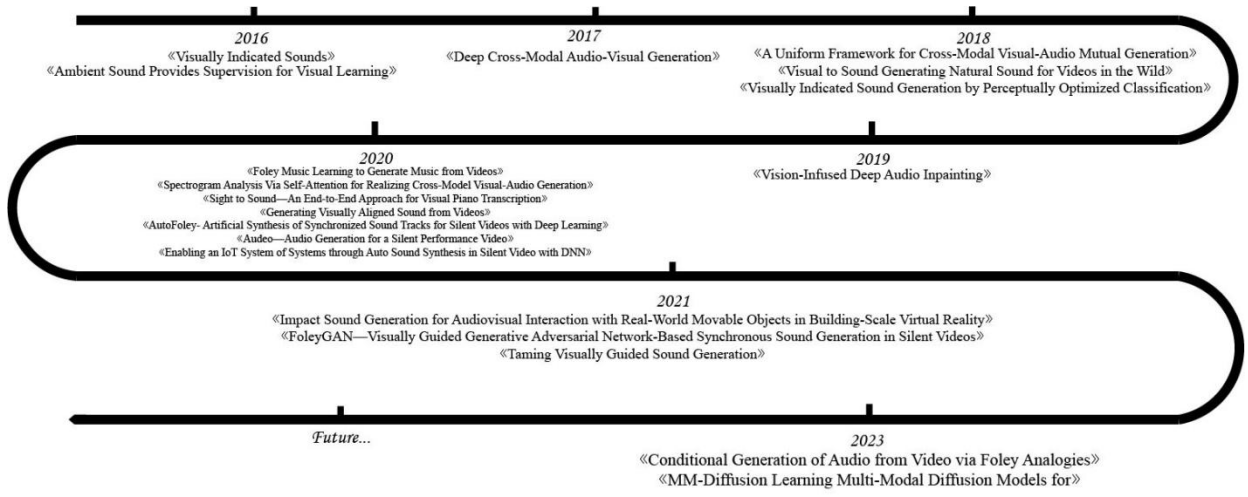

Figure 1: Overview of Foley-Related Papers from 2016 to Present

The figure provides a chronological overview of the 19 cited papers in this article. Research in the field of Foley has been steadily progressing since 2016. The year 2020 marked a peak in the number of related studies, and the subsequent two years also saw the introduction of new research topics.

Table 1: Summary of Model Content

Paper	Model Name	Year	Model Architecture	Advantage	Disadvantage
[1]	Visually Indicated Sounds	2016	CNN+RNN	Accomplished video-to-sound generation.	Capable of unidirectional generation, specifically generating audio from video.
[2]	Ambient Sound Provides Supervision for Visual Learning	2016	CNN+RNN	Utilizing audio to enrich the semantic representation of vision, this marks the first self-supervised approach for audio-visual tasks.	The model can only generate Mel-spectrograms.
[3]	Deep Cross-Modal Audio-Visual Generation	2017	GAN	Achieving bidirectional generation of audio and images in a specific domain using GAN.	The model can only generate Mel - spectrograms.
[4]	CMCGAN	2018	GAN	Implemented a cyclic generation method to enable the cross-utilization of input data.	It needs to go through pre-trained Auto Encoder training before GAN training.
[5]	Visual to Sound Generating Natural Sound for Videos in the Wild	2018	Sample RNN +GAN	In the direction of visual-to-audio modality conversion and generating sound from visual inputs, there is also a proposed direction to transcribe sound from video content.	Without focusing on audio generation quality.
[6]	POCAN	2018	CNN+LTSM	To be able to generate using the same framework for different types of sounds.	This model requires a significant amount of pre-training.
[7]	VIAI	2019	CNN+RNN+GAN	Repairing videos with audio using contextual inference from visual information and partially missing audio.	Supports audio inpainting for partial audio recovery in video clips.
[8]	Graph−Transformer	2020	RNN+GAN	Associating video and audio signals using	Inaccurate recognition of specific human

				MIDI and capturing human key points.	movements, requiring end-to-end training.
[9]	SA-CMGAN	2020	CNN + GAN	Achieving cross-modal visual-audio generation.	The model can only generate Mel-spectrograms.
[10]	An End-to-End Approach for Visual Piano Transcription	2020	CNN+RNN	Achieving end-to-end transcription of piano audio.	This framework is only suitable for transcribing piano audio.
[11]	Reg Net	2020	GAN	The introduction of an audio regularizer aligns the audio and video content, and the application of GAN to spectrograms improves the quality of the generated audio.	Training is required for each individual category.
[12]	FS-LTSM	2020	CNN+LTSM	Distinguishing reading using the characteristics of different levels of processing.	The authors believe that it is necessary to consider the alignment between specific frames sequences in videos and audio.
[14]	Res Net-FSLSTM	2020	FSLTSM + GAN	The introduction of Big GAN and the analysis and summary of large amounts of data in the network using unlabeled information.	Overlapping research ideas with previous work.
[17]	Foley GAN	2021	GAN	All previous work was based on direct visual-sound synthesis, while Foley requires a noise-free relevant dataset.	Only the integration of TRN + GAN was carried out, and the innovation is relatively low.
[18]	Condition Foley	2023	GAN	By designing a self-supervised pre-training model, it is possible to learn the relevant information from conditional audio to visual edits.	The model can only generate Mel-spectrograms.
[19]	MM-Diffusion	2023	GAN U-net	The utilization of two coupled autoencoders within the same model allows for the association of information between video and audio, which are two distinct modalities.	The diffusion model-based approach requires extensive pre-training and data computation.

2.2. Basic methods and principles mainly involved by Foley

Currently, Foley sound synthesis has matured with various models and approaches. These approaches are based on several neural network algorithms, such as CNN, RNN, LSTM, GAN, and diffusion

models are also introduced in the field of deep learning as auxiliary tools to enhance generation performance. Additionally, these models are tailored to specific Foley application scenarios, with each paper offering model improvements specific to its particular task.

Fully connected network, also known as dense layer or fully connected layer, plays a key role in transforming the feature maps extracted by convolutional or pooling layers into the final output and performing tasks like classification, regression, or other predictions.

In the fully connected layer, each "neuron" is connected to all the "neurons" from the previous layer. Each connection is assigned a weight to adjust the importance of input features.

