



上海交通大学硕士学位论文

基于面部捕捉、语音、头部运动的在线实时数 字人驱动

姓 名：花泉 润
导 师：杨旭波教授
学 号：122037990002
申 请 学 位：工学硕士
学 科 / 专 业：专业
院 系：计算机学院

2026 年 1 月 7 日

**A Dissertation Submitted to
Shanghai Jiao Tong University for the Degree of Master**

**FACECAPGES: REAL-TIME FRAME-BY-FRAME
GESTURE GENERATION FROM AUDIO, FACIAL
CAPTURE, AND HEAD POSE**

Author: Jun Hanaizumi

Supervisor: Prof. Xubo Yang

Depart of XXX

Shanghai Jiao Tong University

Shanghai, P.R. China

January 7th, 2026

上海交通大学

学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全知晓本声明的法律后果由本人承担。

学位论文作者签名：

日期： 年 月 日

上海交通大学

学位论文使用授权书

本人同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。

本学位论文属于：

- 公开论文
 - 内部论文，保密 1年 / 2年 / 3年，过保密期后适用本授权书。
 - 秘密论文，保密 ____ 年（不超过 10 年），过保密期后适用本授权书。
 - 机密论文，保密 ____ 年（不超过 20 年），过保密期后适用本授权书。
- （请在以上方框内选择打“√”）

学位论文作者签名：

指导教师签名：

日期： 年 月 日 日期： 年 月 日

摘要

手势是交流中重要的细节补充与情感表达载体。面向虚拟人/数字形象的肢体控制中，传统方案往往依赖穿戴式检测设备或其他专用传感器，以稳定获取人体运动信息，但这类设备带来较高的使用门槛与部署成本。

随着图像识别技术的发展，计算机能够通过相机识别用户动作并实现无穿戴的动捕式交互，在一定程度上降低了设备负担。然而，这类方案仍要求用户在镜头前实际表演手势动作；在远程交流或沉浸式场景中，手势作为情感表达工具往往伴随空间占用与体力消耗，人们需要投入较多精力活动肢体以增强说服力或表达效果，从而使虚拟世界中的手势表达依然显得繁琐。

近年来，手势与语言的关系被进一步分析，使得计算机能够从文本或语音生成与语义一致的手势动画，让数字人物在发言中融入更符合人类交流习惯的非语言表达。若将该技术用于用户的实时数字形象，则有望提供更轻便的手势控制方案：用户无需实际做出手势，只需在相机前说话即可，同时自然地产生语音、面部表情与头部旋转等信息。面部表情可补充情感线索，头部旋转可为手势朝向与叙事空间锚定提供指导，从而改进依赖空间方位与路径方向的手势推理；且这些信息的实时获取技术已相对成熟，具备良好的在线性能。

然而，当前协同语音手势生成技术通常以获取完整文本为前提，通过语义分析生成高精度手势，难以直接适配逐字输入的用户语音实时场景。因此，本论文提出 FaceCapGes 方法，基于语音、面部表情与头部旋转三种实时信息生成在线实时的 3D 手势骨骼动画。该方法可为用户的实时数字形象添加手势动画，无需用户实际做出手势动作；同时模型不依赖未来输入，能够在实时环境中提升虚拟形象的表达能力。

以往研究已对面部表情、语音与手势的关系进行分析，并提出成熟的学习方法。本模型在现有级联架构基础上，增加头部姿态特征分析模块，同时引入滑动窗口机制以实现架构的实时运行。本模型所依赖的框架及额外添加模块性能开销低，与面部捕捉和头部姿态计算任务同时运行时，仍具备良好的实时性能。

主观评价结果表明，在自然性方面，本方法与当前主流方法相当；在响应速度上，具有显著优势。生成的手势与语音高度对齐，且具备良好的实时交互表现。此外，本模型可部署在 iPhone 等轻量设备上，只要输入格式兼容 ARKit 面部捕捉标准，就能广泛应用于各类实时互动场景。

关键词: 协同语音手势生成, 数字人驱动, 面部捕捉, 多模态学习

Abstract

Gestures are a crucial channel for enriching communication with complementary details and emotional expression. In the control of virtual humans or digital avatars, traditional approaches often rely on wearable sensing devices or other dedicated sensors to reliably capture human motion; however, such hardware introduces substantial barriers to use and deployment costs.

With advances in visual recognition, camera-based systems can capture user movements and enable markerless, mocap-style interaction, partially reducing the dependence on wearables. Nevertheless, these approaches still require users to physically perform gestures in front of the camera. In remote communication or immersive scenarios, expressive gesturing often entails non-trivial spatial and physical effort: users must invest energy in moving their limbs to strengthen persuasion or convey emotions, which can make gestural expression in virtual worlds cumbersome.

Meanwhile, growing understanding of the relationship between gestures and language has enabled computers to generate semantically consistent gestures from text or speech, allowing digital characters to incorporate nonverbal behaviors that better match human conversational habits. If applied to a user's real-time avatar, such techniques could provide a more lightweight control paradigm: instead of performing gestures, users only need to speak in front of a camera, naturally producing speech, facial expressions, and head rotations. Facial expressions can supplement affective cues, while head rotations can guide gesture orientation and narrative spatial anchoring, improving the inference of gestures that depend on spatial references and motion directions. Moreover, real-time capture of facial expressions and head pose is relatively mature and offers strong online performance.

However, most existing co-speech gesture generation methods assume access to complete text and rely on semantic analysis to generate high-quality gestures, making them difficult to adapt to real-time scenarios with incremental, word-by-word user speech. To address this, we propose FaceCapGes, a method that generates online, real-time 3D skeletal gesture animations from three streams of real-time signals: speech, facial expressions, and head rotations. FaceCapGes can augment a user's live digital avatar with expressive gestures without requir-

ing the user to physically gesture, and it does not depend on future inputs, thereby enhancing avatar expressiveness in live interactive settings.

Prior work has analyzed the relationships among facial expressions, speech, and gestures, and has established effective learning paradigms. Building on existing cascaded architectures, our model introduces an additional head-pose feature analysis module and incorporates a sliding-window mechanism to enable real-time operation. The underlying framework and added modules incur low computational overhead, and the system maintains robust real-time performance even when running concurrently with facial capture and head-pose estimation.

Subjective evaluations show that our method achieves naturalness comparable to current mainstream approaches, while providing a significant advantage in responsiveness. The generated gestures are tightly aligned with speech and demonstrate strong real-time interactive behavior. Furthermore, our model can be deployed on lightweight devices such as iPhones; as long as the input format is compatible with the ARKit facial capture standard, it can be broadly applied to a wide range of real-time interactive scenarios.

Key words: co-speech gesture generation, virtual avatar driving, face-capture, multimodal learning

目 录

第1章 引言	1
1.1 研究背景和意义.....	1
1.2 研究内容.....	2
1.3 手势的定义与身体姿态的参数化表示.....	3
1.3.1 手势的定义与范围	3
1.3.2 身体姿态的参数化表示	5
1.4 面部表情的定义与参数化表示.....	7
1.5 实时生成的理论基础与可行性分析.....	9
1.6 国内外研究现状.....	10
1.6.1 规则驱动阶段	10
1.6.2 数据驱动阶段	11
1.6.3 多模态扩展阶段	11
1.6.4 总结	11
1.7 研究目标的差异.....	12
1.8 任务约束与可利用条件的差异.....	13
1.9 手势生成评估方法.....	13
1.10 本文研究目标	14
1.11 本章小结	15
1.12 论文组织架构	15
第2章 基于语音、面部、头部姿态的在线实时手势生成系统	17
2.1 研究定位与总体设计思路.....	17
2.2 系统整体框架与模块定位.....	17
2.2.1 信号采集与系统配置层	18
2.2.2 手势生成模型层 (FaceCapGes)	19
2.2.3 渲染驱动层	19

第 3 章 头部姿态在多模态级联架构中的引入	21
3.1 问题定义.....	21
3.1.1 任务描述	21
3.1.2 输入与输出模态	21
3.2 级联架构设计的继承.....	22
3.2.1 级联架构的原理与理论背景	22
3.2.2 说话人 ID 分支移除.....	22
3.2.3 输入模态继承	23
3.2.4 输出模态继承（身体姿态解码）	23
3.3 头部姿态模态的引入.....	25
3.3.1 输入特征与表示	25
3.3.2 特征获取方法	25
3.3.3 级联结构中的位置	25
3.4 模型整体结构.....	26
3.5 本章小结.....	26
第 4 章 基于滑动窗口的实时手势生成自回归训练	28
4.1 单步因果 LSTM 预测器	28
4.1.1 LSTM 基本概念与状态传递机制.....	28
4.1.2 选择单向 LSTM 的因果性动机.....	29
4.1.3 双通路记忆：循环状态与显式历史上下文	30
4.1.4 窗口内预测过程的形式化表达	31
4.2 训练片段构造：基于 CaMN 的固定长度切割策略.....	32
4.2.1 固定长度片段定义	32
4.2.2 重叠切割与样本覆盖	33
4.3 片段内部的滑动窗口展开策略.....	33
4.3.1 前置动作帧与 Warm-up 阶段	33
4.3.2 滑动窗口展开与逐帧自回归生成	34
4.3.3 拼接生成序列与片段级输出	35
4.4 片段级训练目标：监督损失与时序平滑约束.....	35
4.4.1 片段级生成序列与损失计算范围	35
4.4.2 总体优化目标	36

4.4.3 姿态重构与时序平滑损失	36
4.4.4 对抗损失	36
4.4.5 损失权重设置	37
4.5 对抗训练：序列级判别器与前置动作掩码.....	37
4.5.1 判别器输入：基于拼接片段的序列级判别	37
4.5.2 基于 BCE 的对抗损失定义	38
4.5.3 交替优化策略	38
4.6 本章小结.....	38
第 5 章 实验结果与分析	40
5.1 训练配置.....	40
5.2 实验配置.....	41
5.2.1 实验对比模型	41
5.2.2 跨模型评估设置	41
5.3 用户评估.....	42
5.3.1 用户评估系统与实验配置	42
5.3.2 结果与分析	45
5.4 定性分析.....	47
5.4.1 生成动作平滑性	47
5.4.2 头部朝向与手势空间指向一致性	48
5.5 客观评估指标与实现细节.....	50
5.5.1 Fréchet Gesture Distance (FGD)	50
5.5.2 Semantic Relevance Gesture Recall (SRGR)	52
5.5.3 Beat Alignment (BA)	52
5.5.4 L1 范数	53
5.5.5 评估区域设定与公平性说明	54
5.6 定量评估结果.....	54
5.7 消融实验分析.....	55
5.8 性能评估.....	56
5.8.1 单帧推理性能	56
5.8.2 端到端计算链路延迟	57
5.8.3 系统更新率	58

