SonicRender：结构化且时间锚定的多事件音频场景生成渲染

Yunrui Cai[1,2,*]、Xu Li[1,†]、Yan Rong[1],Dingdong Wang[2],Yucheng Zhou[1],Xixin Wu[2], Chen Zhang[1],Zhiyong Wu[3],孟海伦[2,†]
[1]Kling AI[2] 香港中文大学[3]清华大学深圳国际研究生院{yrcai, hmmeng}@se.cuhk.edu.hk, lixu15@kuaishou.com

arXiv:2611.30241v1 [cs.SD] 23 Nov 2026

摘要—**虽然统一音频生成器能够在单个模型中合成语音、音乐和音效，但生成合理的混合音频只是挑战的一部分：复杂的场景请求还会指定事件发生的内容、由谁产生，以及何时被听到。为在单次生成过程中满足这些组合约束，我们提出了 SonicRender，一种统一的多事件音频渲染器。其时序音频场景图（TASG）将持久声源身份与局部化的语音、音乐、环境声和音效事件绑定起来，同时关联这些事件的内容、声学角色、关系以及带时间戳的时间窗口。我们进一步引入 AbsTime，为音频潜变量流提供绝对帧坐标和有效画布时长。通过联合文本–音频注意力和条件流匹配，该渲染器学会将结构化、定时的场景实现为连贯的混合音频。在 MMAG 时间戳基准上，SonicRender 的语音/声音 F1 分数达到 0.79/0.60，超过了我们对比系统中已报告的最佳分数。在 SceneMix 评测中，它取得了 85.4% 的全事件召回率、83.9% 的时间窗口命中率和 95.7 % 的说话人准确率。主观研究进一步表明，在感知音频质量和场景实现方面，该方法均显著优于全部五个对比系统。在语音、通用音频、音乐、多事件负载和时间锚定分析中的结果表明，统一生成器能够实现精确且具备组合感知能力的音频渲染。演示页面：https://caiyunrui.github.io/SonicRender。**

索引词—**统一音频生成、多事件音频场景、结构化条件控制、时间定位、可控音频合成、扩散变换器。**

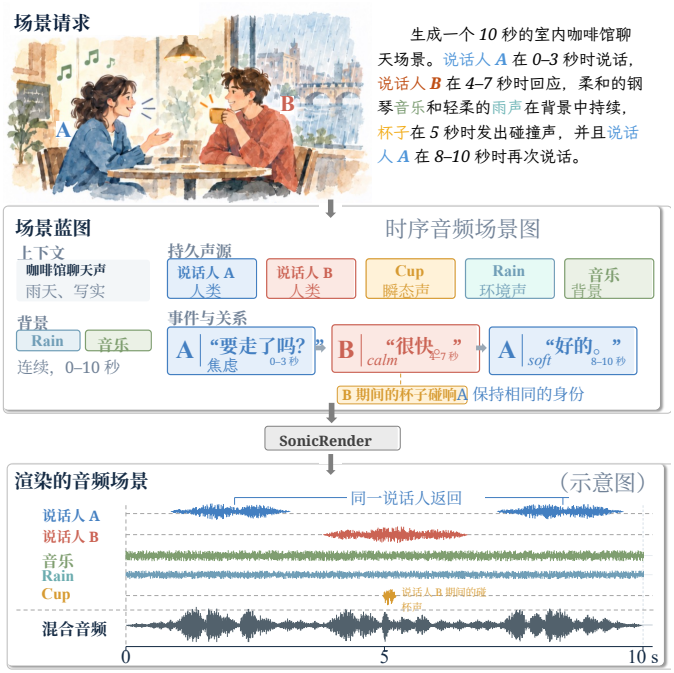

图 1. SonicRender 概览。场景请求被组织为 TASG 蓝图，将持久声源与事件、关系和时间窗口绑定起来。统一渲染器将此蓝图转化为连贯的音频混合，同时确保声源连续性和事件级时间控制。

I. 引言

RECENT 统一音频生成器在共享架构中合成语音、音乐、环境声音及其混合音频 [1]–[4]。然而，渲染逼真的音频场景需要的不只是混合声学纹理：单个请求可能会指定反复出现的角色之间的对话，
贯穿对话的背景音乐、一段打断某轮对话的短暂音效，以及每项活动发生的确切时刻。要实现这样的场景，需要共同控制发生哪些事件、由谁产生、它们如何共存，以及何时被听到。我们的目标是在一个生成的波形中，实现多事件场景的事件完整、源一致和时间锚定 渲染。

即使生成的混合音频整体听起来合理，也可能遗漏轻微的音效、把台词分配给错误的说话人、截断持续播放的背景声，或将某个事件放错时间。随着重叠活动和重复出现的对话轮次不断累积，这些问题会迅速叠加。近期的混合音频研究和基准测试 [2], [4], [5] 强调，事件完整性、时间顺序和说话人一致性是全局音频质量之外的关键瓶颈。要实现较高的组合保真度，就必须在生成过程中始终确保各个声源和事件约束可被识别，同时在单一混合音频中对它们的声学交互进行建模，而不是分别合成独立音轨。

首要挑战是在分解过程中维持声源—事件绑定。声源具有持久的声学身份，而事件则是局部发生的现象：说话人之后可能再次出现，说出新的话语并采用不同的表达方式；音乐声源则可能贯穿多个对话轮次。现有的结构化条件可以表示事件顺序、时间图或角色配置 [4], [6]–[8], ，但扁平或关联松散的描述仍会导致属性在并发事件之间相互混淆。我们通过明确的引用，将持久声源绑定到局部事件

*在 Yunrui Cai 于 Kling AI 实习期间完成的工作。
†通讯作者：Xu Li 和孟海伦。

上：共享的声源记录将身份与特定轮次的内容分开，而事件记录和混合关系则能在重叠情况下区分短暂音效与持续背景。这种组织方式直接针对上述身份混淆、事件遗漏和层截断问题。

第二个挑战是在非自回归潜在生成中对符号时间进行定位。现有方法通过时间矩阵、结构化文本或推理时引导 [9]–[12], 来控制时序，而在音视频生成中也存在类似的脚本到输出的时间对应差距 [13]。然而，“6 秒”这样的文本时间戳是一种符号指令，而扩散变换器（DiT）则会并行对可变长度的潜在序列进行去噪。仅凭相对序列位置（例如旋转嵌入）既无法在不同音频时长之间将某个物理秒数锚定到固定的潜变量帧，也无法传达有效画布的结束位置。我们引入直接的音频侧度量参照，使任意物理时间戳无论场景时长或事件数量如何，都能映射到一致的潜在坐标。

我们提出 SonicRender，一种将源感知规范与音频侧度量时间定位相结合的统一渲染器。其时序音频场景图（TASG）在一个序列化条件中，将持久声源与局部事件、内容、角色、关系和度量窗口绑定起来。AbsTime 为音频隐藏状态添加绝对帧编码，并将有效画布时长加入全局条件。双流多模态扩散变换器（MM-DiT） [14] 通过联合注意力整合事件级文本，并通过全局调制整合场景摘要。经条件流匹配 [15], 训练的共享路由音频容量，将语音、音乐、环境声和音效渲染为一个混合音频。因此，TASG 规定什么发生以及何时发生，而 AbsTime 则将该时间定位到音频时间线上。

公共演讲、通用音频和音乐评测表明，单个检查点即可实现广泛的生成能力。语音评测还涵盖日语、韩语和西班牙语，更多语言、方言和口音则展示于我们的演示页面。在 MMAG [5], 中，SonicRender 相对于原始的小数秒真实窗口取得了 0.79/0.60 的语音/声音 F1，在我们的对比中领先于所报告的系统。在 168 个 SceneMix 请求上，它达到了 85.4% 的全事件召回率、83.9% 的窗口命中率和 95.7% 的说话人准确率；相比之下，信息量匹配的自然语言变体分别为 74.6%、68.0% 和 86.6%。事件负载曲线和匹配消融实验进一步证明了其在更密集组合下的性能提升，以及音频侧坐标在起始位置定位中的作用。一项由 15 名听众参与的研究证实，与全部五个对比系统相比，这些客观评估方面的提升还伴随着显著更好的音频质量和场景实现效果。我们的主要贡献如下：

- 我们提出 TASG，这是一种具备源感知能力且具有时间分辨性的条件表征，将重复出现的声学身份与局部事件、内容、角色、关系和度量窗口绑定起来，用于联合多事件渲染。
- 我们提出了 AbsTime，这是一个轻量级模块，通过绝对帧坐标和有效画布时长为符号时间戳提供定位，为潜在音频生成器中的时间控制提供了一种可复用的设计。
- 我们开发了 SonicRender，这是一种统一渲染器，将 TASG 与联合文本—音频注意力相结合，以实现源—事件交互；通过全局调制提供场景级上下文，并借助共享与路由式声学容量处理异质重叠事件，从而在一个混合音频中生成语音、音乐、环境声和音效。
- 我们引入 SceneMix，这是一套组合式评测套件，涵盖多事件音频场景中的事件完整性、重复说话人一致性和时间遵循度。在 SceneMix 及既有基准上的评测，以及受控消融实验和人工听测，证明了 SonicRender 的有效性。

II. 相关工作

A. 统一音频与多源场景生成

AudioLDM 2 [16],UniSonate [1], UniAudio 2.0 [17],AudioCALM [18], StepAudio 3 Gen [19], 和 AudioX [20]研究跨声学领域或条件模态的共享表征与生成。MiDashengLM-Gen [3]和 Ming-Omni-TTS [21] 进一步在由 LLM 驱动或全模态的框架内统一语音、音乐和声音生成。然而，广泛的声学覆盖范围本身并不能确保混合场景中的每个声源和事件都遵循各自的单独指令。

Dasheng AudioGen [2] 使用结构化多视图描述和语义—声学潜变量，原生合成混合场景。SonicWeave [22] 引入分块路由混合专家，以处理异构且局部连续的声学内容。我们关注如何对这类混合音频进行更精细的控制：实现单独指定的事件和时间窗口，同时保持重复出现的身份和持续背景声。声学容量和场景级连贯性关注的是混合音频听起来如何；源—事件控制还关乎每项请求的活动是否保留其内容、归属和位置。这要求将源—事件组织与渲染器中的物理时间基准相结合。

VibeVoice [23] 面向长时对话轮次处理，ScenA [24] 则依据多个参考语音和声学上下文来生成对话。我们的文本指定场景会将说话者重复出现与音乐、环境声和音效结合起来，而无需强制提供参考语音。WavJourney [25],、AuDirector [26], 和 Audio-Oscar [27] 则侧重协调专用生成、规划或后期制作阶段。我们使用单个渲染器直接生成完整的混合音频。