The calculation process in the fully connected layer can be represented as follows:：

$$y = \sigma(W * x + b) \tag{1}$$

In the equation, where $\boldsymbol{W}$ is the weight matrix, $\boldsymbol{\sigma}$ is the activation function, $\boldsymbol{x}$ is the input vector, $\boldsymbol{b}$ is the bias vector, and * represents matrix multiplication.

Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN): CNN and RNN, as fundamental deep learning models, are widely used in Foley production. For instance, Andrew Owens in [17] employed a trained CNN to predict the sounds that occurred during video recording. Understanding these basic neural network models provides a profound insight into the related algorithmic models.

CNN's core structure consists of convolutional layers and pooling layers. Convolutional layers primarily focus on feature extraction, while pooling layers aim to reduce dimensionality and feature selection. The fundamental task of convolutional layers is to extract image features, as defined in the basic formula below:

$$(f * g)(n) = \sum_{m=-\infty}^{\infty} f(m)\, g(n-m) \tag{2}$$

In this equation，$\boldsymbol{f}$ and $\boldsymbol{g}$ represent two functions，$\boldsymbol{W} * \boldsymbol{H} * \boldsymbol{C}$ stands for the convolution operation，$\boldsymbol{f}$ typically represents the input image，$\boldsymbol{g}$ is the convolution kernel, The convolutional kernel can be thought of as a set of learnable filters. Assuming the size of an input image is $\boldsymbol{W}$,in this equation W represents width, $\boldsymbol{H}$ represents height，$\boldsymbol{C}$ represents the number of channels，and the size of the convolutional kernel is $\boldsymbol{H} * \boldsymbol{C}$, K represents the width and height of the kernel(similarly for the following equation),the convolution operation can be represented as:

$$y_{ij} = \sigma(\sum_{u=0}^{K-1} \sum_{V=0}^{K-1} \sum_{C=0}^{C-1} w_{u,v,c} x_i + u, j + v, c + b) \tag{3}$$

$\boldsymbol{y_{ij}}$ represents the output of the convolutional layer, $\boldsymbol{\sigma}$ denotes the activation function, $\boldsymbol{b}$ represents the bias term.

The main purpose of pooling layers is to reduce the dimensions of data, thereby lowering computational complexity and the number of parameters. Additionally, pooling layers can perform a degree of feature selection. There are two common pooling operations: Max Pooling and Average Pooling.

The mathematical definition of Max Pooling is as follows:

$$y_{i,j} = \max_{u=0} K - 1 * \max_{v=0} K - 1 * x_{i+u,j+u} \tag{4}$$

The mathematical definition of Average Pooling is as follows:

$$y_{ij} = \frac{1}{K^2} \sum_{u=0}^{K-1} \sum_{v=0}^{K-1} x_{i+u,j+u} \tag{5}$$

Where $\boldsymbol{u}$、$\boldsymbol{v}$ represents the starting step point, $\boldsymbol{j}$ indicates the step being taken.

CNN learns the correlation between video content and its corresponding sound, extracting semantic information about sound from video data, thus establishing a connection between visual representation and sound. LTSM, an improved version of RNN designed for handling sequential data, incorporates algorithms to mitigate gradient vanishing issues, effectively addressing problems of gradient vanishing and exploding. In [11], interpolation techniques are employed to map auditory features to a multiscale deep convolutional recurrent neural network (Res Net FSLSTM) associated with intermediate video data, in conjunction with the CNN and FS-LSTM architectures for sound category prediction.

In the work by Andrew Owens, as presented in [1], CNN is used to extract features from input image data, subsequently used as input for RNN to predict sound feature sequences corresponding to video segments. By utilizing CNN, the model can automatically learn visual features from images, which convey information about the content and structure of images, thus aiding in the understanding of the relationship between images and sound. Another pivotal model, the Recurrent Neural Network (RNN), possesses memory capabilities, enabling the processing of sequences of varying lengths and establishing temporal dependencies. The fundamental principles can be summarized as follows:

(1) Recurrent Structure: The recurrent structure is a core feature of recurrent neural networks, incorporating the previous time step's hidden state as input for the current time step, forming a self-feedback loop.

(2) Time Steps: For a sequence data of length t, the RNN processes input data step by step in accordance with time steps, updating the hidden state and output at each time step. Each computational step at every time step of the RNN can be described using the following formula.：

$$h_t = f(W_{hh} * h_{t-1} + W_{xh} * x_t + b_h) \tag{6}$$

$$y_t = g(W_{hy} * h_t + b_y) \tag{7}$$

h represents the layer height, y represents the layer width, and t represents the time step

(3) Hidden State Update: At each time step, RNN updates the current time step's hidden state based on the current input and the hidden state from the previous time step.

(4) Output Computation: Based on the current time step's hidden state, RNN calculates the corresponding output. The output can serve as a prediction or be further involved in subsequent computations.

Recurrent Neural Networks (RNN) also have some issues, such as the problem of gradient explosion when dealing with long sequential data. Additionally, the standard RNN structure struggles with handling very long sequences or large time intervals. Due to the recurrent nature of RNN, computations at each time step depend on the output of the previous time step, making it challenging to perform efficient parallel computations.

To address these problems, researchers introduced Long Short-Term Memory (LSTM) as an improvement over RNN, specifically designed for handling sequential data and solving the issues of gradient vanishing and exploding. In standard RNN, gradients can multiply across time steps, leading to gradient vanishing or exploding. LSTM, on the other hand, uses gate units to control the flow of information, weighting the importance of past temporal information effectively.

The core idea of LSTM is to introduce three gate units: the input gate, the forget gate, and the output gate. These gate units, controlled by learnable weights, determine whether to pass, forget, or output information. This approach results in stable training and inference results.

LSTM introduces a cell state to store and pass information, similar to the hidden state in traditional RNN. However, LSTM uses gate units to decide when to update and pass information to avoid excessive accumulation or decay of information. The hidden state is the primary output of LSTM, and its output information can be passed to the next time step or used for specific tasks.

Generative Adversarial Networks (GAN): Another important model is GAN, which, due to its generative adversarial capability, can produce more realistic audio effects. In [3], conditional GANs were used, connecting the generator network with a discriminator algorithm. Additionally, by utilizing CNN as an information extraction tool, the GAN's ability to generate specific content was enhanced. The basic process of GANs typically involves encoding, extracting image features, generating corresponding audio content, discriminator judgment, and achieving content generation.