5.8.4 帧率设定的可扩展性	58
5.9 本章小结.....	59
第 6 章 结论	60
6.1 本文工作总结.....	60
6.2 未来工作展望.....	60
6.2.1 高层语义信息	60
6.2.2 面向未来趋势的预测性训练目标	60
6.3 本章小结.....	61
参考文献.....	62
附录 A 手势生成对比视频	67
附录 B 代码与实现资源	68
致 谢.....	69
学术论文和科研成果目录.....	70

插 图

图 1.1 FACS 中闭眼动作单元的定义	8
图 1.2 BlendShape 在闭眼形变上的线性插值效果	8
图 1.3 MediaPipe Face Mesh 关键点结构示意图	9
图 2.1 系统整体架构与数据流示意图	18
图 3.1 头部姿态编码器结构示意图	26
图 3.2 FaceCapGes 模型整体结构	26
图 5.1 BEAT 数据集的骨架拓扑结构与驱动范围	40
图 5.2 用户评估工具实机界面	44
图 5.3 用户评估总体主观排名结果	46
图 5.4 平衡拉丁方设置下的主观排名结果	46
图 5.5 生成动作效果对比	48
图 5.6 头部朝向与手势指向一致性的动作对比	49

*

表 格

表 1.1 不同旋转表示方式的空间连续性与使用示例.....	7
表 1.2 离线与在线手势生成任务在约束与可利用信息上的对比.....	13
表 3.1 输入输出模态符号与维度.....	22
表 5.1 对比模型的输入输出模态.....	42
表 5.2 定量评估结果.....	55
表 5.3 消融实验结果.....	55
表 5.4 推理速度评估结果.....	57

*

第1章 引言

1.1 研究背景和意义

近年来，随着元宇宙、虚拟社交与直播等领域的相关技术日趋成熟，用户已能够使用任意外观的虚拟人作为交互载体，在虚拟空间中与异地用户进行交流。虚拟人3D模型的姿态由其内部骨架关节的旋转参数（如欧拉角、四元数等）定义，最终通过蒙皮渲染技术完成可视化。得益于自动骨骼绑定技术，骨骼动画的生成可消除不同3D模型间的骨架拓扑差异，实现跨模型复用。

在虚拟人交互中，穿戴式动作捕捉设备是实时驱动手势的传统方案，贴合肢体的标记点可将肢体运动实时转换为相同的骨骼动画，提供直观、准确的操控。尽管其精度较高，但对于大多数用户而言，此类设备存在功能用途单一、硬件成本昂贵、便携性差等问题，限制了使用频率。因此，在当前的虚拟人交互应用中，仅少数专业用户会使用此类设备，而多数用户在设备限制下无法简单控制虚拟人肢体，导致了两种用户体验之间的不一致。

针对普通用户对低门槛虚拟人交互的需求，基于相机的动作捕捉技术^[1-2]成为主流替代方案。该技术无需额外硬件，仅通过手机或电脑的内置相机即可实时捕捉用户动作，用于转化为骨骼动画。然而，该方案仍存在三种局限：一是这种方法要求用户在空中面向相机做出手势，过程中难以同步操作键盘、鼠标，造成操作冲突；二是手势活动范围受相机视野限制，在自然使用距离通过手机或电脑的内置相机拍摄用户，将严重限制手势的捕捉范围；三是持续的手势动作会产生体力消耗，在直播等长时间使用场景中，用户疲劳问题将变得显著。”

因此，我们提出一种新的需求：一种无需用户实际做出手势，仅通过用户的实时语音、面部捕捉与头部姿态，实时生成与语义和情感相匹配的手势骨骼动画。该方法旨在降低使用门槛，并解决操作冲突与体力消耗问题。

然而，现有研究尚未提供成熟的解决方案满足上述需求。首先，当前多数手势生成方法依赖完整的语音或文本输入。由于用户的语音输入是逐字进行的，计算机需要等待用户的未来输入才能解析当前的语义，造成动画生成的延迟。其次，现有研究未对用户的头部姿态模态做深入研究。头部姿态对手势的节奏与朝向具有明确的关系，且可以通过常规相机实时捕捉，但利用该模态来增强生成手势的自然度的相关研究尚不充分。

为此，本文提出了一种新颖的实时手势生成模型。该模型以帧为单位，输入语音、面部表情与头部姿态数据，并逐帧输出对应的骨骼动画。本文首次在实时手势生成中，将头部姿态作为一种新的模态引入，利用其与自然手势节奏与朝向的高度相关性，结合语音和面部信息共同提升生成动作的自然度。我们采用级联多模态架构与自回归训练来融合这些模态，以学习其联合表征，从而在严格的实时约束下增强手势的表现力。

本文的主要贡献如下：

1. 提出了 FaceCapGes，一种帧级实时手势生成模型，使用户无需动作捕捉设备或实际做出手势，仅通过语音等常见输入即可驱动虚拟人的手勢动画；
2. 将头部姿态作为新模态引入多模态级联架构，在实时生成约束下有效提升了手势的自然性与表现力；
3. 通过实验结果表明，该模型在手势自然性、语音-手势对齐度与实时响应方面展示了良好的性能。其框架适用于所有兼容 ARKit 的设备。

1.2 研究内容

本文的研究目标是构建一种基于语音、面部捕捉与头部姿态的实时数字人手势生成模型，实现无需动作捕捉设备即可驱动虚拟角色自然表达的实时动画系统。该研究旨在解决现有手势生成方法对未来上下文的依赖及实时性不足的问题，从而为虚拟人交互提供更低门槛、更高沉浸度的解决方案。

为实现上述目标，本文的主要研究内容如下：

其一，设计一种帧级手势生成架构。为实现帧级实时推理，本架构在基线模型^[3]的基础上做两种调整：一方面，保留其语音、面部表情等可实时获取的输入模态，移除非实时模态的输入分支与特征处理模块；另一方面，采用滑动窗口式自回归训练方式，确保模型仅依赖历史与当前帧信息进行推理（不依赖未来上下文），同时通过窗口内时序依赖建模保持动作的时间连续性。由此使架构可处理逐帧输入的实时流数据。

其二，提出一种引入头部姿态的新型模态融合策略。在现有语音、面部表情生成手势的模型基础上，将头部姿态作为终端模态引入，设计头部姿态编码器并提取特征，以增强生成手势的朝向的自然性。

其三，搭建实验系统与用户测试环境。本文基于主流渲染引擎构建了虚拟人驱动与手勢动画的可视化系统。该系统作为评估平台，为后续的主观实验提供了统一环

境。

综上所述，本文引入头部姿态的新模态，来辅助手势生成模型从历史与现在信息推理手势的能力，并且提供了一个实验场景验证模型的推理质量与实时性能。

1.3 手势的定义与身体姿态的参数化表示

1.3.1 手势的定义与范围

在人类交流中，手势是常与语音同时出现的身体动作信号，它承担语义补充、情感表达与互动调节等多重功能。这类动作通常呈现出明确的表达意图，使其区别于姿态平衡，移动等纯功能性动作^[4]。因此，手势与语言并非两个相互独立的系统，而是源于共同的认知的表达机制。

在本研究的广义定义下，手势的运动形式不局限于手部运动，还可扩展至头部运动、躯干姿态等与表达相关的所有上半身动作。

Kendon 连续体 Kendon 连续体^[4-5]将与表达相关的手势行为，基于语言化程度做了以下分类：gesticulation（随语手势）→ language-like gestures（语言样手势）→ pantomime（拟态/哑剧式动作）→ emblems（约定俗成的象征手势）→ sign language（手语）。越靠左的类型通常更依赖当前的言语与语境、形式更即兴；越靠右则越接近离散的符号系统，规约化程度高，意义更稳定，可在缺少口语的情况下独立传达。

当前人机交互、虚拟人/数字人驱动等方向的手势生成，多数工作聚焦于 Kendon 连续体最左侧的随语手势，即说话过程中自然出现、与语音节奏与语义强关联的上肢动作。其含义往往依赖当前的口语与语境，脱离它们时通常难以传达清晰含义。

相对而言，连续体右侧的手势更接近文化中约定俗成的符号，可以脱离口语独立传达含义。例如，表达称赞的拍手动作，或表示“请保持安静”的嘴前竖起食指的动作，这些被视为象征手势，通常不在随语手势生成领域的研究对象中。

本文的研究范围据此限定在随语手势的学习与生成。

随语手势的分类 McNeill^[5]将随语手势进一步划分为四种基本类型：

1. **Iconic gestures**（形象性手势）：以具象方式描绘事物的外形、空间路径或动作特征。例如，用手势勾勒一个物体的轮廓，或划出一道线表示移动的轨迹。此类手势与语言内容直接对应，表达具体语义。
2. **Metaphoric gestures**（隐喻性手势）：表达抽象概念或思维结构的手势，比如用

双手做出捧起一个物体放到一边的动作，表示“先把这个话题放一边”。这种手势并不描绘实体，而是以具象化的方式呈现抽象语义。

3. **Deictic gestures**（指示性手势）：指向空间中的对象、人物或方向，常用于对话焦点的指明与注意引导。
4. **Beat gestures**（节奏型手势）：与语音重音、韵律或节奏同步的节奏性动作，通常不承载具体语义，但可用于强调语音节奏，引起听众对说话内容的注意。

这四类手势构成了随语手势在语义与语篇功能上的主要维度，并在自然交流中常以复合形式出现。在手势生成任务中，研究通常将其视为不同的可生成目标：其中节奏型手势由于与语音韵律高度同步、对齐与建模相对容易，长期以来在数据驱动方法中更常被优先刻画；而形象性与隐喻性等语义相关手势则对文本的语义推理有更高要求，因而是更具挑战性的方向。

鉴于本文以低延迟在线驱动为目标，本文优先建模与韵律强耦合的节奏型手势；语义一致的形象性/隐喻性手势留作后续工作。

节奏型手势在随语手势中的重要性 尽管节奏型手势通常不承载具体语义信息，已有研究表明，其在交流效果与听众感知层面仍具有独立的价值。Baars 等的实验^[6]比较了无手势、仅使用节奏型手势、以及包含了形象性、隐喻性、指示性的意义性手势的三种演讲条件，结果显示，相较于完全不做手势，仅使用节奏型手势即可显著提升听众对说话者自然度的主观评价，并一定程度上提升了听众对演讲内容的记忆表现。而包含意义性手势的演讲条件在自然度与听众的记忆保留等指标上并未显著优于节奏型手势条件。这一发现表明，即便缺乏形象性或隐喻性的语义映射，节奏型手势仍能通过与语音韵律的同步，对交流过程产生积极影响。