尽管 Qwen-Audio-3.0-Gen-Preview [4] 也在 DiT 骨干网络中采用了角色配置和时间线格式化，但它依赖文本侧时间标记，缺乏明确的音频侧度量坐标或有效时长基准，也未显式建模跨事件声学角色（例如持续的背景—前景关系）。SonicRender 将关系式源—事件场景图与明确的音频侧帧与时长锚定相结合，从而在统一生成流程中实现严格的完整性和时间遵循度。

B. 结构化条件与声源–事件绑定

结构化条件能显式呈现扁平化字幕可能混杂的属性。Make-An-Audio 2 [6] 采用事件顺序规划；

AudioComposer [28]通过细粒度描述处理内容和风格。DegDiT [7]使用图变换器嵌入事件语义和时间链接，而ControlAu- dio [8]则通过渐进式条件机制使用带分隔符的事件文本和时间。Qwen-Audio-3.0-Gen-Preview [4] 采用场景/角色/事件层级。除了指定这些语义单元之外，直接进行混合渲染还要求这些单元的归属及其交互在各个事件间保持一致。

相比之下，TASG 将持久的源身份与局部事件实例解耦，并通过显式的引用ID和混合关系将二者关联起来。这使重复出现的说话人能够在不同轮次中改变其词汇内容和表达方式，同时保持声音身份；此外，借助联合文本—音频注意力，重叠的前景与背景层也能分别进行寻址。

C. 事件级时间控制与指标落地

Stable Audio [29] 采用时间条件潜在扩散；MC-Diffusion [30] 增加了时间戳和声学风格控制。PicoAudio [9] 和 PicoAudio2 [10] 利用时间上对齐的数据和时间矩阵来引导事件。ControlAudio [8] 通过渐进式训练和推理融入结构化时间规划与音素条件，而DegDiT [7]则将时间属性与事件语义及图链接相结合。这些方法在请求的时间如何进入生成过程中方面各不相同。

推理时方法包括：Temporal Guidance Diffu- sion [31], ，通过更新潜变量来实现所需的事件区域；FreeAudio [11], ，结合时间规划、注意力控制和潜在表示组合；以及 TAG [12], ，利用时间描述引导注意力。Audio ControlNet [32]提供辅助的局部控制特征。在视听生成中，Temporal Context Routing [13] 通过脚本感知路由处理脚本到输出的时间对应关系。

SonicRender 通过轻量级的 AbsTime，在符号化文本时间戳与并行潜在去噪之间建立桥梁，无需在推理时改造注意力机制，也不依赖外部时间矩阵。通过以固定潜在速率注入未归一化的帧索引，并同时提供有效画布时长，渲染器在端到端训练过程中学会了文本时钟值与实际音频位置之间的直接对应关系。

D. 评估组合与时间保真度

全局音频–文本相似度（如 CLAP）和分布距离（如 FAD、KL 散度）无法判断各项声源、事件及时序指令是否得到满足。近期基准测试开始转向更细粒度的评估：Audio-Time [33] 测试时间对齐的声音描述，而T2A-Feedback [34] 和指令遵循协议 [35]则考察事件存在性与顺序关系。MMAG [5] 将评估扩展到涵盖语音、音乐、音效及带时间戳片段的混合音频。

然而，现有基准测试很少对 Dense 多事件组合进行压力测试，也很少评估在没有声音参考的情况下，跨重复出现的对话轮次、通过文本提示实现的说话人身份一致性。作为对 MMAG 的补充，我们提出 SceneMix，用于联合评估事件召回率、非语音保留率、重复说话人准确率以及在不断增加的组合负载下的时间窗口遵循度。

III. 方法

A. 时序音频场景图

我们用一个结构化蓝图表示所请求的声学场景

$$\mathcal{B} = (\mathcal{G}, \mathcal{S}, \mathcal{E}, \mathcal{R}, \mathcal{T}), \tag{1}$$

其中， $\mathcal{G}$ 描述全局上下文， $\mathcal{S}$ 包含持久声源， $\mathcal{E} = \{E_k\}_{k=1}^{K}$ 包含事件实例， $\mathcal{R}$ 指定跨事件关系， $\mathcal{T} = \{\tau_k\}_{k=1}^{K}$ 汇集度量时间约束。事件定义为

$$E_k = (s_k, c_k, a_k, r_k, \tau_k), \tag{2}$$

其中， $s_k \in \mathcal{S}$ 是其源标识符， c_k 是其语言或语义内容， a_k 是其局部属性， r_k 是其声学角色， $\tau_k = [b_k, e_k]$ 是其在指定时的请求时间窗口。稳定的音色特征属于源记录；语言、情感、表达方式、时间范围和重复性则属于各个事件。因此，同一说话人可以在不同轮次改变表达方式，而不必成为新的声源；简短的音效也与持续的背景保持区分。关系用于指定诸如背景衬托之类的声学交互，而事件区间也表达顺序关系和重叠。$\mathcal{T}$ 汇集这些区间，而不是将其重复记录为第二套时间表。

我们将这种序列化形式称为时序音频场景图（TASG）。规范文本按顺序排列场景上下文、源记录、事件记录和混合关系。字段分隔符和标识符保留在提供给编码器的文本中；图的连通性则通过可重复使用的引用来表达。规范化格式减少了表达等效场景要求的方式差异，而源—事件因子分解将可重复使用的身份信息与事件特定的内容、表达方式和时序分离开来。两者结合，简化了渲染器必须学习的场景指令与声学实现之间的关联。以下简化示例说明重复源绑定：

<类型="语音,音乐"，2人咖啡馆聊天/><来源 id="A" 类型="说话人"/><来源 id="B" 类型="说话人"/><来源 id="M" 类型="音乐"/><事件 id="E1" 来源="A" 时间="[00:00 - 00:03]" 类型="语音">"要走了吗？"</事件><事件 id="E2" 来源="B" 时间="[00:04 - 00:07]" 类型="语音">"很快。"</事件><事件 id="E3" 来源="A" 时间="[00:08 - 00:10]" 类型="语音">"好的。"</事件><事件 id="E4" 来源="M" 时间="[00:00 - 00:10]" 类型="音乐" 角色="背景" 范围="持续">轻柔的钢琴声</事件><混音>E4 背景_至 E1 |E4 背景_至 E2 |E4 背景_至 E3</混音>

重复的标识符 A 将两个语音轮次绑定到同一说话人，同时 E1 和 E3 保留各自不同的内容和时序。语音和歌词均按原样存储；非人声事件则包含

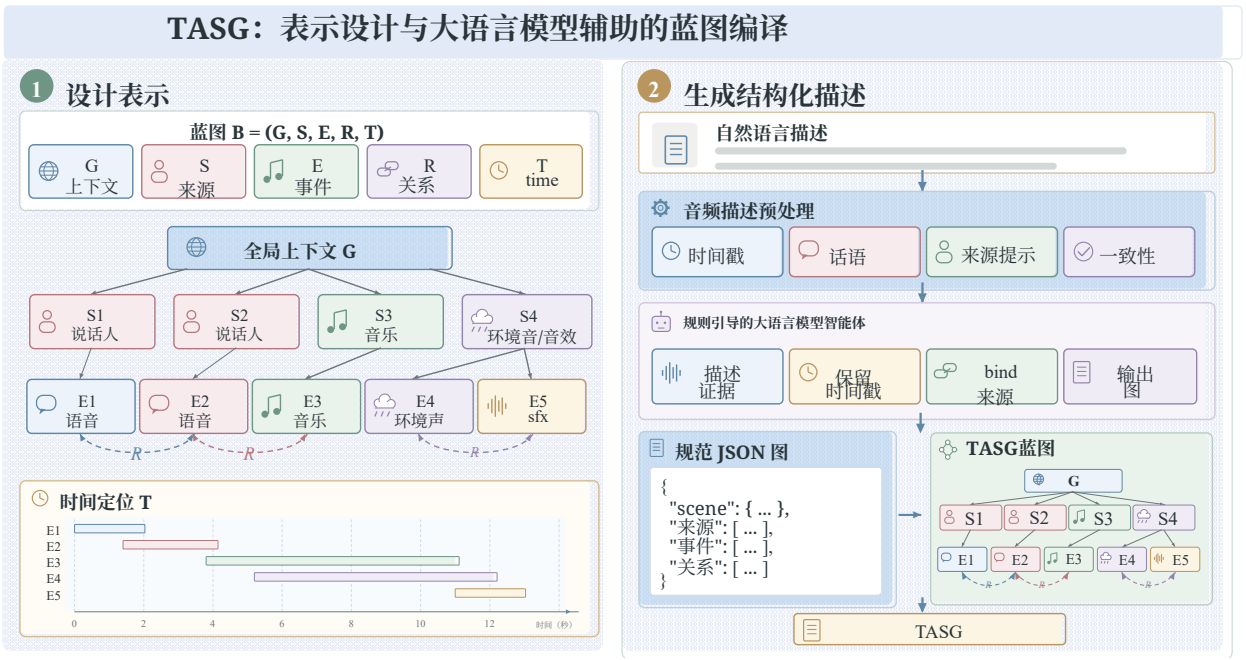

图 2. TASG表示与大语言模型辅助编译。左：全局上下文、持久声源、事件实例、关系和时间窗口共同定义场景蓝图。右：规则引导的大语言模型将自然语言描述整理为规范化场景图，并序列化为定时生成条件。

声学描述。这种因子分解方式在同一条件下明确呈现了声源连续性和事件局部变化。时间戳采用普通文本时钟值，并由与其余字段相同的分词器处理。因此，渲染器可直接通过文本流读取结构化场景。

对于训练数据标注，我们使用 AudioMap [36]，获取时间对齐的密集音频描述，该描述共同捕捉带时间戳的语音话语、说话人特征和非语音声学事件，然后将其转换为规范化TASG记录，同时保留所描述的事件内容、声源属性和时间戳。对于自由形式输入，规则引导的语言模型编译器可以将描述转换为规范化JSON场景图，再确定性地格式化为紧凑的TASG标签序列。这为明确的定时命令和普通场景请求提供了统一接口。渲染器级实验直接使用经过审核的规范化规格，因此实验能够单独考察模型在约束条件明确后实现该场景的能力。图 2 展示了这种表示与编译流程。

在条件训练中，结构化文本输入与音频混合构成一个监督配对。条件中包含声源关联和事件区间，目标混合音频中则包含相应的声学实现。这种组织方式使学习目标能够接触到反复出现的身份信息和局部时间属性，而无需声源分离波形或额外的对齐网络。