Furthermore, in [4], a GAN model named CMCGAN was proposed, allowing cyclic processing of input content. It combines three to four generators with discriminators to ensure that the input content is judged by multiple discriminators, enhancing audio quality. It also achieves an end-to-end output with a Video-to-Audio (V-A) model. In [9], an SA-CMGAN model was introduced, which uses two networks to mutually generate images and audio. Each network includes an encoder, generator, and discriminator. The encoder consists of five consecutive convolutional layers, with a self-attention layer following the first convolutional layer, and three fully connected layers after the last convolutional layer.

GAN can also serve as the foundational model for building an overall framework. For instance, in [6], the author used GAN as the basic framework, and in [7], GAN was used for specific segment audio synthesis. GAN is often used in conjunction with other models. In [5], Sample RNN was selected as the sound generator, and the GAN model framework was used to generate the corresponding audio raw waveform samples based on a given video frame. GANs are employed for the task of generating the original waveform corresponding to a specific video frame.

GANs consist of a generator and a discriminator, and they optimize the model through an adversarial training process to generate realistic sample data. The core idea of GAN is to improve the model by creating competition between the generator and the discriminator. The generator's goal is to generate data distribution similar to real samples, while the discriminator's goal is to distinguish whether the input sample is real or generated. Through continuous competition and iteration in the training process, the quality of generated samples and the ability to discriminate samples gradually improve.

The training process of GANs typically includes the following steps:

First, the generator updates its state to achieve the output.

$$\theta_G \leftarrow \theta_G - \lambda \nabla \theta_G L_G \tag{8}$$

$\boldsymbol{\theta_G}$represents the parameters of the generator, $\boldsymbol{\lambda}$ represents the learning rate, and$\boldsymbol{\nabla \cdot \theta_G L_G}$represents the gradient of the generator loss function with respect to its parameters.

Transmitting this content to the discriminator, comparing whether it can pass the inspection for the discriminator, updating its discrimination state based on the generator's input, and calculating the corresponding discrimination loss in this way.

$$\theta_G \leftarrow \theta_G - \lambda \nabla \theta_D L_D \tag{9}$$

Where $\boldsymbol{\theta_d}$ represents the parameters of the discriminator, $\boldsymbol{\lambda}$ represents the learning rate, and $\boldsymbol{\nabla \cdot \theta_D L_D}$ represents the gradient of the discriminator loss function with respect to its parameters.

During training, the loss functions for both the generator and the discriminator are minimized. The generator aims to produce realistic samples to deceive the discriminator, while the discriminator aims to accurately differentiate between real and generated samples. By iteratively updating the parameters of the generator and the discriminator, GAN gradually improves the quality of the generated samples and the discriminative ability of the discriminator.

Attention Mechanism:

The attention mechanism can be categorized into self-attention mechanisms and multi-head attention mechanisms. In self-attention, the attention weights for each position are calculated by measuring the relevance between queries (Q), keys (K), and values (V) with a scaling factor ($\boldsymbol{d_k}$). The resulting

attention weights are used to weight and sum the values, yielding the self-attention representation. The basic formula for the attention mechanism is as described above. Attention mechanisms assist the decoder in selectively focusing on relevant parts of the input sequence during generation, improving both the quality of generation and contextual coherence.

$$Attention_i(Q, K, V) = softmax(\frac{Q_i K_i^T}{\sqrt{d_k}})V_i \tag{10}$$

Transformer

Compared to RNN, which generates the next hidden information based on the previous node recursively, the Transformer can simultaneously process information from multiple hidden nodes in parallel. Leveraging the attention mechanism, it can handle context-related sequences efficiently, irrespective of the order. Furthermore, Transformers have found applications in image processing. In [8], the author uses a Transformer decoder to map visual representations to the MIDI modality, incorporating a Graph-Transformer module to match hand movements and generate Mel images for audio synthesis.

The core idea of the Transformer is to model dependencies between elements in a sequence using self-attention mechanisms, without being constrained by order. The Transformer model consists of an encoder and a decoder. The encoder typically comprises multiple identical layers, each containing two sub-layers: a multi-head self-attention layer and a feed-forward neural network layer. The encoder processes the input sequence and represents it as context-aware features. The decoder, on the other hand, generates the target sequence using the encoder's output and its own input. While generating each target element, the decoder dynamically focuses on different parts of the input sequence using the attention mechanism.

Diffusion Models

Diffusion Models are used to generate probability distributions, often employed for generating high-quality data samples. The core concept of diffusion models is to start from a simple probability distribution, such as a Gaussian or uniform distribution, and iteratively approach the target distribution through the generation of samples. This process is often described as "diffusing" or "evolving" towards the target data distribution.

In the field of deep learning, diffusion models can be understood as a variation of a Markov chain process. The diffusion process resembles state transitions in a Markov chain but occurs in a latent space, involving both forward and reverse steps. By modeling and training these two processes, diffusion models can learn the underlying distribution of the data. For instance, in [19], the author proposes using an MM-Diffusion model to interdependently generate two coupled encoders, addressing the generation of different modalities with distinct data patterns in multi-modal diffusion models.

The mathematical principles of diffusion models involve transformations of a series of probability distributions and probability density functions, accomplished through the processes of adding noise and denoising in both forward and reverse directions.

In the forward process:

$$x_t = \frac{1}{\sqrt{\beta_t}}(x_{t-1} - \beta_t f(x_{t-1}) + \sqrt{2\beta_t}\epsilon_t) \tag{11}$$

In the above formula, where $\boldsymbol{t}$ is the time step,$\boldsymbol{x_t}$ is the observed sampling point,$\boldsymbol{x_{t-1}}$ is the observed sampling point from the previous time step, $\boldsymbol{\beta_t}$ is the diffusion coefficient (gradually decreasing during training), $\boldsymbol{f(x_{t-1})}$ is a learnable function composed of neural networks,$\epsilon_t$is random noise following a standard normal distribution.

In the reverse process:

$$x_{t-1} = \frac{1}{\sqrt{\beta_t}}(x_t - \beta_t f(x_t) + \sqrt{2\beta_t}\epsilon_t) \quad (12)$$

(Here, the symbols are similar to the forward process but performed in reverse.)

Likelihood estimation and model training: Likelihood estimation for the diffusion model is typically based on the probability density functions of the forward and reverse processes, representing the likelihood of observed data points given a latent space representation.