从功能上看，节奏型手势主要服务于语篇结构与韵律组织，其作用并非传递附加语义，而是通过时间对齐、重音标记与注意力引导，增强语音信息的感知显著性与节奏感。在真实的人机交互与虚拟人系统中，这类手势常被作为一种低语义依赖、但高度稳健的非语言表达形式加以采用。

鉴于本文面向低延迟、严格逐帧的在线驱动场景，系统在任一时间步均无法获取未来文本或完整语义结构，对语义一致性要求较高的形象性与隐喻性手势难以可靠生成。相比之下，节奏型手势主要依赖于当前及局部时间窗口内的语音韵律特征，更适合在实时条件下进行稳定建模与生成。因此，本文选择以节奏型手势作为主要研究对象，并将其视为一种在系统约束下具有明确交互价值的可行随语手势形式。

头部手势的分类 除手部动作外，头部动作同样是手势的重要组成部分。头部的点动与摆动在时间结构上常与手势及语音节奏保持同步^[7]，在语用功能上既能辅助语音韵律的组织，也能表达态度与指向信息。

在不同研究中，头部动作被从多个维度加以分析，其主要功能可归纳为以下几方面：

1. **韵律相关（prosodic）** 动作反映语音重音与句法节奏的对应关系^[8]；
2. **语义或态度相关（semantic/attitudinal）** 动作表达说话者的情绪倾向与交际意图^[4-5]；
3. **指向相关（deictic）** 动作通过转头或注视方向建立叙事空间的参照^[5]。

此外，研究表明头部动作的启动时间往往早于发声^[9]。具体而言，头部动作存在启动与加速过程，若其峰值需与重读音节的时间对齐，则动作必须提前起势。因此，头部动作可能对即将到来的语音韵律具有前瞻性。

这一特征揭示了头部动作与语音之间的时序关系，说明视觉模态中的运动信号有时可先于声学事件出现。本文研究也因此关注头部动作，并将其纳入输入模态。

1.3.2 身体姿态的参数化表示

手势作为身体运动的子集，其生成和识别依赖于身体姿态的连续建模。因此，在进一步讨论手势生成方法之前，在此明确身体姿态的参数化表示方式。

骨架结构的定义 在计算机动画与动作捕捉领域，身体姿态通常由骨架结构和关节旋转参数共同定义。骨架结构描述了人体各关节的拓扑关系及层级依赖；而每个姿态帧由一组关节旋转参数所确定，这些参数定义了相对于父节点的旋转变换。在不同的系统与任务中，骨架结构的具体形式往往有所差异，这种差异直接影响姿态数据的表示与学习方式。

在不同的系统与任务中，骨架结构可以遵循各自的标准，因此，不同的数据集、3D 模型或神经网络往往基于自身定义的关节层级与命名体系进行训练与标注。例如，AMASS 数据集^[10]采用 SMPL 拓扑结构^[11]，BEAT 数据集^[3]使用简化上半身骨架。近年来的自动骨骼绑定与骨架归一化方法，通过学习或优化关节对应关系，实现了不同拓扑之间的姿态重定向（pose retargeting）^[12-13]，从而消除了模型依赖于特定骨架结构的限制。

旋转参数的选取 在确定骨架结构之后，具体的关节状态可通过多种旋转参数进行描述。常见的旋转参数表示方法包括：

1. 欧拉角 (Euler Angles): 通过三个顺序旋转角表示姿态，直观且参数维度低 (3 维)，适合存储，但运算存在万向节锁 (Gimbal Lock) 问题，导致特定姿态下自由度丢失，且插值过程易产生非物理运动；
2. 四元数 (Quaternion): 以四维单位向量 ($q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$) 表示旋转，避免欧拉角的奇异性的同时，实现平滑插值。然而在神经网络回归中通常需要显式处理单位长度约束，且同一旋转存在双覆盖 (q 与 $-q$ 等价)，优化不稳定。
3. Axis-Angle: 以旋转轴 (三维单位向量) 与旋转角度 (标量) 的乘积表示旋转，参数紧凑 (3 维)，是参数化人体模型 SMPL^[11] 的主要姿态存储格式。但角度取值范围存在周期性 ($[0, 2\pi)$)，导致参数空间不连续，影响神经网络学习；
4. 旋转矩阵：以 3×3 的正交矩阵，表示坐标系或物体绕某个轴旋转一定角度后的新姿态。常用于计算机图形学、机器人学和物理仿真中，但正交约束 ($R^T R = I$ 且 $\det(R) = 1$) 提高神经网络的学习难度。
5. Rot6D^[14]: 将 3×3 旋转矩阵的前两列展开为六维向量，通过 Gram-Schmidt 正交化过程自动保持列向量的正交性与单位长度，无需额外约束。该表示既继承了旋转矩阵的数值稳定性，又解决了其参数冗余和正交性约束难题，同时具备连续性优势。

在计算机图形与实时渲染中，通常采用四元数或旋转矩阵进行骨骼变换与插值，以保证数值稳定性和计算效率。然而，在深度学习任务中，这些表示方式的约束会对训练效率造成影响：四元数需满足单位长度约束；旋转矩阵则需满足元素的正交约束。近年来的研究表明^[14]，Rot6D 既规避了上述两种表示的约束难题，又保持了姿态空间的连续性与物理合理性，在动作生成与姿态预测任务中展现出更优的可学习性。

手势生成方法对旋转参数的选取经历了一个演变过程。早期工作采用的方法各异（如欧拉角、Axis-Angle），多受限于特定框架或历史因素；而近期研究，出于对神经网络训练稳定性的追求，已显著趋向于采用 Rot6D 表示。这一趋势的实例对比见表 1.1。

因此，本文在姿态生成模型中统一使用 Rot6D 表示每个关节的旋转状态，以确保训练阶段的平滑收敛与推理阶段的稳定性。

表 1.1 不同旋转表示方式的空间连续性与使用示例

Table 1.1 Spatial Continuity and Usage Examples of Different Rotation Representations

表示方式	维度	连续性	典型应用场景	训练使用示例
欧拉角	3	存在万向节锁	姿态存储	CaMN ^[3]
四元数	4	连续, 但存在单位长度约束	图形学	—
Axis-Angle	3	角度部分不连续	神经网络	DiffSHEG ^[15]
旋转矩阵	9	连续, 但存在正交约束	图形学、机器人学	MambaGesture ^[16]
Rot6D	6	连续	神经网络	近期多数 ^[17-19]

1.4 面部表情的定义与参数化表示

面部表情是非语言交流的重要组成部分, 与语音、手势共同传递情感和态度信息。与身体动作不同, 面部表情主要由皮肤形变和局部运动构成, 无法通过骨骼旋转直接建模, 因此需要专门的参数化表示方式。

目前常见的面部状态描述方法主要有两类:

BlendShape 权重模型 Facial Action Coding System (FACS)^[20]提出复杂表情可分解为若干可组合的基本动作单元 (Action Units, AU), 为表情的参数化表示提供了理论基础。受此启发, BlendShape 模型将面部表情表示为一组可线性叠加的形变基, 每个基形对应一个权重控制的局部表情变化。复杂表情由多个基形的组合生成, 其思想与 FACS 的动作单元体系相呼应。

与基于骨骼的形变不同, BlendShape 直接在顶点层面定义几何偏移量, 因此对网格拓扑结构高度依赖: 每个基形的顶点偏移需与基础网格逐点对应。在具有相同网格结构的角色模型之间, BlendShape 集合可以直接复用; 但若拓扑发生变化, 偏移数据将无法一一对应, 从而难以在不同模型间映射或重定向。这种拓扑依赖性限制了其跨模型的通用性。

尽管如此, BlendShape 在表现精细面部表情和软组织形变方面具有显著优势。其标准化权重接口、实时可驱动性与渲染兼容性, 使其成为虚拟人和表情捕捉系统的主流表示形式, 并被广泛应用于如 ARKit^[21] 等实时动画框架, 以及主流渲染引擎中。

图 1.1 来自^[22], 展示 FACS 对闭眼睛的动作单元 AU45 的定义。

这在 BlendShape 中, 一般对应两种基形: eyeBlinkLeft (闭左眼) 与 eyeBlinkRight (闭右眼)。图 1.2 展示了 eyeBlinkLeft、eyeBlinkRight 的 ARKit BlendShape 基形在权重 $w \in [0, 1]$ 下的线性插值效果 (从张眼到闭眼)。

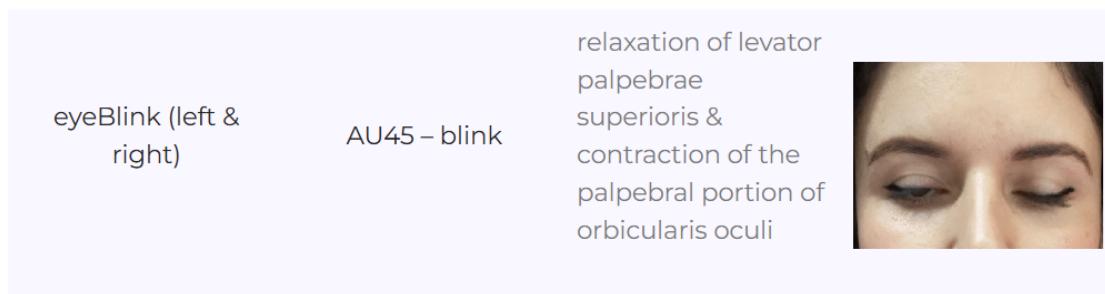


图 1.1 FACS 中闭眼动作单元的定义

Figure 1.1 Definition of Eye Closure Action Units in FACS



图 1.2 BlendShape 在闭眼形变上的线性插值效果

Figure 1.2 Linear Interpolation of BlendShape Deformation for Eye Closure

关键点坐标模型 关键点模型通过检测面部若干语义特征点（2D 或 3D 坐标）来描述几何结构变化。典型实现包括 MediaPipe Face Mesh^[1]与 OpenFace^[23]系列。

与基于网格形变的 BlendShape 不同，关键点模型并不依附于任何具体网格拓扑，而是在几何空间中以语义一致的特征点集合形式定义面部结构。这种表示方式并不描述模型的形变，而是对人脸几何的抽象建模，因此常用于表情识别、头部姿态估计等分析任务，而较少用于驱动渲染。

在多模态学习与特征分析中，BlendShape 表示具有较高的统一性和可量化性，适合以固定维度向量作为模型输入，并易于应用于不同虚拟角色的动画驱动。因此，本文在系统设计中采用与 ARKit 兼容的 BlendShape 参数作为面部模态输入特征。



图 1.3 MediaPipe Face Mesh 关键点结构示意图
Figure 1.3 Illustration of MediaPipe Face Mesh Landmark Topology