B. 音频侧度量时间定位

在 TASG 中，事件窗口 τ_k 以物理秒表示，而音频生成器处理的是潜在序列。设 f_s 为波形采样率，d为音频自动编码器下采样因子。潜在速率为 $r = f_s/d$，物理时间 τ 大致对应于音频帧 $f(\tau) = r\tau$。AbsTime 提供了两种互补信号，使这种度量关系可用于音频计算。

每个有效的潜变量帧 $f \in \{0, \ldots, F_i - 1\}$ 都先经过正弦位置映射，再输入两层 SiLU多层感知机 $a_f = \phi(f)$进行编码。音频隐藏状态变为

$$\tilde{h}_f = h_f + g_{\text{abs}}\phi(f). \tag{3}$$

正弦映射使用单位尺度的帧索引，其频率为

$$\omega_j = 10000^{-j/(H-1)} \tag{4}$$

分量为 $[\sin(f\omega_j), \cos(f\omega_j)]_{j=0}^{H-1}$，其中 $2H$为正弦嵌入维度。该坐标是绝对的，而非根据画布长度 F_i进行归一化：在潜在速率固定时，不同时长下的同一物理时间会获得相同的嵌入。AbsTime 通过在音频隐藏状态中加入显式帧参照，补充骨干网络的旋转位置编码；该参照不仅参与查询和键的计算，也参与注意力值及后续的前馈计算。因此，它在整个音频通路中提供了直接的度量时间信号。

从样本的有效掩码中获得的有效音频潜在帧数量 F_i，经投影 ψ 编码后，添加到全局扩散条件向量 q_i 中：

$$\tilde{q}_i = q_i + g_{\text{dur}}\psi(F_i). \tag{5}$$

这里 q_i 是基础流条件向量，而 ψ 采用相同的正弦波–MLP结构，但使用独立参数。逐帧通路负责定位音频词元；全局通路则传达有效时间线的延伸范围。

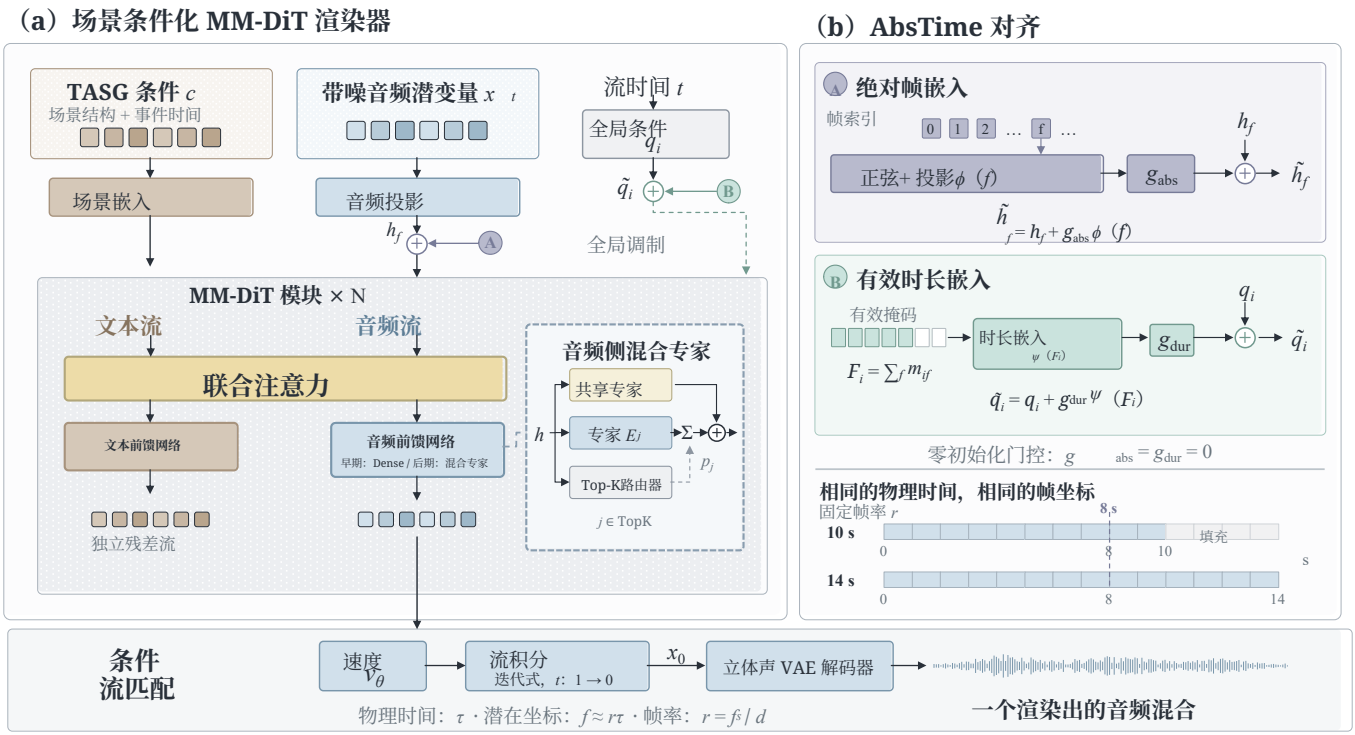

图 3. SonicRender 架构与 AbsTime 对齐。(a) 双流 MM-DiT 将联合文本—音频注意力与后续的音频侧混合专家模块相结合，以渲染一个混合音频。(b) 零门控的绝对帧索引和有效时长嵌入提供时间坐标：在不同画布长度下，同一物理时间保持相同的帧索引。

下方渲染器中的时长增强型 $\tilde{q}_i$ 同时驱动文本池化和自适应层归一化。当为无分类器引导丢弃语义条件时，两种时间信号仍保持启用，因此有条件和无条件预测都指向同一音频画布。标量门控g_{abs} 和 g_{dur} 初始值为零，从而在初始化时保留底层渲染器，并逐步引入度量时间信息。填充内容不参与注意力和训练损失，也不会影响 F_i。

C. 统一渲染器

我们的渲染器是一个双流多模态扩散 Transformer（MM-DiT） [14]。音频自动编码器将完整的混合音频表示为连续潜变量 $z \in \mathbb{R}^{F\times d_z}$。我们使用冻结的 Qwen3-VL-8B-Thinking 模型 [37]，以纯文本模式将 $c = \text{Serialize}(\mathcal{B})$ 编码为词元级隐藏特征$U \in \mathbb{R}^{L\times d_u}$，其中包含 $d_u = 4096$，随后通过学习得到的线性投影映射至 Transformer 维度。这个共享文本编码器为不同声学领域提供条件接口。音频输入将带噪潜变量与可选的掩码参考潜变量拼接后进行投影，再经过残差卷积位置嵌入和 AbsTime。对于纯文本生成，参考潜变量为零。

a) 局部交互和全局场景条件：条件以两个互补层级输入。联合注意力保留对各个来源和事件的令牌级访问。同时，条件感知池化总结场景根据当前流状态和有效时长：

$$\alpha_j = \text{softmax}_j\left(\frac{(W_q\tilde{q})^\top W_k U_j}{\sqrt{d_u}}\right), q_{\text{scene}} = \tilde{q} + W_g\sum_{j=1}^{L}\alpha_j U_j. \tag{6}$$

池化时会排除填充的文本词元。池化后的条件通过每个模块中的独立 AdaLN-Zero 模块调制两个流，并调制最终音频输出的归一化。投影后的 $\tilde{q}$ 词元也会添加到文本流的开头。因此，详细的事件约束仍可单独访问，而依赖流动方式和时长的场景摘要则对整个渲染计算进行条件化。

在每个模块中，文本和音频分别拥有独立的查询／键／值投影、输出投影、残差路径和前馈网络。遵循 M-RoPE [38], 的模态专属坐标原则，我们采用模态解耦的一维 RoPE，文本和音频的位置分别独立起始。查询和键经过 RMS 归一化，单个非因果注意力操作则覆盖拼接后的流：

$$[O_c; O_x] = \text{softmax}\left(\frac{[Q_c; Q_x][K_c; K_x]^\top}{\sqrt{d_h}} + M\right)[V_c; V_x], \tag{7}$$

其中，下标 c 和 x 分别表示文本条件流和音频流，d_h 表示注意力头的维度，M 用于屏蔽填充。输出会被拆分回各自对应的流中。音频旋转位置索引独立于描述长度。该模块将源描述符、事件内容和时序

与不断变化的混合音频相结合，同时保留模态特定变换。

联合注意力为结合 TASG 分离出的共享信息与事件特定信息提供了计算路径。例如，重复出现的说话人对应的音频帧可以访问相同的声源描述，但每轮对话的措辞和表达方式各自对应不同的事件记录。全局调制提供场景级上下文，而 AbsTime 则提供帧坐标，用于学习每个事件的发生位置。因此，渲染器能够将可复用的身份信息与局部变化结合起来，而不必仅依赖汇总后的场景摘要。

训练采用条件流匹配 [15]，基于配对场景条件和目标音频混合进行。对于干净数据潜变量 $x_0 = z$、高斯噪声 $\epsilon \sim \mathcal{N}(0,\ I)$，以及流时间$t = \text{sigmoid}(u)$，其中 $u \sim \mathcal{N}(0,\ 1)$，我们构造 $x_t = (1-t)x_0 + t\epsilon$并优化

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{x_0,\epsilon,t,c}\Big[\|v_\theta(x_t, t, c) - (\epsilon - x_0)\|^2_{\mathcal{M}}\Big]. \quad (8)$$

平方误差在通道和有效预测帧 $\mathcal{M}$上取平均：保留参考音频时为掩码片段，丢弃参考音频时则为所有有效帧。语义条件丢弃会同时移除文本词元路径和池化文本路径。监督作用于完整混合音频，使梯度能够通过联合注意力将声源/事件要求与其声学实现关联起来。时序在同一生成目标中学习，AbsTime 则使不同场景时长下的位置具有可比性。

b) 异构事件的容量：复杂场景会结合不断变化的语音、瞬态效果，以及缓慢变化的音乐或环境声。我们在后续音频层中使用共享与路由混合专家（MoE）前馈块，同时保留稠密早期层以及跨两个流的联合注意力。对于音频隐藏状态 h，路由计算为

$$\text{MoE}(h) = \gamma_s E_{\text{shared}}(h) + \gamma_r \sum_{j \in \text{TopK}(g(h))} p_j(h) E_j(h), \quad (9)$$

其中，g 是逐词元线性路由器，p_j 是针对所选专家重新归一化后的Softmax权重。可学习的缩放因子从（γ_s，γ_r）=（1，0）开始，使路由容量与共享声学变换一同逐步引入。文本前馈块保持稠密。