This can be expressed as:

$$p(x_t|x_{t-1}) \quad (13)$$

Likelihood estimation is a critical part of model training and is often performed through Maximum Likelihood Estimation (MLE). Training the diffusion model typically involves estimating the situation of observed data points and setting parameters for the diffusion coefficient $\boldsymbol{\beta_t}$ and the training function $\boldsymbol{f(x)}$

2.3. Derived Models of Foley

The various deep learning models mentioned above have their advantages and drawbacks. In practical applications, they are often improved or combined according to the specific requirements. More commonly, the current mainstream research direction involves building a unified framework tailored to individual needs. Many of the models introduced recently are composed of combinations of basic models, forming a standardized framework. Analyzing and studying these frameworks in more detail can provide a better understanding of the actual models and methods used in Foley production.

In [17], the CMCGAN framework was proposed, which unifies four cross-modal generation methods for videos and sounds (bidirectional generation of sound and images). The four models share the same matrix weights, and input content from both sound sources and image sources is processed through cross-computation. After passing through two U-net-like models, the original sound is generated. By using this cyclic GAN model, end-to-end output is achieved, which was not possible in the previous cross-modal generation methods. This approach provides a solution for achieving cross-modal generation. The key to the CMCGAN framework is the introduction of latent vectors, allowing it to effectively handle the dimensional and structural asymmetry between the two different modalities. CMCGAN can unify visual-audio generation into a common framework and introduce more sound information for cross-modal generation. Such mutually generative frameworks enable information exchange and complementarity between visual and audio, contributing to an enhanced sense of realism and completeness in audiovisual experiences.

In subsequent research, in [17], a model called POCAN was introduced by combining CNN and LSTM. In this model, a frame from the video is processed through CNN to extract basic image features, which are then input into an LSTM for category classification. Subsequently, POCAN predicts the sound category and regresses the LSTM's hidden state to a spectrogram, which is then transformed into a waveform representation. POCAN, by considering different sound categories, can finely generate sound that matches the video content. Traditional methods often classify visual-indicated sound as a single category variant and do not establish a close connection between the two modalities. POCAN captures semantic information in the audio modality through a pretrained sound classification network, thereby establishing a prior relationship for audio generation. The authors use POCAN to synthesize realistic sounds consistent with the visual content and achieve a consistent correspondence between sound and images.

In [7], a model called VIAI was introduced, consisting of two parts: a pure audio module, VIAI-A, and an audio-visual joint restoration module, VIAI-AV. These two components handle content output separately. The VIAI-A model is responsible for processing the input corrupted audio to form a Mel spectrogram with missing parts. It is then processed by an encoder-decoder to generate the complete

spectrogram, which is input to a pretrained Wave Net decoder. The same audio segment is also input to VIAI-AV to extract information features. The missing audio features from the decoder-encoder are connected to reconstruct the spectrogram, and the results are input into a GAN model for loss calculation. The computed loss is then fed into the pretrained Wave Net decoder to generate the original audio.

This model combines the results of both methods and inputs them into Wave Net for audio synthesis. This joint reconstruction method ensures that the generated audio segments are not only acoustically plausible but also visually consistent with the video content. VIAI leverages both previous data and reconstructed spectrogram information to more accurately generate audio waveforms. By effectively utilizing information from both audio and video and introducing new model architectures and training strategies, VIAI accomplishes audio-visual segment restoration.

Of course, contemporary research also includes architectural approaches based on non-traditional models. In [11], the REGNET architecture is proposed, which comprises a visual encoder, an audio-forward regularizer, a generator, and a discriminator. The visual encoder captures image and motion features that match the audio generation and uses kernels pretrained on ImageNet, specifically BN-Inception.

The encoder includes three 1D convolutional layers, each followed by a rectified linear unit (ReLU) activation function. The audio-forward regularizer provides additional visual supervision during the training process for sound prediction. The input data passes through two bidirectional LSTM layers. The generator outputs Mel spectrograms from the concatenated visual features and the regularizer's output.

The discriminator is used for adversarial training. It takes extracted frame features and a spectrogram as input to distinguish whether the spectrogram comes from a real video or is generated by the proposed REGNET.

The key innovation in REGNET is the introduction of the audio-forward regularizer, which allows real-world sound effects to be integrated into the synthesis of Foley sound effects. This enables the generator to utilize both visual features and authentic sound effects to predict video sound. Consequently, this strengthens the connection between real sound effects and synthesized sound effects, enhancing realism. In traditional modal conversion, the generator may not necessarily produce corresponding real sound effect results. However, REGNET allows for the inference of the required audio alignment from the video content, eliminating the need to infer unrelated sounds. By controlling the capacity of visual information and audio content, and improving the received information, REGNET achieves alignment between video information and audio through this corresponding correction approach.

In [12], the authors propose a scheme for Auto Foley generation, which involves three main steps: (1) Sound feature extraction, (2) Predicting sound categories from certain video frames, and (3) Sound synthesis.

In the sound feature extraction part, the authors present two different models. One model uses interpolation techniques to map audio-visual features onto a Res Net-FSLSTM associated with intermediate video frames. The other model employs the Temporal Relation Network (TRN) to learn the relationship between a few frames from different time intervals in the video, thereby identifying expected actions. The first model aims to learn the audio-visual relationship in fast-moving movie segments, while the second model focuses on action recognition using selected frames from certain videos. These two models serve different purposes, and their main generative contents are combined to complement each other for the final output.

When predicting sound categories from the video, the authors apply a similar approach. They use both a Fast/Slow LSTM (F/S-LSTM) network and a video frame relation network (which combines CNN and TRN) for computation. In the frame sequence network, interpolation techniques are employed to capture detailed motion information from video frames. The Fast LSTM model within

the FS-LTSM network performs fast computations and integrates contextual content. Additionally, the authors use CNN to generate feature vectors from images, followed by labeling and connection. Slow LSTM layers help integrate information over larger time spans to capture essential features.

2.4. Briefly describe the strengths and weaknesses of the model

Research outcomes and final effects in various papers differ based on specific research needs. Some models successfully achieve the generation of complete audio, while others propose basic frameworks without providing a full audio generation process. Some models generate Mel spectrograms but require further conversion to audio.

First, in [1], Andrew Owens achieved the generation of video sound effects using the GHD dataset. This was the first comprehensive Foley research paper. The author used a CNN pre-trained on ImageNet and combined it with the most feature-rich audio from GHD to successfully generate audio features for different material surfaces, achieving one-way video-to-audio content generation. Building upon the research in [1], Andrew Owens attempted to enrich visual semantic representations with audio in [2], proposing a self-supervised audiovisual solution. This approach incorporates audio into video information to achieve more comprehensive Foley audio effects but only generates Mel spectrograms, requiring further conversion to audio.