1.5 实时生成的理论基础与可行性分析

从生成可行性的角度，现有研究普遍认为节奏型手势可在无语义理解的条件下由语音韵律直接驱动生成。多数语音驱动手势研究证实，仅凭语音的能量、时长与音高变化即可合成自然的节奏性上肢动作^[24-26]。这些研究所生成的动作在时间结构上与语音重音同步，体现了语音与手势共享的时间规划机制。

相比之下，iconic（形象性）、metaphoric（隐喻性）与 deictic（指向性）手势均依赖语义或指向关系，需要从上下文分析语义与语境，难以在严格实时的因果条件下生成。而节奏相关特征在音频中则具有更高的可预测性。^[27] 这表明，在缺乏未来语义与全局上下文的实时场景中，仅凭语音模态，模型只能稳定生成节奏层面的动作。

为突破这一限制，本文引入头部姿态模态作为补充输入信号。头部姿态能在实时因果条件下提供部分空间与时间线索：其转头与注视方向反映互动焦点，点头与抬头与语音重读共现，能够在不依赖未来语义信息的前提下，为手势生成提供弱先验约束。这种模态扩展为实时系统提供了理论上的可行性基础，使模型能够在语音之外获得关于节奏、方向与视角的附加信息。

头部姿态对手势预测的贡献 头部动作在自然语音中常呈现出一定的时间前瞻性^[9]：其启动往往早于对应韵律词的发声，这意味着视觉模态可能比声学信号更早反映语音节奏的变化趋势。这种时序特性为实时生成任务提供了潜在的预测窗口，使系统能够在语音节奏变化尚未显现前，就提前捕获相关的动态线索。因此，头部姿态在实时生成中不仅提供同步参考，也可能在时间上形成前驱信号，为手势节奏的自然启动提

供时序优势。

头部姿态对空间锚定与视角一致性的贡献 头部姿态模态为实时语音驱动的手势生成提供了关键的空间参照信号。其与语音韵律在时间组织上高度耦合。即使在无未来语义信息的条件下，头部的转向与注视变化仍能反映说话者的注意焦点与叙述方向，从而帮助模型在动作生成中保持空间的连贯性与方向一致性。这一机制使系统能够在时间与空间两个维度上同步对齐语音与动作，让生成的手势在视觉上更具互动感与表达意图。

在 McNeill 的四类手势体系中，头部姿态的引入主要强化了两类动作的生成：(1) 对 beat 手势而言，它为语音重读和节奏段落提供显式的时间协同信号，使手部与头部动作在韵律层面更加一致；(2) 对 iconic 手势而言，它在具有路径与方向特征的动作中提供空间参考，使模型能够在叙事空间中更稳定地确定动作的方位与轨迹方向。通过这两方面的强化，系统在保持实时性的同时获得了更自然的节奏衔接与空间表达。

与此同时，本文明确头部姿态模态的作用边界：其核心优势在于捕捉方向、焦点与时序节奏，而非手型语义或复杂形态描摹等细粒度语义特征。换言之，它主要改善手势的位置、方向与视角依附，而非手势的形状描绘或语义内容。对于依赖抽象语义或外指参照的 metaphoric 与 deictic 手势，仍需语言或上下文模态的补充。

总体而言，头部姿态为实时生成提供了介于韵律与语义之间的关键中层约束。其时间上的前瞻性与空间上的指向性共同帮助模型在低延迟条件下保持自然、连贯且空间协调的动作表现，从而在因果生成框架内有效拓展了语音驱动手势的可表达范围，并为节奏主导型动作的实时生成提供了结构的支持。

1.6 国内外研究现状

近年来，语音驱动手势生成经历了从规则设计到数据驱动模型、再到多模态扩展与实时生成的持续演变。这一过程不仅体现了算法架构的更新，也反映了研究目标与应用场景的变化：从基于语言规则的行为映射，到学习语音—动作关系的深度生成模型，再到面向交互的多模态实时系统。

1.6.1 规则驱动阶段

早期的手势生成系统主要依赖语言学规则与专家知识构建^[7,28-30]。这类方法通过语义分类或韵律规则将语音片段映射为预定义的手势模板（如指示、肯定、节奏性动

作), 并以有限的动作库组合出手势序列。它们可在虚拟代理或机器人中实现基于语音的同步动作。然而, 手势词典与语法规则的人工设计成本较高, 难以覆盖自然语音中的多样变化, 导致生成结果缺乏自然性与个体差异。

1.6.2 数据驱动阶段

随着大规模语音与动作配对数据的出现, 研究者开始采用统计学习和深度神经网络模型学习语音一手势映射关系。在此阶段, 语音通常作为唯一输入模态, 模型通过 LSTM、MLP 等结构预测连续手势序列。典型代表如 CaMN 模型^[3], 其基于 BEAT 数据集^[3]训练级联网络, 将 LSTM、全连接网络与 GAN 结构相结合, 实现从语音到动作的端到端预测。

然而, 该类模型多使用欧拉角或离散旋转参数作为手势表示, 生成结果容易出现抖动与不连续。后续工作引入更平滑的表示方式, 如 Rot6D^[14,17-19]或 Axis-Angle^[15], 显著提升了动作流畅性。与此同时, 为解决语音与手势间的多对多映射问题, 研究者引入了 VQ-VAE^[17,31]与扩散模型^[15,32-35], 在保持自然性的同时提升了生成多样性与表现力。

尽管这些方法在客观指标与视觉效果上优于传统模型, 但通常依赖完整语句级上下文。在用户语音下的流式逐字输入场景中, 为获取未来上下文以进行语义判别与韵律对齐, 需引入缓冲机制, 因此即使推理较快的模型^[15], 整体延迟也因上下文缓冲造成端到端的显著延迟。

1.6.3 多模态扩展阶段

为进一步提升动作表现力与语音理解能力, 部分研究引入视觉模态或语言语义特征。例如, CaMN^[3]在语音输入的基础上融合面部捕捉信息以增强表现; EMAGE^[17]与 DiffSHEG^[15]同时生成手势与面部动作; DiffTED^[34]实现了端到端的视频合成。

这些多模态生成方法在提升虚拟智能体的自然感与沉浸感方面表现优异, 但其任务假设仍基于整句输入, 因此主要用于 AI 虚拟形象生成或离线内容创作场景, 而非实时用户交互。

1.6.4 总结

本节关注近年来代表性生成范式及其在线化需求; 其中部分方法在离线设定下效果突出, 但本文受严格因果与低延迟约束未予实现, 仅作为后续工作参考与对照基线。近年来的手势生成研究在建模策略上呈现两类趋势:

1. **扩散模型（Diffusion-based generation）：** 扩散模型在手势生成中常带来较高的动作自然度与多样性^[15,32-34,36]。现有工作多以固定长度的语音/文本片段作为条件，在迭代去噪采样中生成整段手势序列，因而在直接迁移到在线场景时会引入片段缓冲的延迟。另一方面，也已有研究开始探索离线实时或任意长度生成的采样与拼接策略（如基于扩展/外延采样的设计），表明扩散式框架具备向流式化演进的潜力^[15]。
2. **语义增强方向（Semantic-aware generation）：** 为覆盖 iconic、metaphoric 等更依赖语义的手势，引入语义表征作为额外条件^[36-37]，以提升动作与语义的一致性与表现力。在线场景下，该方向的主要挑战通常来自语义信号的获得方式：现有工作使用整句/整段文本嵌入，基于更充分的上下文以稳定语义对齐，因而需要通过有限前瞻或缓冲策略来权衡延迟与语义质量。

总体而言，上述方法多在离线或半离线设定下达到最佳效果；若面向严格低延迟的在线实时生成，需配合流式条件建模、有限前瞻与快速采样等机制。本次将作为后续工作的技术储备与对照基线。

1.7 研究目标的差异

语音驱动的手势生成研究在总体目标上虽一致——即让虚拟角色的动作与语音内容、节奏协调一致——但在使用场景与系统角色上存在显著差异。

现有研究大体可分为两类：

为 AI 的虚拟形象生成手势 这一类研究的目标是让 AI 驱动的文本对话系统的虚拟形象具备手势表现力。

模型可以一次性生成下一句语音或文本，因此可以访问完整的未来信息，包括整句音频、文本和语义上下文。

典型方法通过编码完整句子的节奏与语义，预测整段动作轨迹，以最大化动作与语义的一致性和整体流畅性。

这类方法适合 AI 驱动的系统或离线生成的应用场景，如合成视频。

为用户的虚拟人生成实时手势 本文所聚焦的目标类型属于第二类。

在用户实时说话的过程中，系统需根据当前语音流（以及可选的面部表情与头部姿态）即时生成同步手势。

此任务具有严格的实时性约束与因果性限制：模型在每一时刻只能使用当前及过去的信息，而不能访问未来语音或文本内容。

该任务更接近实时交互系统，而非内容生成系统。

1.8 任务约束与可利用条件的差异

这两类研究目标在可利用的信息条件和评价重心上存在本质区别：

表 1.2 离线与在线手势生成任务在约束与可利用信息上的对比

Table 1.2 Comparison of Constraints and Available Information between Offline and Online Gesture Generation

对比维度	AI 虚拟形象生成	用户虚拟人实时生成
输入信息	完整句级语音或文本（可使用未来信息）	实时语音流，仅使用过去与当前帧信息
输出目标	整句手势序列（离线生成）	连续流式手势（逐帧生成）
时间约束	不必要实时	帧级实时性（<50 ms 推理延迟）
评价重点	整体语义一致性与美学自然度	瞬时同步性、动作平滑与交互稳定性
应用场景	离线动画、内容合成、AI 虚拟直播	实时虚拟人、视频会议、用户虚拟直播
语音模态	作为输入或由文本生成（TTS 输出）	作为实时输入特征（语音流）
手部手势	生成目标（输出）	生成目标（输出）
面部表情	通常为生成目标（输出）	可通过设备实时采集，作为输入辅助推理
头部姿态	通常为生成目标（输出）	可实时采集并作为输入特征，用于同步推理

前者可以在生成阶段规划动作节奏与语义对应，而后者需在无未来信息的条件下维持自然与同步，并保证输出连续、平滑且无突跳，从而对模型结构、输入模态与延迟控制提出更高要求。

1.9 手势生成评估方法

语音驱动手势生成研究的评价体系通常涵盖生成质量、时序匹配与表达多样性等多个维度。研究者既关注生成动作的自然性与视觉流畅度，也重视其与语音信号在节奏和语义层面的对应关系。近年来，随着实时生成和多模态扩展任务的发展，相关评估方法也逐渐体系化，可概括为以下几类：