标准的辅助负载均衡损失 [39]，仅根据有效音频词元计算，并在各路由层之间求和，以补充流匹配：

$$\mathcal{L} = \mathcal{L}_{\text{FM}} + \lambda_{\text{bal}}\mathcal{L}_{\text{bal}}, \quad \mathcal{L}_{\text{bal}} = \sum_{l \in \mathcal{L}_{\text{MoE}}} N_e \sum_{j=1}^{N_e} \bar{p}_j^{(l)} \bar{\ell}_j^{(l)}, \quad (10)$$

其中，N_e 是每层的路由专家数量，$\bar{p}_j^{(l)}$ 是专家 j的平均路由器概率，$\bar{\ell}_j^{(l)}$ 是被路由至专家 j 的有效音频词元比例（按 K归一化）。

联合注意力整合场景需求，而音频侧混合专家则提供实现其声学组合所需的能力。单个混合区间可能包含语音、持续音乐和瞬态声音效，因此需要在同一潜在序列中进行异构变换。将混合专家置于较后的块中，可以利用已融合声源、事件和时序信息的音频状态进行路由。共享专家提供通用变换，而路由专家则针对这些局部声学模式，提供依赖输入的处理能力。图3展示了完整架构。

D. 推理

模型接受用户提供的TASG或由自然语言编译而成的场景，请求的时长决定潜在画布。我们在由条件流匹配诱导的干净潜变量空间中采用自适应投影引导（APG） [40]。条件速度和文本丢弃速度会被转换为干净估计；二者的差值经过反向动量平滑，并分解为与条件估计平行和正交的分量。平行分量会被衰减，随后将所得更新加到条件估计上，再转换回速度空间。两个分支均保留相同的Ab-sTime坐标和时长。从 $t = 1$处的高斯噪声$x_1 \sim \mathcal{N}(0,\ I)$开始，欧拉积分在64个步骤中推进至 $t = 0$，引导尺度为4.0，最终得到的干净潜变量 x_0 被解码为一个完整的混合音频。

IV. 实验设置

A. 实现细节

我们使用约30,000小时的八百万个内部音频片段训练SonicRender，训练800,000个优化步数，全局批次大小为256。音频由预训练立体声音频自编码器生成的64通道潜变量表示，采样率为44.1kHz，下采样因子为2,048（约21.53个潜在帧/秒）。文本特征通过对两个通义千问隐藏层（索引为 -8和 -28）取平均后进行层归一化预计算；渲染器最多接受1,024个文本词元。MM-DiT包含24个块、1536的隐藏维度和24个注意力头。其最后四个音频块使用一个共享专家和八个采用前二选择的路由专家。由此得到的MoE8渲染器共包含约30亿个参数，每个音频词元包含23亿个激活参数（激活计算量与MoE4相当，而Dense骨干网络为22亿个）。我们采用AdamW，峰值学习率为 3×10^{-5}，使用线性预热与衰减，以及 $\lambda_{\text{bal}} = 0.01$。

B. 模型、基准测试与协议

完整的 SonicRender 模型采用 TASG、两种 AbsTime 路径以及一个 MoE8 MM-DiT 渲染器。我们使用同一检查点，在 Seed-TTS-Eval [41]（英语词错误率和普通话字符错误率）、AudioCaps [42]（FAD、KL 散度和 CLAP）、MusicCaps [43]（KL 散度和 CLAP）以及 MMAG [5] 上进行评估。为评估英语和普通话以外的多语言语音生成能力，我们还在 CV3-Eval [44] 的日语（JA）、韩语（KO）和西班牙语（ES）子集上进行评估。我们采用与 Seed-TTS-Eval 实验相同的检查点和文本条件合成设置，并按照基准测试针对各语言设定的评分单位，报告日语和韩语的字符错误率以及西班牙语的词错误率（%）。

MMAG 包含 3,974 个主集样例和 1,828 个时间戳子集样例。遵循其官方评测协议，我们报告代表性指标：KL 散度、CLAP、词错误率和 AudioBox。

美学制作复杂度（PC，反映多源丰富度），以及时间戳子集上的片段级时间对齐情况（语音 F1 和声音 F1）。由于 MMAG 将事件边界指定到亚秒级小数精度（分辨率为 0.01 秒），而我们的规范TASG表示使用整数秒时钟标记，因此我们在推理时将请求的边界四舍五入到最接近的整数秒，并采用两种评测参照报告片段级 F1：（i）MMAG 原始的亚秒级真实标签窗口（主要黑色分数），用于测试端到端的时间准确性，尽管输入经过了取整；（ii）与取整后的提示相匹配的整数化窗口（灰色分数），用于单独评估模型对精确条件区间的遵循情况。

我们的外部比较涵盖表 I–IV 中列出的任务专用模型和统一音频生成器。在混合场景实验中，每个外部模型都会以其原生支持的输入格式接收同一场景的结构化呈现，其中事件内容、源身份、关系和时序均通过该系统提供的控件表达。这种原生格式协议充分利用了模型可用的条件能力。内部自然语言基线和 TASG 模型共享 MoE8 骨干网络、AbsTime、训练与评估场景信息、精确的话语以及指标时间戳。因此，表示方式的比较改变了场景指令的组织与绑定方式，同时保留了其底层语义内容。所有 SceneMix 比较均使用经过审核的场景规范，且没有任何生成系统能够访问评估音频或隐藏标签。我们在内部客观评估中使用三个生成种子（0、42 和 666），并使用种子 42 生成供人工聆听的样本。

C. SceneMix：组合式多事件场景

为评估组合与时间保真度，我们构建了 SceneMix，这是一个包含 168 个人工编写的多事件音频场景请求的数据集。每个请求指定 2–8 个事件及其时间窗口和声学关系。其中 150 个样本将语音与非语音层结合，涵盖音乐、音效和环境声：40 个样本包含一名说话人，50 个样本包含两名说话人之间的交流，60 个样本涉及至少三名说话人。其余 18 个样本仅包含音乐和音效，不含语音。场景时长介于 6 至 12 秒之间，涵盖重复出现的声源、持续背景声、瞬态效果、重叠和延迟回归。

对于每个生成器，提示渲染大语言模型使用针对模型的指令，将编写好的描述转换为其支持的输入格式：SonicRender 使用 TASG，对比系统则使用相应的结构化提示。这些转换结果保留了所请求的内容、声源关系、时间戳以及逐字语音转录，从而能够比较每个系统如何实现相同的多事件场景。

D. 场景保真度指标

SceneMix 衡量事件完整性、声源绑定和时间定位。我们在固定的逐事件协议下使用 Gemini 3.1 Pro Preview，针对生成的音频评估每个请求的事件，并分别进行内容和时间判定。令 $\mathcal{Q}$ 表示请求的可听事件，而 $d_k = 1$ 当事件 k 被判定为已实现时，否则为零。我们报告

$$\text{Event_Recall} = \frac{\sum_{k\in\mathcal{Q}} d_k}{|\mathcal{Q}|},\ \text{Sound_Recall} = \frac{\sum_{k\in\mathcal{Q}_{\text{ns}}} d_k}{|\mathcal{Q}_{\text{ns}}|}, \tag{11}$$

其中 $\mathcal{Q}_{\text{ns}}$ 包括非语音事件，如音效、音乐和环境声。事件召回率衡量完整合成内容的覆盖程度；声音召回率则单独捕捉其中的非语音层。词错误率采用独立的盲法转录流程计算。说话人准确率（SpkAcc）是指：在请求的语音轮次对中，观测到的说话人相同/不同关系与源标识所示关系一致的比例；未匹配的语音轮次对计为失败。

对于定时目标 $\mathcal{Q}_t$，当事件被判定为与其请求的时间窗口对齐时，$w_k = 1$。判定依据事件的时间范围：短暂事件应发生在允许区域内，而连续背景应在其请求的时间跨度内持续存在。该指标衡量时间指令的完成情况，同时适应语音、音效和背景层的不同持续时长。我们报告

$$\text{WHR} = \frac{\sum_{k\in\mathcal{Q}_t} d_k w_k}{|\mathcal{Q}_t|}, \text{CWA} = \frac{\sum_{k\in\mathcal{Q}_t} d_k w_k}{\sum_{k\in\mathcal{Q}_t} d_k}. \tag{12}$$

WHR 在所有定时目标上联合评估内容和时序；CWA 则评估已完成定时目标中的时序。部分或不确定的内容判定仍视为未成功目标。在相同汇总计数下，二者的关系为 WHR $=$ CWA $\times$ 召回率($\mathcal{Q}_t$)。所有模型均采用相同的评估协议。单独的语音起始分析通过 WhisperX 强制对齐提取词级起始 b_k 时间戳，并检验 $|b_k - b_k| \le \delta$；缺失的起始点计为失败。该分析与 MMAG 片段 F1 一起，为窗口符合度提供更细粒度的时间定位和区间一致性补充。

此外，Gemini 3.1 Pro Preview 会以 1–5 分制评估整体技术质量（AI-Tech）和整体场景指令遵循度（AI- Sem）。

E. 人工评估

我们从 Scene-Mix 中选取具有代表性的 20 个场景，对 SonicRender 和五种对比系统进行主观评估。这些场景涵盖单人至多说话人对话、非语音合成内容，以及包含 2 至 8 项并发或依次发生活动的不同事件负载。15 位听众根据相同的自然语言场景描述，对匿名且随机排序的样本采用两个五分制量表进行评分。MOS-Q 衡量总体音频质量，包括清晰度、自然度和伪影情况；MOS-R 衡量场景实现和相关性，包括请求的内容、事件、声源关系以及前景／背景构成。分数越高越好。当时间顺序和重叠会影响场景含义时，它们也会影响 MOS-R；精确的时间戳一致性则通过时间指标单独评估。听音样本使用种子 42。我们报告平均分及其 95%置信区间，并在两个维度上使用成对显著性检验（Wilcoxon 符号秩检验，$p < 0.01$）比较 SonicRender 与各对比系统。

表 I 公开单任务生成。Seed-TTS-Eval 报告英语词错误率和普通话字符错误率（%）；AudioCaps 使用 FAD、KL 散度和 CLAP；MusicCaps 使用 KL 散度和 CLAP。短横线表示条目不可用。