Similar to this, [3] also only generates Mel spectrograms. Lele Chen believed that visual-to-sound (V2S) and sound-to-visual (S2V) processes are interconnected. The author used GAN models to achieve bi-directional cross-modal generation, providing a direction for cross-modal generation research. [5] employed GAN models to achieve audio from visuals. In contrast to [1], this paper also used Sample RNN as an audio generator, GAN as a discriminator, and jointly outputs, but it only generates Mel spectrograms. In [4], Wang LiHao proposed a unified framework to achieve cross-modal audio and visual information output. The author introduced the CMCGAN model, which achieves audio output within this unified framework. However, this model is only suitable for specific requirements, and different generated content needs to use a pre-trained Auto Encoder to generate audio, making the generation process less concise and efficient.

Subsequent research, such as [6] and [7], introduced new model approaches that generated Mel spectrograms. Studies in the instrument domain, such as [8-10], generate Mel spectrograms for subsequent electronic audio transcription, capturing feature movements. [13] used the Audeo model to generate audio for piano video segments, transforming it based on existing piano audio segments, ultimately generating audio specific to piano content and achieving the conversion from Mel to audio. [11] used REGNET to propose a method to generate Mel spectrograms. [12] and [14] analyze corresponding spectrograms. [15]-[16] and [18] evaluate the effectiveness of the generation models by comparing their fitting degree. [19] combined the audio and video generation processes and introduced an innovative approach for multimodal joint generation. This model, using diffusion probability modeling for the generation process, significantly improved the quality and diversity of generated audio and video. Innovatively, it designed a coupled U-Net architecture to handle joint audio and video generation. This architecture can process both data modalities simultaneously, achieve cross-modal alignment through cross-modal attention mechanisms, and improve the quality and consistency of generated results.

3. Foley training datasets

3.1. Dataset overview

The construction of datasets can mainly be categorized into two situations: one involves collecting and creating data from scratch, while the other involves augmenting or referencing existing datasets. The choice of the dataset typically depends on the specific requirements of each research paper. For instance, in [1], Andrew Owens constructed the GHD dataset by manually striking different materials. Collecting dataset content can involve web videos, such as video clips from YouTube, or image

libraries like ImageNet. Depending on the task requirements, these contents are often processed, categorized, or used to create a dataset.

Datasets used for Foley tasks mostly consist of audio or video data. These contents are categorized based on the needs of the specific research. Some datasets contain a lot of noise, while others consist of silent videos or video clips. When dealing with limited sample sizes in existing datasets, data augmentation is typically based on collecting related information from existing content to find visually or acoustically similar video sound effects. For example, in the GHD dataset, the videos are comprised of sounds produced by striking or scraping 15 different material surfaces. Most Foley datasets consist of sounds from everyday life. In practical applications, distinctions are made based on duration, content, and the methods of collection.

3.2. The current status of datasets

Table 2: Dataset Acquisition Status

Dataset name	Address	Data volume	Main sound categories
GHD、Sub-URMP、AVFD、Extreme Countix-AV	As per the following table		
VIG	[6]	632 segments of 10 seconds of audio were marked, which were taken from educational videos on YouTube.	YouTube video
MUSICES	[7]	Videos of solo performances of nine musical instruments.	Instrument sound
VEGAS	[5]	In total, there are 28,109 videos with an average length of 7 seconds, totaling 55 hours	Object sounds
Piano YT	[10]	Split the data into 209 training and validation videos and 19 test videos	Over 20 hours of piano performance videos uploaded on YouTube
AFD	[17]	A total of 27,800 video samples, each lasting 3 seconds.	Diverse video samples

In this section, data sets are summarized in tabular form. This article focuses on four data sets: GHD, Sub-URMP, AVFD, and Extreme Countix-AV. The other data sets are YouTube videos or have no available open-source download links. The four mentioned data sets have complete data and textual explanations, so this article primarily reviews these data sets.

Table 3: Dataset download address

Dataset Name	Download Link	Size(GB)	Alternative Link
GHD	https://andrewowens.com/vis/	50.13	https://pan.baidu.com/s/12hrQ2y38Hlif3uoWVOsKZA?pwd=gphn
Sub-URMP	https://labsites.rochester.edu/air/projects/URMP.html	28.71	https://pan.baidu.com/s/11r644TLiiRJw6VWuZcQpxw?pwd=r9ko
AVFD	https://github.com/thuhcsi/icassp2022-FastFoley	2.13	https://pan.baidu.com/s/1rVtuSuNZVk_vxlZDtVO3UA?pwd=ut46
Extreme Countix-AV	https://drive.google.com/file/d/1eKYbN_fXetv6Dw_ks8eNeNkErGvrsDC6/view?pli=1	0.59	https://pan.baidu.com/s/1NfmGWZ900ZsoVkgQQctsA?pwd=6h58
Unable to Download Dataset (No Public Download Link Available):VEGAS、Piano-YT、AFD、VGG Sound、VAS.			
Download Dataset Using YouTube-dl：VIG [http://www.github.com/kanchen-usc/VIG You can use this repository's guidelines to obtain YouTube IDs for downloading.]、MUSICES[https://github.com/roudimit/MUSIC_dataset You can use this repository's guidelines to obtain YouTube IDs for downloading.].(YouTube-dl tool) [Tzahi12345/YoutubeDL-Material: Self-hosted YouTube downloader built on Material Design (github.com) YouTube-dl address]			

Table 2 and Table 3 present all the datasets referenced in the papers cited in this article. Among them, four datasets with specified download links will be the main datasets summarized in this article. The datasets that could not be downloaded and do not have available download links, as well as two datasets composed of YouTube videos, with marked videos that can be obtained, can be downloaded using the YouTube-dl tool, which is indicated in this article.