1. **自然性 (Naturalness)** ——衡量生成动作在运动平滑性、速度变化及能量分布上的合理性，常采用 FGD (Fréchet Gesture Distance)^[24]、运动速度统计、或主观“自然度”评分等指标。
2. **同步性 (Synchronization)** ——评估手势在时间上与语音重音或韵律事件的对齐程度，常用 BA (Beat Alignment)^[27] 等方法，以及基于重读检测的主观同步性评价。
3. **多样性 (Diversity)** ——衡量模型在不同语音输入下生成的动作变化程度，通常以轨迹分布的方差、速度曲线差异或 L1 范数等指标度量，以防止模型陷入单一模式或过度平滑。
4. **语义相关性 (Semantic Relevance)** ——反映生成动作与语义关键词或情绪类别的一致性，可通过 SRGR (Semantic Relevance to Gesture Ratio)^[3] 等指标或人工标注语义标签对齐评估。
5. **实时性与稳定性 (Latency & Robustness)** ——在面向交互系统的研究中，还需评估帧级推理延迟与输出平滑性，以确保动作流连续且系统响应及时。

总体而言，现有评价体系既包含客观的运动学与统计指标，也结合主观感知评分，在不同任务目标下可形成“自然性—同步性—多样性—相关性”的综合评估框架。

1.10 本文研究目标

本文研究的目标是设计一种能够在实时条件下运行的语音驱动手势生成模型，使用户无需动作捕捉设备或特定硬件，仅通过语音输入与相机前的面部表情、头部动作即可驱动虚拟人的上肢与头部动作。与以往主要面向离线生成或预先制作型虚拟形象生成的研究不同，本文关注的任务场景为用户实时交互，因此系统必须在严格因果的条件下运行，即仅利用当前与过去的输入帧信息进行动作预测，避免依赖未来语音或文本内容。此外，实时运行对推理延迟与计算效率也提出了更高要求。

为满足上述需求，本文进一步引入用户的头部姿态作为辅助输入模态，为手势生成过程提供额外的非语言信号。头部姿态能反映注意方向与交互焦点，并在语音节奏发生变化时为动作的时序组织提供参考。通过将语音、面部与头部信号联合输入模型，系统能够在实时条件下获得更丰富的上下文线索，从而支撑连续动作的稳定生成。

本文以 CaMN 模型^[3]为基础进行扩展。CaMN 原为离线级联结构，其输入包括语

音与面部捕捉特征，输出包含手部上肢动作与头部姿态。本文将其输入机制改写为逐帧输入的流式推理形式，并在此基础上引入头部姿态特征分析模块，将头部姿态作为独立通道输入至级联网络的后级层，以实现语音、面部与头部信号的联合驱动。最终，系统能够在实时语音流输入条件下逐帧生成骨骼动画输出，满足实时交互场景对因果性与低延迟的要求。

因此，本文的研究内容汇总如下：

1. 面向实时交互的任务定义与系统流程设计：提出严格因果、帧级推理的语音驱动手势生成任务设定，并构建从语音、面部与头部实时采集到骨骼动画输出的端到端驱动流程。
2. 基线模型的实时适配与训练策略：以 CaMN 级联架构为基础，将其离线输入机制改写为逐帧流式推理形式，并结合单向时序建模与滑动窗口自回归训练策略，使模型在仅依赖历史与当前帧信息的条件下生成连续动作。
3. 头部姿态输入模态的建模与融合结构设计：将头部旋转姿态作为新的实时输入模态，设计头部姿态特征表示与编码器，并探索其与语音、面部表情的联合输入方式，形成后级融合的多模态级联手势生成结构。
4. 实验与评估平台实现：实现统一的数据处理、推理与渲染流程，使不同模型能够在相同输入条件下进行动作生成与对比，并支持客观指标与主观实验的评估流程。

1.11 本章小结

本章综述了语音驱动手势生成领域的相关研究现状与发展脉络。首先，对手势的概念与在计算机中的参数化表示进行了阐述，说明了手势与面部表情、头部姿态在虚拟人交互中的角色与差异。随后，从研究目标的角度分析了不同任务设定之间的区别，指出现有大多数工作聚焦于为 AI 虚拟形象生成整句级动作，而缺乏面向用户实时交互的研究。在此基础上，回顾了手势生成方法从规则驱动到数据驱动、从单模态到多模态的演变过程，总结了现有模型虽在生成质量上取得显著进展，但在实时性与因果性方面仍存在局限。

最后，结合本文的研究目标，提出了面向实时交互的语音驱动手势生成方案。

1.12 论文组织架构

(待更新) 本文共分为五个章节，内容安排如下：

第一章为绪论，介绍本研究的背景与意义，阐述研究目标、主要内容及核心贡献，并说明论文的整体组织结构。

第二章为相关工作，回顾了国内外在语音驱动手势生成、多模态学习及实时面部捕捉技术等方面的研究现状，分析了现有方法的不足，并明确了本文的研究定位与创新点。

第三章为方法，详细介绍了本文提出的 FaceCapGes 模型的整体架构与算法流程，包括头部姿态的多模态输入编码机制、姿态解码机制、滑动窗口自回归训练策略及对抗优化过程。

第四章为评估，详细介绍了客观评估结果与用户调研结果，对生成质量进行综合分析，并进一步通过消融实验与推理速度测试验证头部姿态输入与帧级自回归生成策略在实时交互场景下的有效性。

第五章为结论，总结本文的主要研究成果与贡献，讨论当前系统的局限性，并对未来在语义手势生成与未来趋势预测建模方向上的研究进行展望。

第2章 基于语音、面部、头部姿态的在线实时手势生成系统

2.1 研究定位与总体设计思路

手势可根据语义依赖性与时间结构复杂度区分为可语义生成与可韵律生成两大类。本文的研究聚焦于严格实时的语音驱动任务，在此条件下模型无法访问未来语音或完整语义，因此重点生成与语音韵律同步的节奏型手势，并通过面部与头部模态的联合输入进一步增强表达性与自然度。

基于上述定位，本文提出了一种基于音频、面部 BlendShape 权重和头部姿态输入的帧级多模态级联手势生成模型 FaceCapGes。模型旨在在实时条件下实现自然、同步且具有一定指向性的上半身动作生成，在不依赖语义理解或未来上下文的前提下，通过多模态输入弥补语音模态预测能力的不足。

为避免与系统级说明混淆，本文在第 2.2 节给出端到端系统框架与各模块的功能划分；在第 3.1 节形式化定义任务目标、输入输出模态与符号体系；在第 3.2 节详细介绍模型的级联结构、模态编码方式；在第 ?? 节详细介绍滑动窗口、自回归训练等实时适配策略；在第 3.3 节介绍头部姿态的引入方法与编码器；在第 3.4 节展示模型的整体结构图；最后在第 5.1 节说明实现细节与训练配置。

2.2 系统整体框架与模块定位

本节介绍整个系统的端到端驱动流程及模块职责划分。如图 2.1 所示，系统整体架构由五个层级组成：用户配置层、设备层、中间件层、手势生成模型层以及渲染与驱动层。各层之间通过多模态信号接口进行连接，实现从信号采集到虚拟人动作生成的端到端实时处理。

FaceCapGes 模型位于中间层，承担多模态输入到上半身姿态输出的核心推理任务，而输入采集与渲染模块分别负责信号获取与结果展示。

为实现基于语音、面部捕捉与头部姿态的实时数字人驱动系统，本文构建了完整的信号采集、动作生成与渲染展示的处理管线。FaceCapGes 模型作为该系统的核心计算模块，负责在实时约束下从多模态输入推理出当前帧的上半身骨骼姿态。

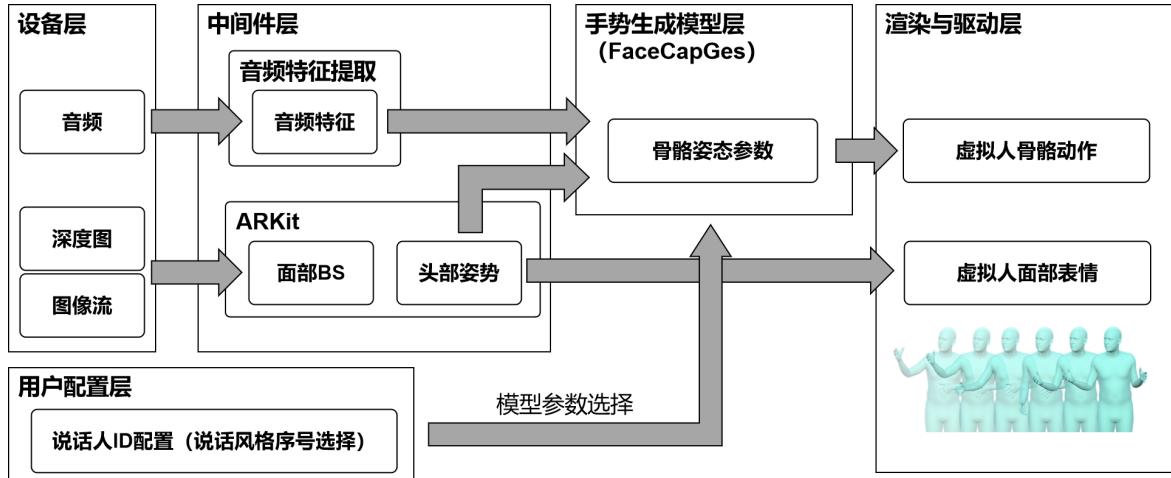


图 2.1 系统整体架构与数据流示意图
Figure 2.1 System Overview and Data Flow Diagram

2.2.1 信号采集与系统配置层

该层位于系统整体架构的输入端，用于从用户端设备实时获取多模态信号，并在系统初始化阶段完成运行参数的配置。整体结构可划分为设备层、中间件层与用户配置层三个部分，如图 2.1 所示。

设备层 设备层负责采集语音与视觉模态信号。语音信号由麦克风实时录制，采样率与帧移可根据运行设备性能调整；视觉信号由前置深度相机摄像头获取面部深度图与视频流，并作为 ARKit 面部追踪模块的输入。

中间件层 中间件层通过 Apple 提供的 ARKit 框架^[21]，将设备层的原始图像流与深度图转化为结构化特征。ARKit 输出两类主要数据：(1) **面部表情特征** ARKit 提供 52 维 BlendShape 系数向量，用于描述关键肌肉群的局部形变状态。该特征能够反映用户的表情、口型与情感变化，并以帧级形式同步输出。(2) **头部姿态特征** ARKit 在 ARFaceAnchor 中提供一个齐次变换矩阵 $\mathbf{T} \in \mathbb{R}^{4 \times 4}$ ，用于描述人脸锚点相对会话世界坐标系的位姿：