	Seed-TTS-Eval（文本转语音）		AudioCaps（文本转音频）			MusicCaps（文本转音乐）	
模型	英语词错误率↓	普通话字符错误率↓	FAD↓	KL 散度↓	CLAP↑	KL↓	CLAP↑
特定任务生成器							
F5-TTS [45]	1.8%	1.6%	–	–	–	–	–
CosyVoice 3 [44]	1.5%	0.7%	–	–	–	–	–
Qwen3-TTS [46]	1.2%	0.8%	–	–	–	–	–
Higgs Audio V2 [47]	1.0%	0.8%	–	–	–	–	–
TangoFlux [48]	–	–	**2.26**	**1.15**	0.480	–	–
MusicGen-Large [49]	–	–	–	–	–	1.31	0.280
AudioX [20]	–	–	2.45	1.27	0.440	**0.96**	0.240
AudioLDM 2 [16]	–	–	2.29	1.41	0.419	1.20	0.301
Stable Audio Open [50]	–	–	4.13	2.14	0.350	1.44	0.300
UniFlow-Audio [51]	–	–	5.74	1.43	0.476	1.87	0.241
统一生成器							
Ming-Omni-TTS-0.5B [21]	2.2%	0.9%	–	2.26	0.347	–	–
Ming-Omni-TTS-16.8B [21]	2.0%	0.8%	–	1.64	0.424	–	–
Dasheng AudioGen [2]	12.2%	–	3.19	1.86	0.438	1.37	0.334
UniMoE 音频 [52]	1.3%	0.8%	–	–	–	1.25	0.340
UniAudio 2.0 [17]	3.6%	2.3%	–	3.26	0.170	1.80	–
MiDashengLM-Gen [3]	2.8%	3.9%	5.01	1.91	0.360	1.24	0.330
SonicWeave [22]	1.0%	0.8%	2.75	1.26	0.475	1.06	0.312
我们的模型							
自然语言 +AbsTime	3.9%	3.2%	3.17	1.56	0.432	0.99	0.310
SonicRender	**0.9%**	**0.6%**	2.58	**1.15**	**0.507**	0.97	**0.384**

V. 结果与分析

A. 语音、声音与音乐的统一生成

表 I 显示，使用单个检查点，SonicRender 在标准语音、声音和音乐基准测试中均优于统一基线和若干专用系统。SonicRender 在 Seed-TTS-Eval 上的英语词错误率为 0.9%，普通话字符错误率为 0.6%，在表中列出的统一生成器中数值最低。与之匹配的自然语言模型分别达到 3.9%/3.2%，这表明即使在相对简单的单任务设置中，结构化场景条件也能改善词汇实现。因此，说话人/声源绑定不仅对复杂混合音频有用：明确的内容组织还能为常规语音合成提供更强的条件接口。

在 AudioCaps 上，SonicRender 获得了 0.507 的 CLAP 和 1.15 的 KL 散度，而其自然语言对应模型的数值分别为 0.432 和 1.56。在 MusicCaps 上，SonicRender 取得了整张表中最高的 CLAP（0.384），并在所有统一生成器中实现了最低的 KL 散度（0.97），与专业模型 AudioX 的 0.96 接近。SonicRender 并未以牺牲单一领域的质量为代价来实现多事件控制，而是在三种声学场景中均展现出更优的分布保真度和文本–音频对齐能力。

表 II 将语音评估范围扩展到英语和普通话之外。使用同一个统一检查点，SonicRender 在 CV3-Eval 的日语和韩语子集上分别取得了 5.23% 和 2.27% 的字符错误率，并在西班牙语上取得了 4.78% 的词错误率。这些较低的转写错误率表明，系统能够在这三种语言中准确实现语音内容，从而扩展了统一渲染器的语言覆盖范围。我们的演示页面进一步展示了更多语言、中文方言和地区变体（例如粤语、四川话和东北官话），以及英语口音（例如英国英语和印度英语），其中还包括伴有音乐、环境声和音效的示例。

表 II CV3-Eval 上的多语言语音生成。

模型	日语 字符错误率↓	韩语 字符错误率↓	西班牙语 词错误率↓
Fish Audio S2 [53]	**3.96%**	2.76%	**2.00%**
VoxCPM2 [54]	5.96%	5.69%	3.80%
SonicRender	5.23%	**2.27%**	4.78%

B. MMAG 上带时间戳的混合音频生成

MMAG 提供了一个公开的混合场景测试，可同时评估内容质量和时间戳跟随能力（表 III）。在主集上，SonicRender 的词错误率为 0.02、CLAP 为 0.40、PC 为 4.21，优于匹配的自然语言模型（词错误率为 0.07、CLAP 为 0.32、PC 为 4.05）。在混合音频中，较高的 CLAP 和 PC 分数尤为重要，因为仅有流畅的语音轨道，可能仍无法满足音乐或瞬态要求。总体 KL分数为 0.90，接近 Dasheng AudioGen-Base 的 0.87，这表明指令遵循能力的提升同时保持了具有竞争力的分布匹配表现。

时间戳子集使时间控制主张得以明确。由于我们的训练语料库将 TASG 时钟标记标准化为整数秒精度，我们将 MMAG 的小数

表 III MMAG 主集与时间戳子集生成。推理期间，我们的模型将请求的时间戳四舍五入到整数秒；片段级 F1 分别根据 MMAG 原始的亚秒级真实标签窗口（黑色主要分数）和相应的整数化窗口（灰色分数）进行评估。

	主集：整体字幕				时间戳：定时字幕			
模型	KL↓	WER↓	CLAP↑	PC↑	WER↓	CLAP↑	语音-F1↑	声音-F1↑
AuDirector（智能体式） [26]	1.60	0.08	0.34	3.69	0.15	0.35	0.72	0.57
LTX-2（视听） [55]	1.31	0.04	0.34	3.04	0.05	0.33	0.64	0.44
UniAVGen（视听） [56]	2.28	0.10	0.25	2.35	0.15	0.23	0.63	0.29
Dasheng AudioGen-Base [2]	**0.87**	0.11	0.34	3.55	0.43	0.31	0.60	0.41
Dasheng AudioGen-Fine [2]	0.89	0.11	0.33	3.64	0.13	0.31	0.62	0.42
Ming-Omni-TTS-0.5B [21]	2.80	0.03	0.26	1.60	–	–	–	–
Ming-Omni-TTS-16.8B [21]	2.46	0.03	0.27	1.79	–	–	–	–
自然语言 + AbsTime	0.98	0.07	0.32	4.05	0.08	0.28	0.72 / 0.75	0.50 / 0.54
SonicRender	0.90	**0.02**	**0.40**	**4.21**	**0.03**	**0.39**	**0.79** / 0.86	**0.60** / 0.65

表 IV SceneMix 评测：事件与非语音完整性、内容、时序、说话人绑定以及感知/语义质量。WHR 对每个请求的定时事件进行评分，将遗漏计为失败。平均意见分在 20场景听力子集上评估，并以均值 ± 95%置信区间半宽报告。

	客观评估					AI评审		人工平均意见分	
模型	声音召回率↑	事件召回率↑	词错误率↓	WHR↑	说话人准确率↑	AI-Tech↑	AI-Sem↑	MOS-Q↑	MOS-R↑
Dasheng AudioGen-Fine [2]	56.4%	59.5%	10.2%	40.8%	49.0%	3.10	2.95	$2.74_{\pm 0.32}$	$2.88_{\pm 0.30}$
MiDashengLM-Gen [3]	62.6%	75.5%	8.1%	48.7%	70.8%	3.27	3.20	$3.52_{\pm 0.35}$	$3.69_{\pm 0.34}$
Ming-Omni-TTS-0.5B [21]	37.0%	50.2%	3.2%	35.9%	58.7%	4.14	2.84	$3.55_{\pm 0.32}$	$3.28_{\pm 0.41}$
Ming-Omni-TTS-16.8B [21]	48.4%	53.0%	2.7%	34.6%	65.8%	4.32	2.96	$3.82_{\pm 0.27}$	$3.37_{\pm 0.30}$
自然语言 +AbsTime	59.9%	74.6%	5.2%	68.0%	86.6%	4.29	3.93	$3.90_{\pm 0.23}$	$3.74_{\pm 0.34}$
SonicRender	**71.1%**	**85.4%**	**1.1%**	**83.9%**	**95.7%**	**4.53**	**4.34**	$\mathbf{4.25}_{\pm 0.20}$	$\mathbf{4.44}_{\pm 0.25}$

时间戳在推理时四舍五入到最近的整数秒，以匹配训练令牌分布。值得注意的是，即便存在固有的边界误差（≤ 0.5s），将生成的混合音频直接与基准测试原始的小数秒真实窗口进行评估，仍得到 0.79 的语音 F1 和 0.60 的声音 F1，超过所有对比系统。其带时间戳集合 CLAP 也达到 0.39，优于匹配的自然语言模型的 0.28。这证实 SonicRender 能够生成可识别的异构事件，并尽管存在提示级量化，仍能将它们准确地放置在时间线上。

针对经过整数化量化的真实窗口（灰色条目）评估相同的生成输出，可以进一步区分渲染器对其条件化时间表的内在遵循程度。在这一分辨率匹配的评估下，语音和声音的 F1 分数分别从 0.79/0.60 提升至 0.86/0.65（对于自然语言变体，则从 0.72/0.50 提升至 0.75/0.54）。由于 AbsTime 本身以 21.53 Hz 的精细潜在帧率运行（每帧 ≈ 46ms），黑色分数与灰色分数之间的差距仅反映文本侧令牌约定，而非架构限制；同时，在两项标准下均保持领先的性能，证实了精确的度量时间定位。

C. SceneMix 上的场景保真度

SceneMix 衡量请求是否被呈现为一个完整场景，包括其中不那么突出的声学成分。在表 IV 中，SonicRender 达到 85.4% 的全事件召回率和 71.1% 的声音召回率，同时英语词错误率为 1.1%，说话人准确率为 95.7%。两项召回率之间的差距意义重大：在以对话为中心的场景中，音效、音乐和环境声等非语音成分更容易被遗漏，因此只报告全事件召回率会掩盖场景保真度的一个重要方面。在外部系统中，MiDashengLM-Gen 达到 75.5% 的全事件召回率和 62.6% 的声音召回率；SonicRender 分别提升了 9.9 和 8.5 个百分点。

时间与身份指标进一步印证了相同的结论。SonicRender 的联合窗口命中率（WHR）达到 83.9%，显著高于该列中最强外部参赛系统 MiDashengLM-Gen 的 48.7%。这意味着，被测事件不仅能够被听见，还被置于其允许的时间窗口内。说话人准确率为 95.7%，而 MiDashengLM-Gen 为 70.8%，这反映出在多个局部实现中保持源身份的挑战。SonicRender 的 AI-Tech（4.53）和 AI-Sem（4.34）得分也均为最高，证实严格的事件及时序控制能够直接提升整体感知质量和语义忠实度。

匹配表示比较尤其具有信息价值，因为两个变体接收相同的场景事实，并使用相同的 MoE8骨干网络和 AbsTime。将这些事实组织为 TASG 后，全事件召回率提高了 10.8 个百分点（从 74.6% 提升至 85.4%），声音召回率提高了 11.2 个百分点（从 59.9% 提升至 71.1%）。说话人准确率提高了 9.1 个百分点，从