Table 4: Dataset Structure

Dataset Name	Whether Split into Training, Testing, Validation	Whether Includes Perspective Changes	Dataset Directory Structure	Directory Structure and File Descriptions
GHD	NO	NO	-Vis-data --*.mp4 --*.wav --*.txt	Undivided Directory mp4: Original video data of video source segments wav: Video audio tracks txt: Labels related to audio effects of video In the format "timestamp, audio category," one category per line For example: 2677438 ceramic hit static
Sub-URMP	Yes	Yes	Sub-URMP --*.wav --*.img --*.txt	Distinguish between the "chunk" and "img" directories and use 20% of the data from each directory for the validation dataset. .wav: separate each type of musical instrument sound from the original dataset. .img: extract video frames from the original dataset. .txt: provide basic information about this sub-dataset and explain the author's personal audio directory path.
AVFD	Yes	Yes	AVFD --*avi --*mp4	Undivided Directory .avi: Some videos are encoded in this format. .mp4: Most videos are encoded in this format. No .txt label information.
Extreme Countix-AV	No	No	Countix-AV --*mp4 --*wav	Distinguishing between audio and video collections, where: .wav: The audio collection corresponding to .mp4 files. .txt: A textual description of this dataset, which is a combination of Countix-AV and VGG Sound.

Table 4 provides a detailed breakdown and analysis of the data structure within each dataset directory. It also includes a statistical overview of the dataset based on the file structure. In this structure, "-Vis-data" represents a directory, "--*.wav" indicates that all the files within this folder are in the .wav format, and .mp4 signifies the file extension. This format offers a clearer organization of the dataset's file structure.

For example, a minimal structural unit within the GHD dataset consists of .mp4, .wav, and .txt files. In this structure, the .txt file contains information about the extracted frames, and the .mp4 and corresponding .wav files are stored separately as video and audio files for individual training.

Similarly, for specific video or audio files within the other datasets, we have uniformly organized the file structures, as summarized in the tables. The explanations provide detailed information about the data source, data structure, and file organization, helping to clarify the file structures within the datasets for the convenience of researchers for reference and research purposes.

Table 5: Dataset Content Statistics

Database name	Number of Sound Categories	Specific Sound Categories	Number of Video Data per Sound Category	Notes
GHD	15 categories	Ceramic Glass Clay Water Fabric Wood Paper Leaves Rock Carpet Plastic bag Gravel Plastic Plasterboard Tile Grass Metal	Impact Type: Ceramic (Quantity: 437, Percentage: 2.24%) Glass (Quantity: 382, Percentage: 1.95%) Clay (Quantity: 3279, Percentage: 16.67%) Water (Quantity: 986, Percentage: 4.99%) Fabric (Quantity: 2085, Percentage: 10.52%) Wood (Quantity: 4587, Percentage: 23.25%) Paper (Quantity: 1802, Percentage: 9.08%) Leaves(Quantity: 2515, Percentage: 12.68%) Rock (Quantity: 2795, Percentage: 14.09%) Carpet (Quantity: 377, Percentage: 1.91%) Scraping Type: Plastic Bag (Quantity: 440, Percentage: 2.12%) Gravel (Quantity: 437, Percentage: 2.12%) Plastic (Quantity: 2176, Percentage: 10.55%) Drywall (Quantity: 698, Percentage: 3.37%) Tiles (Quantity: 349, Percentage: 1.68%) Grass (Quantity: 981, Percentage: 4.73%) Metal (Quantity: 4118, Percentage: 19.78%) Unlabeled Information: Quantity: 18133, Percentage: 43.46%	Maximum labels: 17 Minimu-mlabels: 5 Average labels 10.2
Extreme Countix AV	8 categories	Change in viewpoint; Cluttered background; Fast motion; Low optical flow; Low resolution; Occlusion; Out-of-view motion; Scale variation;	Change in viewpoint: 10 minutes and 9 seconds Cluttered background: 3 minutes and 22 seconds Fast motion: 3 minutes and 26 seconds Low optical flow: 1 minute and 17 seconds Low resolution: 4 minutes and 25 seconds Occlusion: 1 minute and 36 seconds Out-of-view motion: 2 minutes and 25 seconds Scale variation: 3 minutes and 45 seconds	

AVFD	12 categories	Punching Chopping and scraping Clapping hands Footsteps Gunshot Hammering Assembly with hammer Power tools Sawing Table tennis Typing Waterfall	Boxing: 62 clips, 9 minutes and 50 seconds Cutting, Plucking: 181 clips, 13 minutes and 53 seconds Clapping: 210 clips, 16 minutes and 44 seconds Footsteps: 200 clips, 14 minutes and 49 seconds Gunshots: 65 clips, 7 minutes and 49 seconds Hammer: 1,269 clips, 1 hour and 47 minutes and 53 seconds Hammering (collective): 299 clips, 23 minutes and 13 seconds Power Tools: 922 clips, 1 hour and 16 minutes and 10 seconds Saw: 977 clips, 1 hour and 21 minutes and 34 seconds Ping Pong: 73 clips, 8 minutes and 12 seconds Typing: 97 clips, 12 minutes and 24 seconds Waterfall: 1,190 clips, 1 hour and 38 minutes and 17 seconds.	
Sub URMP	8 categories	Bassoon Cello Clarinet Double Bass Flute Horn Oboe Saxophone Trombone Trumpet Tuba Viola Violin	Bassoon: 1735 pairs, 9 minutes and 50 seconds Cello: 9800 pairs, 13 minutes and 53 seconds Clarinet: 8125 pairs, 16 minutes and 44 seconds Double Bass: 1270 pairs, 14 minutes and 49 seconds Flute: 5690 pairs, 7 minutes and 49 seconds Horn: 5540 pairs, 1 hour, 47 minutes and 53 seconds Oboe: 4505 pairs, 23 minutes and 13 seconds Saxophone: 7615 pairs, 1 hour, 16 minutes and 10 seconds Trombone: 8690 pairs, 1 hour, 21 minutes and 34 seconds Trumpet: 1015 pairs, 8 minutes and 12 seconds Tuba: 3285 pairs, 12 minutes and 24 seconds Viola: 6530 pairs, 1 hour, 38 minutes and 17 seconds Violin: 7430 pairs, 1 hour, 8 minutes and 37 seconds	Preprocess and train on 20% of the total dataset.

Table 4 mainly introduces the data types in the datasets, and statistics were conducted based on the video categories in these four datasets. In these datasets, GHD uses common materials and produces sound effects by hitting or scraping them with a stick. GHD classifies sounds into scraping and hitting categories based on the surface properties of the materials, and statistics are done based on the number of labeled information to calculate the proportion and number of labeled videos.

Extreme Countix-AV is constructed by collecting publicly available video data, which do not have a consistent or fixed subject. Therefore, the videos are categorized based on the angles of shooting, such as viewpoint changes: shooting moving objects; scale changes: zooming videos; occlusion: covering part of the frame. These videos lack clear labeling information and can be considered as supplementary data for additional training in the alignment of audio and video tasks.