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}, \quad \mathbf{R} \in \mathbb{R}^{3 \times 3}, \mathbf{t} \in \mathbb{R}^{3 \times 1}. \quad (2.1)$$

其中左上角的 \mathbf{R} 为旋转矩阵，右上角的 \mathbf{t} 为平移向量。本研究从矩阵中提取旋转部分 \mathbf{R} ，并将其转换为 Rot6D^[14]表示形式，以提升旋转空间的连续性与模型训练的稳定性。

同时，音频流在中间件层中被传入特征提取模块以生成时间序列特征。模型训练阶段使用 Librosa 库离线提取 Mel 频谱、短时能量与基频 F_0 等声学特征，以保证特征精度与一致性。系统运行阶段可由等价的实时特征提取模块（如 torchaudio 或 TensorFlow Audio）逐帧生成对应特征，以实现端到端的低延迟运行。

用户配置层 用户配置层负责系统初始化阶段的模型与参数设定。用户可在应用中选择说话风格，对应加载不同说话人 ID 配置下的模型权重。该配置仅在系统启动时生效，不参与实时推理过程。

本层提供的多模态信号经中间件处理后，以统一的数据接口传递至手势生成模型，实现语音、表情与头部姿态的实时融合输入。

2.2.2 手势生成模型层（FaceCapGes）

FaceCapGes 模块位于系统的中间层，是本文提出的核心计算单元。该模块接收来自信号采集与系统配置层的三类输入特征：语音特征、面部 BlendShape 系数以及头部姿态参数，并在不依赖未来帧的条件下，逐帧预测用户当前时刻的上半身骨骼姿态。

生成的骨骼姿态采用 Rot6D^[14]连续旋转表示形式，覆盖上半身 47 个关节的旋转参数。模型内部通过级联多模态编码结构提取时序相关特征，并利用单向 LSTM 编码器完成时间依赖建模，从而在保持实时性的同时，生成与语音节奏、表情变化及头部朝向高度一致的自然手势。

FaceCapGes 输出的姿态数据通过统一接口传递至渲染与驱动模块，与实时面部捕捉信号共同驱动虚拟角色的整体动作。由于模型仅依赖当前与历史帧输入，可与输入层以固定帧率并行运行，实现端到端的低延迟推理。

2.2.3 渲染驱动层

该模块位于系统输出端，负责将手势生成模型与面部捕捉结果共同转化为虚拟人的实时动作表现。系统将 FaceCapGes 模型输出的上半身骨骼姿态与 ARKit 实时检测的 52 维面部 BlendShape 系数传递至渲染引擎，由引擎内的模块解析并映射至目标虚拟人的骨骼与表情控制接口，从而实现多模态动作驱动。

渲染模块采用基于 GPU 的蒙皮计算与实时光照模型，以确保动画的平滑性和视觉一致性。最终，系统能够在实时流式输入条件下稳定运行，同步呈现语音、表情与身体动作，以自然流畅的数字人形象实现从多模态信号输入到可视化输出的完整驱

动流程。

第3章 头部姿态在多模态级联架构中的引入

??

3.1 问题定义

在整体系统中，FaceCapGes 模块承担着从多模态输入信号到上半身骨骼姿态预测的核心任务。为了明确模型的输入输出结构与学习目标，本节对该问题进行形式化定义。

3.1.1 任务描述

目标是在实时条件下，根据用户当前时刻的语音、面部表情与头部姿态信息，预测其对应的上半身骨骼姿态。模型需能够逐帧生成与语音节奏、面部动态和头部转动方向相协调的自然手势动作，而不依赖未来的输入帧或整句语音信息。

形式上，可以将该任务定义为一个多模态时序映射函数：

$$\hat{\mathbf{v}}_t^B = f_{\theta}(\mathbf{v}_{t-N:t}^A, \mathbf{v}_{t-N:t}^F, \mathbf{v}_{t-N:t}^H), \quad (3.1)$$

其中 f_{θ} 表示由参数 θ 控制的生成模型， N 为历史窗口长度。各模态输入定义如下： \mathbf{v}_t^A 表示语音模态在时刻 t 的特征向量，由麦克风信号经特征提取模块得到； \mathbf{v}_t^F 表示面部模态的输入，为 ARKit 输出的 52 维标准化 BlendShape 系数； \mathbf{v}_t^H 表示头部模态的输入，为 ARKit 得出的头部旋转矩阵经 Rot6D 表示；而 $\hat{\mathbf{v}}_t^B$ 为生成模型在当前时刻预测的上半身骨骼姿态向量。模型仅利用当前及过去 N 帧的输入信息估计 $\hat{\mathbf{v}}_t^B$ ，从而满足严格的实时推理约束。

3.1.2 输入与输出模态

FaceCapGes 模型的输入由三种可同时实时获取的模态组成：语音特征、面部 BlendShape 权重及头部姿态参数；输出为当前帧的上半身骨骼旋转状态。各模态的符号与维度如表 3.1 所示。

输入序列 $(\mathbf{v}_{t-N:t}^A, \mathbf{v}_{t-N:t}^F, \mathbf{v}_{t-N:t}^H)$ 描述了用户在过去 N 帧内的语音与表情动态信息。模型通过学习其时序变化规律，逐帧生成对应的骨骼姿态输出 $\hat{\mathbf{v}}_t^B$ 。在推理阶段，模型仅访问至时刻 t 的输入序列，无法访问任何未来帧信息，保证了生成过程的因果性与实时性。

表 3.1 输入输出模态符号与维度
Table 3.1 Notations and Dimensions of Input/Output Modalities

模态	符号	维度	描述
语音特征	v_t^A	\mathbb{R}^{1067}	由音频信号提取的时序特征（Mel 频谱、短时能量、基频等）
面部 BlendShape	v_t^F	\mathbb{R}^{52}	ARKit 输出的标准化表情权重向量
头部姿态	v_t^H	\mathbb{R}^6	采用 Rot6D 表示的头部旋转参数
骨骼姿态（输出）	\hat{v}_t^B	$\mathbb{R}^{6 \times 47}$	上半身 47 个关节的旋转状态

3.2 级联架构设计的继承

3.2.1 级联架构的原理与理论背景

现有语音驱动手势生成模型多采用多模态融合结构，其中以 CaMN^[3]为代表的级联架构在设计理念上具有代表性。其核心思想是将语音、面部表情与身体动作视为语义表达的不同层级：语音模态承担语义与节奏驱动作用，面部模态反映情感与意图，身体动作则是语言与情绪的外化呈现。CaMN 采用自上而下的处理顺序，即依次对语音、面部和动作模态进行建模，从而以层次化结构保持模态间的语义依存关系。

这种设计符合人类交流中“语言、表情、动作”一体化的认知规律^[4-5]。语音先规划语义与节奏，面部表情作为情绪强化信号随后产生，最终通过身体动作完成完整的非语言表达。模型中，语音编码器输出的时间嵌入被输入至面部编码器，再与面部特征融合后驱动动作解码器，从而保持语义一致性并增强表现力。

然而，CaMN 的原始设计面向离线整句生成任务，需要访问未来上下文以维持全局连贯性。在实时场景下，这种依赖将引入显著延迟并破坏因果性。FaceCapGes 在继承其层次思想的同时，对输入模式、训练方式与模态选择进行了系统性重构，以满足帧级实时约束。

3.2.2 说话人 ID 分支移除

如图 2.1 所示，用户配置层会设置说话人 ID 配置用于模型切换，但该模态在本模型中不属于网络输入。在基线模型 CaMN 中，输入模态包含显式的说话人 ID 向量，用于在同一模型内区分不同演讲者的风格差异。然而在实时交互场景下，该分支并非必要：用户身份通常固定，且说话风格的变化频率远低于帧级推理速度。因此，FaceCapGes 移除了 ID 输入分支，采用针对每个说话人独立训练模型参数的方法。实验表明，该方式能在保持收敛稳定的同时提升动作的自然性与节奏一致性。从系统使用角度看，不同模型可视为说话风格配置，用户仅在需要时切换对应参数，该操作发

生频率低，不会影响实时推理性能。

3.2.3 输入模态继承

语音特征通过时间卷积网络（Temporal Convolutional Network, TCN）和多层感知机（Multilayer Perceptron, MLP）编码，以捕捉短时节奏模式；面部模态采用相似结构，并在中间层融合语音嵌入，从而增强语音与表情之间的语义关联。语音编码器 E_A 与面部编码器 E_F 的输出定义为：

$$z_t^A = E_A(v_{t-N:t}^A), \quad z_t^F = E_F(v_{t-N:t}^F; z_t^A) \quad (3.2)$$

其中 $z_t^A \in \mathbb{R}^{128}$, $z_t^F \in \mathbb{R}^{32}$ 。这两个编码器负责提取低层次语音节奏与表情动态信息，为后续模态融合提供稳定上下文表征。

此外，系统在此基础上引入头部姿态模态 v_t^H ，用于补充空间方向与节奏信号。其编码器 E_H 将 Rot6D 表示的头部旋转型量映射为紧凑潜在表征：

$$z_t^H = E_H(v_{t-N:t}^H), \quad (3.3)$$

编码器结构将在第 3.3 节详细说明。

3.2.4 输出模态继承（身体姿态解码）

在输入模态经过编码与融合后，模型需将多模态特征映射至对应的身体姿态空间。为实现层次化的动作生成与结构协调，本文将上半身的输出区域划分为两个互补分支：躯干（Torso, T）与上肢（Upper limbs, U）。躯干部分包含脊椎的三个主要控制关节，用于确定身体的姿态基准与运动节奏；上肢部分包含双臂及手部关节，负责生成与语音节奏及情绪表达相呼应的细节动作。最终的上半身姿态表示为两者的组合：

$$v_t^B = v_t^T \otimes v_t^U, \quad (3.4)$$

其中 \otimes 表示通道维度拼接操作。

该分层设计继承了 CaMN 的层次预测思路：模型首先生成相对稳定的躯干姿态以确定整体方向，再以此为条件预测上肢动作，从而在实时生成中保持整体协调性与自然度。

融合输入的序列化表示 在时刻 t ，来自语音、面部与头部编码器的特征 z_t^A 、 z_t^F 、 z_t^H 与历史动作上下文共同构成 LSTM 解码器的输入。为保持严格因果性，我们显式提

供最近 N 帧的历史上半身姿态 $\mathbf{v}_{t-N:t-1}^B$ 作为条件信息，并对当前时刻的动作输入使用占位符进行对齐。具体而言，定义历史动作对齐序列 \mathbf{s}_τ 为：

$$\mathbf{s}_\tau = \begin{cases} \mathbf{v}_\tau^B, & \tau \leq t-1, \\ \mathbf{0}, & \tau = t, \end{cases} \quad \tau \in \{t-N, \dots, t\}, \quad (3.5)$$