表V 结构化条件控制、AbsTime 和音频侧容量的消融实验。CWA 评估检测到的事件中的窗口位置；WHR 评估所有定时目标。在每个 MMAG F1 单元格中，黑色/灰色分数分别表示针对原始小数秒真实窗口和整数化真实标注窗口进行评估的结果。

		SceneMix：内容与绑定				时间		MMAG 时间戳	
配置	骨干网络	声音召回率↑	词错误率↓	说话人准确率↑	AI-Sem↑	WHR↑	CWA↑	语音-F1↑	声音-F1↑
TASG（不含 AbsTime）	MoE8	69.2%	1.7%	94.8%	4.25	71.6%	85.0%	0.70 / 0.73	0.48 / 0.53
自然语言 +AbsTime	MoE8	59.9%	5.2%	86.6%	3.93	68.0%	91.2%	0.72 / 0.75	0.50 / 0.54
TASG + AbsTime（完整）	MoE8	**71.1%**	**1.1%**	**95.7%**	**4.34**	**83.9%**	**98.3%**	**0.79 / 0.86**	
TASG + AbsTime	MoE4	69.0%	2.2%	94.3%	4.19	78.9%	96.5%	0.76 / 0.84	0.58 / 0.64
TASG + AbsTime	Dense	60.6%	4.5%	90.6%	4.02	77.9%	96.5%	0.75 / 0.84	0.56 / 0.61

86.6% 提升至 95.7%；WHR 提高了 15.9 个百分点，从 68.0% 提升至 83.9%。英语词错误率从 5.2% 降至 1.1%。这些改进涵盖了事件包含、词汇内容、身份识别和时间定位，而不是集中体现在某个易于优化的单一指标上。这表明，明确的声源—事件分配和局部时间窗口能够为统一渲染器提供更实用的条件结构。

这项听力研究进一步印证了这些客观评估方面的提升。SonicRender 的 MOS-Q 为 4.25 ± 0.20，MOS-R 为 4.44 ± 0.25（95%置信区间），在六个系统中得分最高。与自然语言+AbsTime 变体相比，平均得分分别提升 0.35 和 0.70 分。成对显著性检验显示，在 MOS-Q 和 MOS-R 两项指标上，SonicRender 均显著优于其余五个对比系统。这些结果表明，其组合能力的提升能够转化为可感知的更高音频质量和更忠实的场景实现。

为评估 Gemini 3.1 Pro Preview 作为自动评判器的可靠性，我们在 20场景听力子集上（六个系统共 120 个样本）测量其与人工评估的一致性。在系统层面，AI-Tech 和 AI-Sem 与 MOS-Q 和 MOS-R 的斯皮尔曼相关系数分别均达到 $\rho = 0.94$。在一项由两名人工评分员独立标注的 30 个样本的分层审查中，根据人工共识标注进行测量，Gemini 在事件存在性方面达到 91.0% 的一致率（$\kappa = 0.70$），在时间窗口对齐方面达到 88.2%（$\kappa = 0.63$），在说话人配对一致性方面达到 95.1%（$\kappa = 0.74$）。

D. 消融实验：结构、时间定位与容量

表 V 比较了五种训练配置：完整模型、信息匹配的自然语言变体、不含 AbsTime 的模型，以及 MoE4/稠密模型替代方案。该矩阵分别考察了有组织的场景条件、音频侧时间参照，以及骨干网络层面路由容量的贡献。由于各项比较采用相同的评分场景要求，因此可以综合考察事件覆盖率、内容准确率和位置准确率的变化。

a) 声源感知场景组织：在固定 MoE8 和 AbsTime 的情况下，完整的结构化模型在声音召回率、说话人准确率和 WHR 方面分别比自然语言条件高出 11.2、9.1 和 15.9 个百分点。表 IV 报告的全事件召回率也提高了 10.8 个百分点。这种联合提升十分重要：更高的语音可懂度本身无法解释更好的非语音保留率，而未发生变化的时序机制也无法解释声源绑定方面的全部差异。因此，信息组织方式——为持久身份、局部活动及其时间窗口分别建立记录——是该模型场景控制能力的重要组成部分。

b) 音频侧时间定位：移除 AbsTime 后，WHR 从 83.9% 降至 71.6%（下降 12.3 个百分点），条件窗口准确率从 98.3% 降至 85.0%（下降 13.3 个百分点）；而声音召回率仅变化 1.9 个百分点（71.1% 对 69.2%）。声音召回率相近，表明剩余的表征和骨干网络仍能生成许多请求中的非语音事件。更明显的时间指标下降揭示了音频侧时间定位发挥作用的环节：将可听事件安排在预期时刻。在独立的 MMAG 定时测试集上，移除 AbsTime 后，语音/声音 F1 从 0.79/0.60 降至 0.70/0.48。因此，两项评估都表明，显式音频时间线在事件定位方面具有稳定优势。

c) 主干网络容量：完整的 MoE8 渲染器达到了 71.1% 的声音召回率和 83.9% 的 WHR，相比之下，MoE4 为 69.0%/78.9%，Dense 为 60.6%/77.9%。最明显的容量相关变化体现在声音召回率上：从 Dense 到 MoE8 提升了 10.5 个百分点，同时英语词错误率从 4.5% 降至 1.1%，说话人准确率从 90.6% 提升至 95.7%。以原始亚秒级窗口进行评估时，MMAG F1 也从 0.75/0.56 提升至 0.79/0.60。这些结果表明，路由式音频容量有助于表示异质的局部声学事件，而场景结构和 AbsTime 则提供共享控制。

E. 对不断增加的事件负载的鲁棒性

事件数量分析检验了结构化场景渲染的一项核心预测：随着请求包含更多独立指定的事件，声源/事件绑定应有助于防止遗漏。图4 按请求事件数量对请求进行分组，并评估每组中所有目标的出现情况。因此，它追踪的是生成组合的完整性，而不是其起始时间窗口的正确性。

尽管对于两个内部模型而言，简单的双事件场景在召回率达到 100% 时便趋于饱和，但随着组合密度增加，结构化渲染的优势会急剧扩大。在六个事件时，

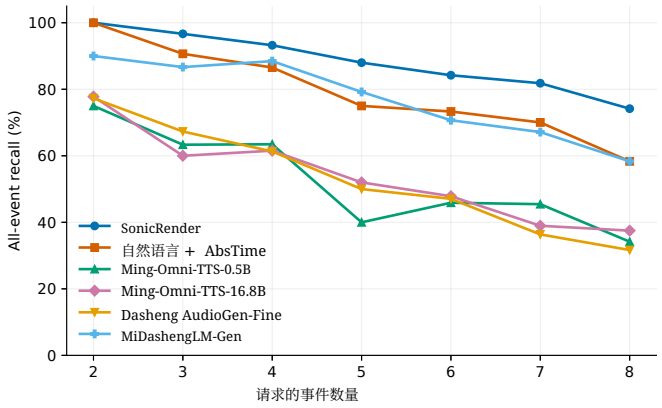

图 4. 事件完整性与组合负载的关系。召回率按每个事件数量组内请求的语音和非语音事件计算，每个事件的权重相同。自然语言 + AbsTime 使用匹配的骨干网络和场景信息。

SonicRender 仍能保留约 84% 的请求事件；在八个事件时，其召回率仍约为 74%，而同一分组中匹配的自然语言基线和 MiDashengLM-Gen 均约为 58%。该曲线还显示出比竞争系统更平稳的下降趋势；一旦多个活动类别必须共存，后者的召回率可能会急剧下滑。这一表现对于包含持久背景层和短时插入事件的场景尤其重要：声学成分的数量会增加，但每个成分仍然是一个独立的请求义务。因此，所测得的优势不仅体现在平均内容覆盖率上；随着请求事件数量增加，这一优势仍然存在。

F. 起始时间对齐与误差容差

前文的 WHR 分析关注定时事件是否出现，以及是否符合请求指定的时间窗口。我们还在 TASG 条件下，进一步考察更局部的问题：语音起始时间误差。图5(a)绘出了请求的起始时间与自动估计的起始时间，虚线对角线表示理想对齐。SonicRender 的观测结果聚集在对角线附近，而不含 AbsTime 的配置则反复出现正偏移，尤其是对于请求在场景较晚时段出现的语音。多个请求位置都能观察到这种差异，因此它比单一的平均绝对误差更能体现这一效果。

图5(b)量化了这些估计满足逐渐放宽的起始时间误差阈值的频率。在0.5秒容差下，完整模型的命中率约为48%，而不使用AbsTime时为19%，相差29个百分点。在1.0秒时，相应比率为85%和55%；在1.5秒时，则为95%和83%。差距最大的是要求准确定位的容差：当事件不能只是在场景中的某处发生时，显式度量坐标最有价值。所有目标起始时间都计入分母，因此遗漏目标不会导致条件命中率虚高。这些自动估计的起始时间为单独定义的 SceneMix WHR和CWA提供了互补的诊断依据。

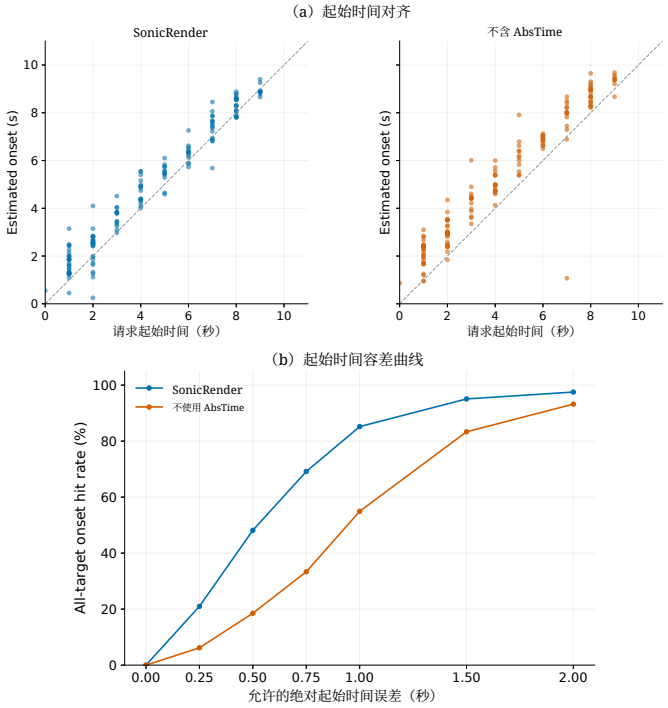

图5. 在匹配的 TASG 输入条件下，AbsTime 对语音起始时间对齐的影响。(a) 已定位语音事件的请求起始时间与估计起始时间；虚线对角线表示精确对齐。(b) 所有目标语音事件中，在各绝对误差容差范围内成功定位的比例。