AVFD is composed of actions, tools, and natural scenes, and it is made up of many video clips that may only contain video footage of a momentary action. The total duration of these segments and the number of related videos are counted to facilitate standardized training for specific tasks.

Sub-URMP is a subset of URMP and primarily consists of musical instrument sounds, including various common orchestral instruments. The dataset focuses on eight classes of musical instruments, and it was processed and organized based on the directory structure. Statistics were conducted on the

audio-video pairs, and the total duration is calculated. This dataset is often used for tasks related to music transcription generation and audio restoration.

Table 6: Statistics of Video Basic Information

Database	Dataset Quantity	Total Duration	Average Duration	Resolution	Frame Rate
GHD	Collect Original Videos: 977 Segment Clip Videos: 2931 Number of Categories: 54 Total Number of Data Entries: 41,883	29h41min26s	36s	1920*1080p 600*340p	29.97FPS
Sub-URMP	17555 video pairs	19h51min17s	6min17s	1920*1080p	2FPS
AVFD	5565 videos	4h23min25s	8min16s	640*360p	29.97FPS
Extreme Countix-AV	233 video entries and 224 audio entries	2h 47s	3min7s	1280*720p	24FPS

Table 6 primarily provides essential information about the dataset, including video duration, resolution, and frame rate, along with statistics on the number of videos. In terms of content composition, we categorized the data based on video content, labeling, sound categories, the number of entries, and proportion. Basic video information, such as the number of videos, duration, frame rate, and resolution, is summarized in Table 6.

Through this section's organization of the dataset, we aim to offer a comprehensive and structured dataset summary, enabling precise and efficient access to the data. Additionally, the Python code used for dataset organization has been updated on the GitHub repository, Here is the link: https://github.com/stq5515/overviewcode

3.3. Dataset Evaluation Methods

The quality of a model's generated output depends on the training quality of the dataset. To better assess the actual performance of a model, various evaluation metrics are typically used, such as IS (Inception Score), FID (Fréchet Inception Distance), SVM classifier, and more. IS evaluates generated image quality by calculating the entropy of the classification probability distribution of generated images and the mean of real images. FID measures the distance between the feature representations of generated images and real images, reflecting the difference in distribution between them. SVM classifier evaluation tests whether generated images can be correctly classified by a pretrained classifier, indirectly indicating image quality and recognizability. In addition to the mentioned methods, there are many other evaluation approaches available. Researching these evaluation methods provides an intuitive understanding of a model's generative capabilities.

In [2], Support Vector Machine (SVM) was chosen as the evaluation criterion. The authors used grid search on the validation set to maximize the Mean Average Precision (mAP). This evaluation revealed that the author's binary-based model approach had the most significant advantages. Additionally, [17] and [13] both used the Inception Score (IS) for assessment. The core idea of IS involves utilizing a pretrained image classification model, such as the Inception network, to classify the generated images. Typically, IS calculates the entropy of the probability distribution of the generated images' classifications to measure their diversity. It also computes the average classification probability of the generated images on the pretrained classifier to assess their authenticity. These two metrics are combined to generate a comprehensive score, used to measure the quality and diversity of the generated images.

Specifically in the research described in the papers, [17] obtained the IS score by calculating the KL divergence between the conditional category distribution of generated samples and the marginal category distribution. The authors also used features from a pretrained audio retrieval CNN to compute the IS score to assess the quality of generated spectrograms. In [13], they evaluated semantic

diversity by calculating the Kullback-Leibler (KL) divergence between MIDI generated samples and original samples. [8] computed the average cross-entropy loss for three proposed models (Frame, Seq, and Flow) during training and testing, comparing them with the source data. The authors designed retrieval experiments using visual features as queries to retrieve the most matching audio from the database.

In addition to the comprehensive evaluation methods, some papers also introduced their own evaluation models. For example, [6] set up an original audio task based on the GHD dataset ([1]). They performed Recall at top K and classification tasks and compared the model's performance by comparing classification accuracy on the author's trained model and accuracy on a five-layer neural network classifier. [7] utilized PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index). PSNR measures the similarity between the original and reconstructed images by calculating the mean square error (MSE) between them. SSIM typically calculates brightness similarity, contrast similarity, and structural similarity. They are then weighted and summed to obtain a comprehensive similarity index. SDR (Signal-to-Distortion Ratio) calculates the signal-to-noise ratio between the original and distorted parts of the signal to assess the severity of distortion. OPS (Overall Perceptual Score) is based on deep learning model training and simulates the human visual system's judgment of image quality better. OPS is calculated using a trained neural network to predict the difference between human subjective ratings and model output for evaluating the quality of the final restored original audio results.

The aforementioned evaluation methods are all based on machine-based approaches, involving computational methods for assessment. For most sound synthesis models, conducting human subjective evaluations is a more widely used approach to provide an intuitive representation of the actual sound synthesis quality. In [1], [3-5], for instance, psychological experiments were conducted using Amazon Mechanical Turk (AMT) by recruiting volunteers. Participants were asked to differentiate between synthesized audio and real sound effects. A three-person evaluation model was employed, requiring judgments on authenticity. Additionally, comparative experiments were conducted to compare existing synthesized sound effects with the results of previous research models. By using benchmark comparisons, the current models' synthesis quality could also be assessed.

Table 6: Summary of Evaluation Methods

Evaluation Method	Explanation
Subjective evaluation	[1] conducted a psychophysical test on Amazon Mechanical Turk, where participants were asked to distinguish between real and synthesized sounds. Subjective evaluations are almost used in every paper to assess the quality or distinguish the effectiveness of generated sounds, often through rating or discrimination.
IS ID FID	**IS：Inception Score，** a metric used to assess the quality of images generated by GANs (Generative Adversarial Networks). It involves generating a large number of image samples from a generative model and using an Inception-v3 CNN model to classify these generated images. IS measures image clarity and diversity by calculating the entropy of the classification distribution for each image and the average KL divergence (Kullback-Leibler Divergence) between image classifications. **ID：Identity Disentanglement** the separation of identity information (often a person's or object's identity) from other features in the input data. Identity disentanglement is achieved by training a model to learn how to represent and recognize different identity features. **FID：Fréchet Inception Distance，** a metric used to evaluate the quality of images generated by Generative Adversarial Networks (GANs). A lower FID score indicates that the generated images are more similar to real images in terms of distribution, reflecting higher quality. FID is an effective metric for assessing image generation quality, as it considers both visual quality and diversity of images, not just the visual quality alone.
comparative evaluation	Comparisons are mostly conducted by training the same model on different datasets and then evaluating the generated results. For example, in studies [4–6], three-person evaluation experiments were conducted on Amazon Mechanical Turk (AMT) to directly compare the sound generation performance of each proposed method. In addition to this, study [4] aimed to validate the superiority of CMCGAN by comparing it with baseline models S2I and I2S (models from [3]).