则窗口内每一帧的融合输入向量可写为：

$$\mathbf{z}_\tau^{fuse} = \mathbf{z}_\tau^A \otimes \mathbf{z}_\tau^F \otimes \mathbf{z}_\tau^H \otimes \mathbf{s}_\tau, \quad \tau \in \{t-N, \dots, t\}, \quad (3.6)$$

并将其按时间维堆叠得到长度为 $N+1$ 的因果上下文输入序列：

$$\mathbf{Z}_t^{fuse} = (\mathbf{z}_{t-N}^{fuse}, \dots, \mathbf{z}_t^{fuse}). \quad (3.7)$$

躯干与上肢的级联解码 随后，融合输入序列 \mathbf{Z}_t^{fuse} 分别送入躯干与上肢两路单向 LSTM 解码器，得到窗口内的输出序列：

$$\mathbf{O}_t^T = \text{LSTM}_T(\mathbf{Z}_t^{fuse}), \quad \mathbf{O}_t^U = \text{LSTM}_U(\mathbf{Z}_t^{fuse}), \quad (3.8)$$

其中 $\mathbf{O}_t^T = (\mathbf{o}_{t-N}^T, \dots, \mathbf{o}_t^T)$, $\mathbf{O}_t^U = (\mathbf{o}_{t-N}^U, \dots, \mathbf{o}_t^U)$ 。由于本文在时刻 t 的目标是预测当前帧动作，我们仅取序列末端输出作为当前帧的潜在表征：

$$\mathbf{z}_t^T = \mathbf{o}_t^T, \quad \mathbf{z}_t^U = \mathbf{o}_t^U. \quad (3.9)$$

最后，通过两路独立的 MLP 模块将潜在表征还原为旋转参数：

$$\hat{\mathbf{v}}_t^T = \text{MLP}_T(\mathbf{z}_t^T), \quad \hat{\mathbf{v}}_t^U = \text{MLP}_U(\mathbf{z}_t^U), \quad (3.10)$$

并拼接得到当前帧的完整上半身动作预测：

$$\hat{\mathbf{v}}_t^B = \hat{\mathbf{v}}_t^T \otimes \hat{\mathbf{v}}_t^U. \quad (3.11)$$

上述内容描述了模型在一个因果上下文输入序列内完成对当前帧动作的解码过程。关于循环状态的传递方式、滑动窗口展开策略以及片段级优化目标，将在第4.1章与第4.3章中进一步说明。

3.3 头部姿态模态的引入

3.3.1 输入特征与表示

本文仅使用头部旋转信息作为输入特征，不引入头部平移位移。在 BEAT 数据集中，演讲者多为站姿录制，过程中存在一定幅度的身体移动与位移变化；而目标应用场景下的用户运动模式可能不同，例如在坐姿交互中身体位移通常较小。由于平移分量更易受到拍摄坐标系、相机距离与场景布置等因素影响，直接建模头部位移可能引入额外的场景依赖性，从而削弱跨场景泛化能力。因此，本文仅采用头部旋转作为输入，并使用 Rot6D^[14] 进行表示，以获得连续且无奇异性的旋转特征表达。

3.3.2 特征获取方法

如第 2.2 所述，头部姿态可与面部表情共同由面部捕捉工具（如 ARKit）实时提取。对于 BEAT 数据集，其原始数据中未单独提供头部姿态信号。因此，我们在训练预处理中利用骨架层级关系，沿骨架链路复合从根节点到头部关节的相对旋转，从而得到头部在全局坐标系下的绝对旋转表示，并进一步转换为 Rot6D 形式作为模型输入。

3.3.3 级联结构中的位置

在模型结构设计中，我们考察了头部姿态特征与其他模态的多种组合方式。具体而言，分别尝试了：(1) 将头部姿态特征在编码阶段与语音或面部特征进行早期融合；(2) 在解码阶段以前两者的嵌入结果为条件，预测头部姿态特征作为辅助信号。实验结果显示，这两种交互方式均未带来显著性能提升，部分设置甚至出现训练收敛速度下降或动作节奏轻微错位的情况。

这一现象与认知层面的规律相符。头部动作虽然与语音韵律在时间上存在同步性，但在认知层面并非由语音或表情直接驱动，也难以反向推导这些模态的动态变化。换言之，三者更可能属于并行协同关系，共享节奏与注意机制，但不构成单向的预测链。

基于此观察，本文在最终架构中采用弱耦合的后级输入设计：头部姿态特征在语音与面部特征编码完成后，以独立通道的形式拼接至多模态隐向量 z_t^{fuse} ，而非在编码阶段进行显式交互。该处理方式在保持整体结构简洁性的同时，仍保留头部姿态在方向、节奏及注意焦点方面的补充作用。

实验表明，在此配置下模型的整体自然度与时序稳定性得到改善，说明头部姿态虽非语音或表情的从属模态，但作为空间与节奏的辅助信号仍具有积极贡献。

编码器结构 图3.1所示为头部姿态编码器结构。该编码器由两层前馈网络组成，输入为Rot6D表示的6维向量：

$$\mathbf{z}_t^H = E_H(\mathbf{v}_{t-N:t}^H), \quad (3.12)$$

其中 E_H 的具体形式为：

$$\mathbf{h}_1 = \text{ReLU}(W_1 \mathbf{v}_t^H + b_1), \quad (3.13)$$

$$\mathbf{z}_t^H = W_2 \mathbf{h}_1 + b_2, \quad (3.14)$$

网络维度设置为：输入6，中间层36，输出12。在特征层面，其输出与语音、面部嵌入拼接后输入解码器，形成从语义到反应的多层次信号流。



图3.1 头部姿态编码器结构示意图

Figure 3.1 Architecture of the Head Pose Encoder

3.4 模型整体结构

图3.2展示了FaceCapGes从音频、面部、头部编码器分别提取模态特征后拼接，输入至LSTM解码器生成躯干与手部动作的过程。其中，LSTM的输出仅保留最后一帧作为当前时刻预测，符合帧级实时推理设定。实际训练中，训练阶段历史姿态序列比目标长度少一帧，需进行零填充进行对齐。

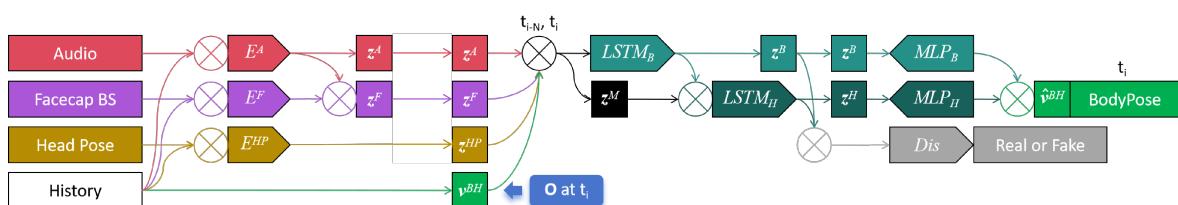


图3.2 FaceCapGes模型整体结构

Figure 3.2 Overall Architecture of FaceCapGes

3.5 本章小结

本章围绕FaceCapGes的模型结构设计，系统介绍了面向严格实时交互场景的多模态级联架构与关键输入输出模态设定。首先，本文在任务设定上明确了逐帧在线生

成与严格因果约束：模型在任意时刻仅利用当前及过去帧的语音、面部与头部信息进行动作预测，从而区别于依赖未来上下文的一次性离线生成方法，并为后续结构设计提供了统一前提。

在模型结构方面，本文继承 CaMN 的级联框架，将语音、面部表情与身体动作视为具有层次关联的表达信号，并进一步系统性地引入头部姿态作为独立输入模态，以补充空间方向与节奏信息。同时，本文仅使用头部旋转信息进行建模，并给出了适用于不同数据来源（面部捕捉系统与数据集骨架数据）的统一特征获取与表示方式，从而提升跨场景泛化能力与时序连续性。

在解码端，本章说明了面向上半身骨骼姿态的层次化输出设计：将动作划分为躯干与上肢两路分支，并通过级联解码保持整体协调性与自然度。上述模型结构为后续章节提出的滑动窗口自回归展开策略与片段级优化目标提供了清晰的结构基础。下一章将进一步介绍模型在训练与推理阶段的统一展开过程，并证明该策略在严格因果约束下能够实现稳定的实时动作生成。

第 4 章 基于滑动窗口的实时手势生成自回归训练

本章讨论在严格因果 (causal) 约束下, 如何将第 ?? 章所定义的窗口内一步预测模型, 展开为可用于训练与推理的逐帧自回归生成过程, 并据此构建片段级 (segment-level) 的监督与对抗训练目标。与离线生成不同, 实时生成场景中无法一次性观测未来的多模态输入 (如语音、面部与头部特征), 因此模型必须在每个时间步仅依赖历史信息进行下一帧动作预测。本章首先分析因果性对循环神经网络结构的约束, 并说明为何采用单向 LSTM 替代双向 LSTM (BiLSTM), 从而保证模型在推理阶段满足严格因果性。

在此基础上, 本章进一步引入滑动窗口与前置动作帧的概念, 描述片段切割训练、窗口展开生成以及自回归写回历史动作缓冲的统一流程, 并给出片段级损失计算与对抗训练策略。需要强调的是, 第 ?? 章所给出的融合特征构造与窗口内一步预测 (如式 (3.6) 与式 (3.8)) 描述了单个窗口内的前向计算; 而本章所提出的滑动窗口展开, 则决定了模型如何在片段内部逐帧生成并形成连续输出, 以及训练目标如何在片段尺度上进行定义。

4.1 单步因果 LSTM 预测器

为实现严格因果的实时手势生成, 本章首先建立一个用于“下一帧动作预测”的基本单元: 单步因果 LSTM 预测器。该预测器在每个时间步利用当前可用的多模态输入特征, 并结合历史动作上下文与循环状态, 对下一帧动作进行估计。在后续章节中, 我们将该单步预测器以滑动窗口方式展开, 从而形成对一个动作片段的自回归生成过程, 并据此定义片段级别的监督损失与对抗训练目标。

4.1.1 LSTM 基本概念与状态传递机制

手勢动作序列具有显著的时间依赖性: 当前姿态不仅受当前输入模态 (如语音、面部表情、头部姿态等) 影响, 也与过去的动作状态密切相关。循环神经网络 (Recurrent Neural Network, RNN) 通过引入随时间递推的隐状态来建模序列依赖, 而长短记忆网络 (Long Short-Term Memory, LSTM) 进一步通过门控机制缓解长序列训练中的梯度消失问题, 从而更适合用于动作序列建模。