散点图和容差分析共同有助于解释表V中的落地消融结果。不使用AbsTime时，模型仍会生成许多事件，但往往会把语音安排得过晚。使用AbsTime后，相同的符号调度与音频潜在位置的对应关系更加一致。因此，时间坐标机制不仅提高了事件是否生成的概率，也改善了事件发生的位置。

G. 分层场景的定性分析

图6的面板(a)和(b)展示了两个场景，体现了不同形式的声学构成。在面板(a)中，一段简短话语、持续的音乐层、重复的电子蜂鸣声以及短促的猫叫声，持续时间和作用各不相同。频谱图中可以区分持续音乐与局部高能事件，展现了延展性纹理与特定时刻音效的共存。在面板(b)中，金属敲击声逐渐加快，随后出现重型门闩事件和短促的警报声，而低频贝斯铺底与远处风声则贯穿整个序列。这种有序的瞬态模式体现了超越诸如悬疑氛围这类全局描述的细致事件结构。

这两个对话案例强调在不同对话轮次和竞争层中保持声源身份。在图6(c)中，一名女性说了两次，一名男性播报股票更新，另一名女性插话，重复出现的通知铃声则始终是声景的一部分。可见的有声区域和瞬态声占据混合音频的不同部分，其中也包括重叠要求。图6(d)描绘了一段广播热线节目：主持人在两位来电者之后再次发言，背景中伴有持续弦乐和

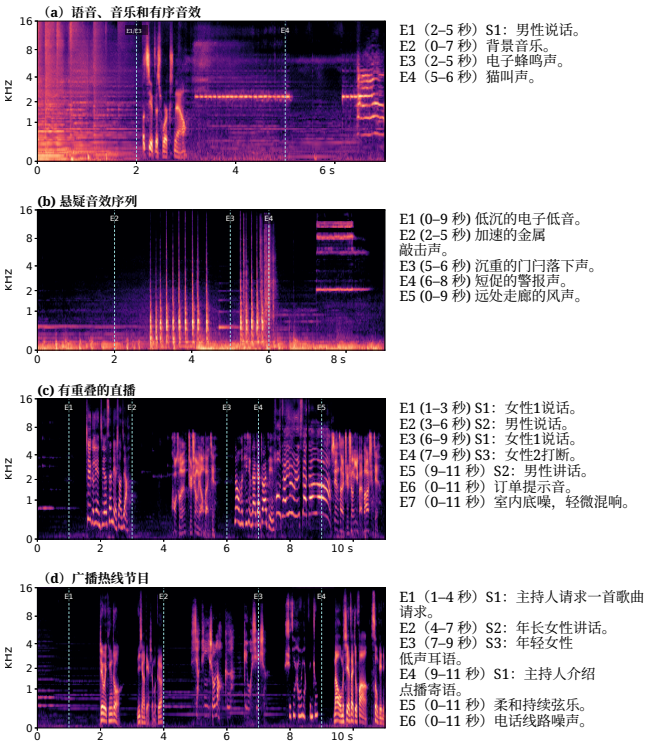

图6。生成的多事件场景：（a）语音、音乐和有序音效；（b）悬疑音效序列；（c）重叠的直播对话；以及（d）广播热线节目。梅尔频谱图与删节输入规范相对应。事件标签和虚线垂直线标识请求的起始点；重复的源标识符表示重复出现的说话人。

电话噪声。稳定的声源标识符将重复的语音活动绑定到同一位指定说话人，即使词汇内容和局部表达方式有所变化也是如此。

这些示例通过呈现复杂多事件生成中所涉及的声学形态，使其可供检查，从而补充了定量证据：谐波背景、瞬态攻击声、重复出现的对话以及持续噪声层。虽然频谱图可视化了频谱能量和时序，但重复出现的保真度则通过说话人准确率和听音协议单独测量。总的来说，这四个场景展示了事件完整性评估和时间定位评估所处理的异质、时间组织内容。

VI. 结论

我们提出了 SonicRender，这是一种统一渲染器，用于生成来源一致且时间锚定的多事件音频。其 TASG 表示以规范格式组织持久身份和局部事件需求，而 AbsTime 则提供音频侧帧坐标和有效时长。联合注意力和全局调制整合这些条件，共享路由音频容量则支持语音、音乐、环境声和音效的异质混合。我们在既有基准测试和自建的 SceneMix 套件上进行评估，结果表明组合保真度有所提升：SonicRender 在 MMAG 上的语音/声音 F1 得分达到 0.79/0.60；与匹配的自然语言变体相比，其在 SceneMix 上的事件召回率和窗口命中率分别提高了 10.8 和 15.9 个百分点，分别。受控消融实验验证了结构化条件控制、时间定位和声学容量的贡献，而人工听测也证实音频质量和场景实现有所改善。这些发现凸显了将组织化场景规范与时间锚定的混合渲染相结合的价值。未来工作将研究亚秒级时间控制，以及并发来源之间明确的物理声学交互。

参考文献

[1] C. Qiang、X. Wang、K. Yin、Y. Liang、Y. Guo、T. Ma、Z. Zhang、T. Wang、C. Gong、Y. Chen、R. Fu、L. Wang 和 J. Dang，“UniSonate：利用文本指令实现语音、音乐和音效生成的统一模型”，载于计算语言学协会年会（*ACL*）论文集，2026年，第28 043–28 054页。[2] J. Mei、H. Dinkel、Y. Niu、X. Sun、G. Li、Y. Liao、J. Zhou、J. Zhang、J. Luan 和 M. Wu，“Dasheng AudioGen：从文本生成连贯音频场景的统一模型”，*arXiv*预印本 *arXiv:2605 27838*，2026。[3] X. Sun、H. Dinkel、G. Li、J. Mei、Y. Niu、Z. Han、Y. Jiang、J. Zhou、L. Fan 和 J. Luan，“MiDashengLM-Gen：通过大语言模型驱动的自回归流匹配实现统一音频场景生成”，*arXiv*预印本 *arXiv:2608.11804*，2026。[4] J. Dai、X. Duan、X. Fan、Y. Feng、J. Li、X. Li、Y. Li、L. Min、Y. Shi、X. Song、Y. Wang、C. Wen、M. Wu、B. Xiang、H. Zhang、H. Zhao 和 R. Zheng，“Qwen-Audio-3.0-Gen-Preview技术报告”，*arXiv*预印本 *arXiv:2607.27011*，2026。[5] Z. Zheng、X. Xu、J. Mei、Y. Li、M. Lv、W. Wu、C. Zhang 和 M. Wu，“MMAG：多控制混合音频生成基准测试”，*arXiv*预印本 *arXiv:2608.06900*，2026。[6] J. Huang、Y. Ren、R. Huang、D. Yang、Z. Ye、C. Zhang、J. Liu、X. Yin、Z. Ma 和 Z. Zhao，“Make-An-Audio 2：时间增强的文本到音频生成”，*arXiv*预印本 *arXiv:2305.18474*，2023。[7] Y. Liu、C. Li、W. Zhang、W. Wang、M. Yu、R. Fu、Z. Lin、W. Wang 和 D. Yu，“DegDiT：利用动态事件图引导的扩散变换器实现可控音频生成”，*IEEE/ACM*音频、语音与语言处理汇刊，第34卷，第1300–1311页，2026。[8] Y. Jiang、Z. Chen、Z. Ju、Y. Dai、W. Dou 和 J. Zhu，“ControlAudio：通过渐进式扩散建模应对文本引导、时间指示和可理解的音频生成”，载于计算语言学协会年会（*ACL*）论文集，2026年，第1394–1413页。[9] Z. Xie、X. Xu、Z. Wu 和 M. Wu，“PicoAudio：在文本到音频生成中实现音频事件的精确时间戳和频率可控性”，*arXiv*预印本 *arXiv:2407.02869*，2024。[10] Z. Zheng、Z. Xie、X. Xu、W. Wu、C. Zhang 和 M. Wu，“PicoAudio2：利用自然语言描述实现时间可控的文本到音频生成”，*arXiv*预印本 *arXiv:2509.00683*，2025。[11] Y. Jiang、Z. Chen、Z. Ju、C. Li、W. Dou 和 J. Zhu，“FreeAudio：无需训练的时序规划，用于可控的长篇文本到音频生成”，载于 *ACM*多媒体国际会议（*ACM MM*）论文集，2025。[12] H. Zhang、J. Zhang、C. Zhang、S. Wang 和 W. Yang，“TAG：通过大语言模型引导的手动描述和控制实现结构化时间音频生成”，载于 *IEEE*声学、语音与信号处理国际会议（*ICASSP*）论文集，2026。[13] Y. Liu、Q. Zhang、H. Wang、D. Zhou、J. Zhang、X. Li、Y. Shi、J. Liu、R. Huang、Y. Zou 等，“缺失的时间联系：用于脚本驱动音视频生成的时间上下文路由”，*arXiv*预印本 *arXiv:2609.02367*，2026。[14] P. Esser、S. Kulal、A. Blattmann、R. Entezari、J. M¨uller、H. Saini、Y. Levi、D. Lorenz、A. Sauer、F. Boesel、D. Podell、T. Dockhorn、Z. English 和 R. Rombach，“扩展用于高分辨率图像合成的整流流变换器”，载于 国际机器学习会议（*ICML*）论文集，2024年，第12 606–12 633页。[15] Y. Lipman、R. T. Q. Chen、H. Ben-Hamu、M. Nickel 和 M. Le，“用于生成建模的流匹配”，载于国际学习表征会议（*ICLR*）论文集，2023年。[16] H. Liu、Y. Yuan、X. Liu、X. Mei、Q. Kong、Q. Tian、Y. Wang、W. Wang、Y. Wang 和 M. D. Plumbley，“AudioLDM 2：通过自监督预训练学习整体音频生成”，*IEEE/ACM*音频、语音与