This paper has reviewed the situations of mainstream datasets, including the total video length, the number of videos, the categories included, as well as the resolution, frame rate, and other basic information. The datasets have been labeled, categorized, and organized based on corresponding

textual timestamps, establishing an overall structure for data segmentation. The process of constructing and analyzing datasets can reveal the characteristics, distribution, and underlying patterns of the data, providing robust support for addressing complex real-world problems.

4. Conclusion

This paper has provided an overview of the recent developments in Foley research, categorizing tasks into specific domain-based sound synthesis, general-purpose sound synthesis, and video sound restoration. These diverse research directions and tasks aim to enhance the precision of sound generation in specific domains. The paper also systematically summarizes mainstream deep learning models, highlighting their advantages and disadvantages. It suggests that the Foley sound synthesis field can further benefit from combining various deep learning models to propose new frameworks for high-quality sound generation.

In the future, Foley sound generation tasks can explore the following areas:

(1) More diverse and high-quality open datasets encompassing a broader range of sound categories.

(2) Standardized benchmark evaluation systems that allow for the comparison and optimization of different training frameworks using consistent metrics.

(3) The exploration of automated Foley sound synthesis methods that offer high quality, high fidelity, and the ability to generate sounds of varying lengths. Current methods often focus on a single sound category, have limited flexibility in sound duration, or may lack sound variety.

The paper also presents an introduction to the basic status of Foley sound applications, generated results, and architectural frameworks. In the section related to dataset research, it discusses the training results using existing datasets and highlights differences in features when various models are trained on the same dataset. Additionally, it provides essential information about various datasets, including content, updates, and download locations. This systematic review aims to provide a fundamental understanding of the current state of Foley research, facilitating further in-depth investigations.

References

[1] Andrew Owens， Phillip Isola，Josh McDermott，Antonio Torralba，Edward H. Adelson, William T. Freeman al. "Visually Indicated Sounds." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[2] Andrew Owens, Jiajun Wu, Josh H. McDermott, William T. Freeman, Antonio Torralba al. "Ambient Sound Provides Supervision for Visual Learning." In Proceedings of the European Conference on Computer Vision (ECCV), 2016.

[3] Lele Chen, Sudhanshu Srivastava, Zhiyao Duan, Chenliang Xu， et al. "Deep Cross-Modal Audio-Visual Generation." In Proceedings of the ACM International Conference on Multimedia (ACM MM), 2017.

[4] Wangli Hao, Zhaoxiang Zhang, He Guan, et al. "A Uniform Framework for Cross-Modal Visual-Audio Mutual Generation." In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2018.

[5] Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L. Berg et al. "Visual to Sound Generating Natural Sound for Videos in the Wild." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[6] Kan Chen, Chuanxi Zhang, Chen Fang, Zhaowen Wang, Trung Bui, and Ram Nevatia, et al. "Visually Indicated Sound Generation by Perceptually Optimized Classification." In Proceedings of the International Conference on Multimodal Learning (MULA), 2018.

[7] Hang Zhou, Ziwei Liu, Xudong Xu, Ping Luo, Xiaogang Wang, et al. "Vision-Infused Deep Audio Inpainting." In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2019.

[8] Chuang Gan, Deng Huang, Peihao Chen, Joshua B. Tenenbaum，and Antonio Torralba et al. "Foley Music: Learning to Generate Music from Videos." In Proceedings of the European Conference on Computer Vision (ECCV), 2020.

[9] Huadong Tan, Guang Wu, Pengcheng Zhao, Yanxiang Chen, et al. "Spectrogram Analysis Via Self-Attention for Realizing Cross-Model Visual-Audio Generation." In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2020.

[10] A. Sophia Koepke, Olivia Wiles, Yael Moses, Andrew Zisserman et al. "Sight to Sound: An End-to-End Approach for Visual Piano Transcription." In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2020.

[11] Peihao Chen, Yang Zhang, Mingkui Tan, Hongdong Xiao, Deng Huang, and Chuang Gan et al. "Generating Visually Aligned Sound from Videos." In Proceedings of the Institute of Electrical and Electronics Engineers (IEEE) 2020.

[12] Sanchita Ghose, John J. Prevost, et al. "Auto Foley: Artificial Synthesis of Synchronized Sound Tracks for Silent Videos with Deep Learning." In Proceedings of the Institute of Electrical and Electronics Engineers (IEEE) 2020.

[13] Kun Su, Xiulong Liu, Eli Shlizerman, et al. " Audeo: Audio Generation for a Silent Performance Video." In Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)2020.

[14] Sanchita Ghose，John J. Prevost et al. "Enabling an IoT System of Systems through Auto Sound Synthesis in Silent Video with DNN." In Proceedings of the SoSE 2020 • IEEE 15th International Conference of System of Systems Engineering (SoSE) 2020.

[15] Katashi Nagao，Kaho Kumon，Kodai Hattori。et al. "Impact Sound Generation for Audiovisual Interaction with Real-World Movable Objects in Building-Scale Virtual Reality." In Proceedings of the Applied science (AS) 2021.

[16] Vladimir Iashin, Esa Rahtu, et al. "Taming Visually Guided Sound Generation." (BMVC) 2021.

[17] Sanchita Ghose, John J. Prevost, et al. "Foley GAN: Visually Guided Generative Adversarial Network-Based Synchronous Sound Generation in Silent Videos." In Proceedings of the Institute of Electrical and Electronics Engineers (IEEE) 2021.

[18] Yuexi Du, Ziyang Chen, Justin Salamon, Bryan Russell, Andrew Owens, et al. "Conditional Generation of Audio from Video via Foley Analogies." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2023.

[19] Ludan Ruan, Yiyang Ma, Huan Yang, Huiguo He, Bei Liu, Jianlong Fu, Nicholas Jing Yuan, Qin Jin, Baining Guo. "MM-Diffusion Learning Multi-Modal Diffusion Models for Joint Audio and Video Generation" In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2023.