*udio, Speech, and L*语言 *P*过程 *i*ng, vo l. 32 , pp. 2871 –2883, 2024年。

[17] D. Yang、Y. Wang、D. Chong、S. Liu、X. Wu 和 H. Meng，《UniAudio 2.0：具有文本对齐的因式分解音频标记化的统一音频语言模型》，arXiv 预印本 arXiv:2602.04683，2026 年。[18] H. Liu、K. Luo、W. Wang、Q. Chen、B. Ma、X. Li 和 W. Xue，《AudioCALM：面向通用音频生成的连续自回归语言建模》，arXiv 预印本 arXiv:2606.23080，2026 年。[19] B. Lin、B. Zhao、B. Wang 等，《StepAudio 3 Gen 技术报告》，arXiv 预印本 arXiv:2609.12945，2026 年。[20] Z. Tian、Z. Liu、Y. Jin、R. Yuan、L. Xue、X. Tan、Q. Chen、W. Xue 和 Y. Guo，《AudioX：任意内容到音频生成的统一框架》，载于《国际学习表征会议》（ICLR）论文集，2026 年。[21] Inclusion AI，《Ming-Omni-TTS：通过精确控制实现语音、音乐和声音的简单高效统一生成》，项目报告与模型发布，2026 年。[在线]。可访问：https://github.com/ inclusionAI/Ming-omni-tts [22] Y. Cai、X. Li、Y. Zhou、J. Li、D. Wang、D. Yang、X. Wu、C. Zhang、Z. Wu、P. Wan 和 H. Meng，《SonicWeave：用于统一音频场景生成的分块路由混合专家》，arXiv 预印本 arXiv:2608 09571，2026 年。.[23] Z. Peng、J. Yu、W. Wang、Y. Chang、Y. Sun、L. Dong、Y. Zhu、W. Xu、H. Bao、Z. Wang、S. Huang、Y. Xia 和 F. Wei，《VibeVoice：采用下一标记扩散的表现力播客生成》，载于《国际学习表征会议》（ICLR）论文集，2026 年。[24] M. Finkelson、D. Segal、E. Richardson、S. Armon、N. Goldring、P. Panet、N. Zabari、B. Brazowski、O. Patashnik 和 Y. HaCohen，《基于真实环境先验的参考驱动多说话人音频场景生成》，arXiv 预印本 arXiv:2606.19325，2026 年。[25] X. Liu、Z. Zhu、H. Liu、Y. Yuan、Q. Huang、M. Cui、J. Liang、Y. Cao、Q. Kong、M. D. Plumbley 和 W. Wang，《WavJourney：利用大语言模型进行组合式音频创作》，《IEEE/ACM音频、语音与语言处理汇刊》，第 33 卷，2844，2025 年 2830 页。–。[26] Y. Ren、X. Xu、Z. Zhang、W. Wu、B. Li 和 C. Zhang，《AuDirector：用于沉浸式音频叙事的自反思闭环框架》，arXiv 预印本 arXiv:2605.11866，2026 年。[27] Y. Duan、Q. Xu、H. Wu、Z. Liu、W. Guan、J. Liu、Z. Ma、K. Xu 和 X. Chen，《Audio-Oscar：用于复杂音频场景生成、编排与优化的多智能体系统》，arXiv 预印本 arXiv:2606.07397，2026 年。[28] Y. Wang、H. Chen、D. Yang、Z. Wu 和 X. Wu，《AudioComposer：利用自然语言描述实现细粒度音频生成》，载于《IEEE声学、语音与信号处理国际会议》（ICASSP）论文集，2025 年。[29] Z. Evans、C. Carr、J. Taylor、S. H. Hawley 和 J. Pons，《快速时间条件潜空间音频扩散》，载于《国际机器学习会议》（ICML）论文集，2024 年，第 12 652–12 665 页。[30] Z. Guo、J. Mao、R. Tao、L. Yan、K. Ouchi、H. Liu 和 X. Wang，《使用多个条件扩散模型的音频生成》，载于《AAAI人工智能会议》论文集，第 38 卷第 16 期，2024 年，第 18 153–18 161 页。[31] T. Du、J. Chen、J. Lu、Q. Xu、H. Liao、Y. Chen 和 Z. Wu，《无需训练的时间引导扩散可控文本到音频生成》，载于《IEEE多媒体与博览会国际会议》（ICME）论文集，2024 年。[32] H. Zhu、Y. Xiao、X. Li、Z. Ma、J. Yu、B. Zhang、M. Yang 和 X. Chen，《Audio ControlNet：用于细粒度音频生成与编辑》，arXiv 预印本 arXiv:2602.04680，2026 年。[33] Z. Xie、X. Xu、Z. Wu 和 M. Wu，《AudioTime：时间对齐的音频文本基准数据集》，arXiv 预印本 arXiv:2407.02857，2024 年。[34] Z. Wang、K. Lei、C. Zhu、J. Huang、S. Zhou、L. Liu、X. Cheng、S. Ji、Z. Ye、T. Jin 和 Z. Zhao，《T2A-Feedback：通过细粒度人工智能反馈提升文本到音频生成的基础能力》，载于《计算语言学协会年会》（ACL）论文集，2025 年，第 23 535–23 547 页。[35] C.-Y. Kuan、S. Kim、B. Kim、S. Kim、B.-R. Lu、Q. Tang、A. Gandhe、H.- Y. Lee、C.-C. Kao 和 C. Wang，《通过音频感知大语言模型的细粒度反馈改进文本到音频指令遵循》，载于 Interspeech 论文集，2026 年。[36] Y. Rong、F. Ma、X. Li、J. Wang、C. Zhang 和 L. Liu，《AudioMap：面向时间感知密集音频描述的完形填空与选择强化学习》，arXiv 预印本 arXiv:2608.09559，2026 年。[37] S. Bai、Y. Cai、R. Chen 等，《Qwen3-VL 技术报告》，arXiv 预印本 arXiv:2511.21631，2025 年。[38] P. Wang、S. Bai、S. Tan、S. Wang、Z. Fan、J. Bai、K. Chen、X. Liu、J. Wang、W. Ge、Y. Fan、K. Dang、M. Du、X. Ren、R. Men、D. Liu、C. Zhou、J. Zhou 和 J. Lin，《Qwen2-VL：增强视觉语言模型在任意分辨率下对世界的感知”，arXiv预印本 arXiv:2409.12191，2024年。[39] W. Fedus、B. Zoph 和 N. Shazeer，《Switch Transformers：借助简单高效的稀疏性扩展至万亿参数模型》，《机器学习研究杂志》，第23卷，第120期，第1–39页，2022年。[40] S. Sadat、O. Hilliges 和 R. M. Weber，《消除扩散模型中高引导尺度导致的过饱和与伪影》，载于《国际学习表征会议（ICLR）论文集》，2025年。[41] P. Anastassiou、J. Chen、J. Chen、Y. Chen、Z. Chen、Z. Chen、J. Cong、L. Deng、C. Ding、L. Gao 等，《Seed-TTS：高质量、多用途语音生成模型家族》，arXiv预印本 arXiv:2406 02430，2024年。。[42] C. D. Kim、B. Kim、H. Lee 和 G. Kim，《AudioCaps：为真实场景中的音频生成描述》，载于《北美计算语言学协会分会会议：人类语言技术（NAACL-HLT）论文集》，2019年，第119–132页。[43] A. Agostinelli、T. I. Denk、Z. Borsos、J. Engel、M. Verzetti、A. Caillon、Q. Huang、A. Jansen、A. Roberts、M. Tagliasacchi 等，《MusicLM：从文本生成音乐》，arXiv预印本 arXiv:2301.11325，2023年。[44] Z. Du、C. Gao、Y. Wang、F. Yu、T. Zhao 等，《CosyVoice 3：通过扩展规模与后训练实现真实场景语音生成》，arXiv预印本 arXiv:2505.17589，2025年。[45] Y. Chen、Z. Niu、Z. Ma、K. Deng、C. Wang、J. Zhao、K. Yu 和 X. Chen，《F5-TTS：借助流匹配伪装成流畅且忠实语音的童话讲述者》，arXiv预印本 arXiv:2410.06885，2024年。[46] H. Hu、X. Zhu、T. He、D. Guo、B. Zhang、X. Wang、Z. Guo、Z. Jiang、H. Hao、Z. Guo、X. Zhang、P. Zhang、B. Yang、J. Xu、J. Zhou 和 J. Lin，《Qwen3-TTS技术报告》，arXiv预印本 arXiv:2601.15621，2026年。[47] Boson AI，《Higgs Audio V2：重新定义音频生成中的表现力》，《技术报告与模型发布》，2025年。 [在线]。可用地址：https://github.com/boson-ai/higgs-audio [48] C.-Y. Hung、N. Majumder、Z. Kong、A. Mehrish、A. Zadeh、C. Li、R. Valle、B. Catanzaro 和 S. Poria，《TangoFlux：借助流匹配和CLAP排序偏好优化实现超快速且忠实的文本到音频生成》，载于《国际学习表征会议（ICLR）论文集》，2026年。[49] J. Copet、F. Kreuk、I. Gat、T. Remez、D. Kant、G. Synnaeve、Y. Adi 和 A. D´efossez，《简单且可控的音乐生成》，载于《神经信息处理系统进展（NeurIPS）》，第36卷，2023年，第47 704–47 720页。[50] Z. Evans、J. D. Parker、C. Carr、Z. Zukowski、J. Taylor 和 J. Pons，《Stable Audio Open》，arXiv预印本 arXiv:2407.14358，2024年。[51] X. Xu、J. Mei、Z. Zheng、Y. Tao、Z. Xie、Y. Zhang、H. Liu、Y. Wu、M. Yan、W. Wu、C. Zhang 和 M. Wu，《UniFlow-Audio：从全模态生成音频的统一流匹配》，arXiv预印本 arXiv:2509.24391，2025 年。[52] Z. Liu 、 Y. Li 、 X. Zhang 、 Q. Teng 、 S. Jiang、X. Chen、H. Shi、H. Chen、F. Meng、M. Zhao 等，《UniMoE-Audio：借助动态容量混合专家实现统一语音与音乐生成》，载于《计算语言学协会年会（ACL）论文集》，2026年，第9107–9119页。[53] S. Liao、Y. Wang、S. Liu、Y. Cheng、R. Zhang、T. Li、S. Li、Y. Zheng、X. Liu、Q. Wang、Z. Zhou、J. Liu、X. Chen 和 D. Han，《Fish Audio S2技术报告》，arXiv预印本 arXiv:2603.08823，2026年。[54] Y. Zhou、G. Zeng、X. Liu、X. Li、R. Yu、J. Gui、J. Wu、Z. Wang、X. Shen、R. Ye、Z. Zhang、J. Zhou、B. Bai、W. Sun、M. Deng、Q. S hi、Z. Wu 和 Z. Liu，《VoxCPM2 06928，2026 2606技术报告》，arXiv 预印本 arXiv: 。[55] Y. HaCohen、B. Brazowski、N. Chiprut、Y. Bitterman、A. Kvochko 等，《LTX-2：高效的联合视听基础模型》，arXiv预印本 arXiv:2601.03233，2026年。[56] G. Zhang、Z. Zhou、T. Hu、Z. Peng、Y. Zhang、Y. Chen、Y. Zhou、Q. Lu 和 L. Wang，《UniAVGen：借助非对称跨模态交互实现统一音频与视频生成》，载于《IEEE/CVF计算机视觉与模式识别会议（CVPR）论文集》，2026年。