在标准的 LSTM 结构中, 网络在每个时间步接收当前输入特征 \mathbf{x}_t , 并维护两类

递推状态：隐藏状态 \mathbf{h}_t 与记忆单元状态 \mathbf{c}_t 。其中， \mathbf{h}_t 可视为与当前输出相关的短期表示，而 \mathbf{c}_t 作为更稳定的记忆轨道，用于在更长时间尺度上保留信息。其递推过程可形式化表示为：

$$(\mathbf{h}_t, \mathbf{c}_t) = \text{LSTM}(\mathbf{x}_t, \mathbf{h}_{t-1}, \mathbf{c}_{t-1}). \quad (4.1)$$

上述表达式为逐时间步的递推形式。实际实现中，深度学习框架通常将该递推过程以序列形式批量展开，即将一段输入序列送入 LSTM，并由网络在内部迭代计算每一时刻的 $(\mathbf{h}_t, \mathbf{c}_t)$ 。

根据隐状态传播方向的不同，LSTM 主要分为单向 LSTM (Unidirectional LSTM) 与双向 LSTM (Bidirectional LSTM)。单向 LSTM 沿时间正向递推，时刻 t 的状态仅依赖于过去与当前输入，因此满足因果约束；其隐状态可写为 $\mathbf{h}_t = f(\mathbf{x}_{\leq t})$ 。相比之下，双向 LSTM 同时构建正向与反向的递推过程，其输出在时刻 t 会同时依赖过去与未来的输入信息，即 $\mathbf{h}_t = f(\mathbf{x}_{\leq t}, \mathbf{x}_{\geq t})$ 。该结构在离线场景下能够更充分利用上下文，但由于反向路径引入未来信息依赖，因此不满足严格因果的实时生成需求。

基于以上差异，下一节将进一步阐述本文选择单向 LSTM 作为动作生成骨干网络的原因，并说明在严格因果的自回归生成过程中，如何结合循环状态与显式历史上下文以提升预测稳定性。

4.1.2 选择单向 LSTM 的因果性动机

实时手势生成要求模型在时刻 t 只能依赖于截至当前时刻可观测的信息，即仅允许使用 $\{\mathbf{x}_\tau\}_{\tau \leq t}$ （以及过去生成的动作历史）来预测当前或下一帧动作。在此严格因果 (strict causality) 约束下，模型不得访问未来输入 $\{\mathbf{x}_\tau\}_{\tau > t}$ ，否则将导致训练阶段的信息泄露，并使推理阶段无法复现相同的条件分布。

双向 LSTM (Bidirectional LSTM, BiLSTM) 通过同时构建前向与反向递推路径，在输出时刻 t 的表示时会融合来自未来时间步的反向信息。该结构在离线分析或全序列可见的任务中能够更充分利用上下文，从而提升预测性能；然而在实时生成任务中，未来输入在时刻 t 尚不可用，反向路径的依赖无法满足。因此，若直接使用 BiLSTM 进行训练，会隐式假设推理时同样可访问未来输入，导致训练与推理条件不一致 (train-test mismatch)，并可能造成生成阶段性能明显退化或出现不稳定行为。

相比之下，单向 LSTM (Unidirectional LSTM, UniLSTM) 沿时间正向递推，其状态更新仅依赖于过去与当前输入，从结构上满足因果约束。对于下一帧动作生成问题，UniLSTM 能够自然地被解释为一个逐步更新的预测器：在每个时间步基于当前

可观测输入与历史状态输出下一帧动作估计，并将生成结果反馈至下一步，从而形成自回归（autoregressive）的实时生成流程。此外，单向递推的状态传递机制也使得模型能够在有限的输入上下文之外保留更长程的动态信息，为后续滑动窗口展开提供了必要的记忆能力。

基于以上原因，本文采用单向 LSTM 作为动作生成的时序建模骨干网络，以满足严格因果的实时生成需求。在后续的滑动窗口展开策略中（见第4.3节），该单步预测器将被逐帧迭代调用，从而在片段级别完成动作序列的自回归生成，并用于定义监督损失与对抗训练目标。

4.1.3 双通路记忆：循环状态与显式历史上下文

尽管单向 LSTM 通过递推状态 $(\mathbf{h}_t, \mathbf{c}_t)$ 在理论上可以保留长程历史信息，但在自回归的动作生成任务中，仅依赖循环状态往往会造成两个实际问题。第一，状态本身是对历史的压缩表征，短期细节可能因遗忘机制而被弱化；第二，在长序列的自回归展开过程中，生成误差会逐步反馈进入模型并影响状态更新，从而导致状态漂移（state drift）与累积误差，尤其在引入序列级对抗训练时，判别器可能进一步放大这种不稳定性。因此，若完全依赖循环状态作为唯一的历史信息通道，模型在实时生成中容易出现动作抖动、风格发散或长期一致性下降等现象。

为提升逐帧生成的稳定性，本文在循环状态之外显式引入固定长度的历史动作上下文作为短期条件信息。具体而言，设 N 为历史动作上下文长度（在第??章中已给出定义），在预测时刻 t 的动作时，我们显式提供最近 N 帧的动作历史序列 $\mathbf{v}_{t-N:t-1}^B$ 作为可观测输入，使模型在每一步预测中均能获得短期运动细节与局部连续性约束。为保持因果性，该历史动作序列不包含当前待预测时刻 t 的真实动作，而采用占位符进行对齐（例如以零向量或掩码符号表示），从而形成长度为 $N+1$ 的因果上下文序列。

在该设计下，模型的历史信息将通过两条互补路径传递：一方面，循环状态 (\mathbf{h}, \mathbf{c}) 作为长期记忆通道，用于编码超过 N 帧范围的更长程动态趋势、说话风格与运动节奏；另一方面，显式历史动作上下文作为短期精确条件，在每一步预测中直接提供最近 N 帧的局部运动信息，从而缓解状态漂移带来的不确定性，并提升自回归生成的鲁棒性。换言之，本文采用“短期显式上下文 + 长期隐式状态”的双时间尺度记忆结构，以在严格因果约束下平衡表达能力与生成稳定性。

需要注意的是，由于相邻时间步的显式上下文存在高度重叠，采用固定长度上下文进行逐帧预测会引入一定的重复计算开销。然而在本文设定中，上下文长度 N 较

小（本工作中 $N=16$ ），该开销在实时系统中是可接受的；同时，显式上下文的反复对齐能够有效稳定自回归展开过程，并与序列级对抗训练兼容。下一节将进一步给出窗口内预测过程的形式化表达，说明在每个时间步如何将序列输入与跨步循环状态结合，并取序列末端输出完成下一帧动作预测。

4.1.4 窗口内预测过程的形式化表达

本节给出单步因果预测器在时刻 t 的窗口内前向计算过程，以形式化说明本文如何在显式历史上下文与跨步循环状态的共同作用下完成下一帧动作预测。如第??章所述，本文采用多模态融合特征 \mathbf{z}_t^{fuse} 作为时序解码器输入，并在每一步预测中显式提供长度为 $N+1$ 的因果上下文序列：

$$\mathbf{Z}_t^{fuse} = (\mathbf{z}_{t-N}^{fuse}, \dots, \mathbf{z}_t^{fuse}), \quad (4.2)$$

其中历史动作序列仅包含 $\mathbf{v}_{t-N:t-1}^B$ ，并在末帧使用占位符对齐长度，以满足严格因果约束。

序列输入与跨步状态传递 对于躯干分支与上肢分支，本文分别维护一套单向 LSTM 的隐藏状态与记忆单元状态： $(\mathbf{h}_{t-1}^T, \mathbf{c}_{t-1}^T)$ 与 $(\mathbf{h}_{t-1}^U, \mathbf{c}_{t-1}^U)$ 。在时刻 t ，我们将上下文序列 \mathbf{Z}_t^{fuse} 输入对应的 LSTM 解码器，并以跨步传递的状态作为初始状态：

$$\mathbf{O}_t^T, (\mathbf{h}_t^T, \mathbf{c}_t^T) = \text{LSTM}_T(\mathbf{Z}_t^{fuse}, \mathbf{h}_{t-1}^T, \mathbf{c}_{t-1}^T), \quad (4.3)$$

$$\mathbf{O}_t^U, (\mathbf{h}_t^U, \mathbf{c}_t^U) = \text{LSTM}_U(\mathbf{Z}_t^{fuse}, \mathbf{h}_{t-1}^U, \mathbf{c}_{t-1}^U), \quad (4.4)$$

其中 $\mathbf{O}_t^T = (\mathbf{o}_{t-N}^T, \dots, \mathbf{o}_t^T)$ 与 $\mathbf{O}_t^U = (\mathbf{o}_{t-N}^U, \dots, \mathbf{o}_t^U)$ 表示解码器在窗口内每个时间步的输出序列。

末端输出用于当前帧预测 由于本文的目标是在时刻 t 预测当前帧动作，我们仅使用窗口末端对应的输出 \mathbf{o}_t^T 与 \mathbf{o}_t^U 作为当前帧的潜在表征：

$$\mathbf{z}_t^T = \mathbf{o}_t^T, \quad \mathbf{z}_t^U = \mathbf{o}_t^U. \quad (4.5)$$

该实现与主流深度学习框架中序列式 LSTM 接口一致：输入序列长度为 $N+1$ ，输出序列长度同为 $N+1$ ，而当前帧预测仅取最后一帧输出（即 $\mathbf{O}_t[\text{end}]$ ）。

动作解码与姿态拼接。 随后, \mathbf{z}_t^T 与 \mathbf{z}_t^U 分别通过独立的多层感知机 (MLP) 解码为旋转参数, 得到当前帧的躯干与上肢动作预测:

$$\hat{\mathbf{v}}_t^T = \text{MLP}_T(\mathbf{z}_t^T), \quad \hat{\mathbf{v}}_t^U = \text{MLP}_U(\mathbf{z}_t^U). \quad (4.6)$$

最终, 将两路输出拼接得到当前帧的完整上半身动作:

$$\hat{\mathbf{v}}_t^B = \hat{\mathbf{v}}_t^T \otimes \hat{\mathbf{v}}_t^U. \quad (4.7)$$

以上过程定义了本文单步因果预测器在每个时间步的计算单元: 其输入为长度 $N+1$ 的因果上下文序列与跨步循环状态, 其输出为当前帧动作 $\hat{\mathbf{v}}_t^B$ 与更新后的循环状态 $(\mathbf{h}_t, \mathbf{c}_t)$ 。在第4.3节中, 我们将进一步描述该单步预测器如何随时间滑动展开, 从而在片段尺度上生成长度为 M 的动作序列, 并用于定义监督损失与对抗训练目标。

4.2 训练片段构造: 基于 CaMN 的固定长度切割策略

为在严格因果约束下训练单步预测器并实现片段级的自回归生成, 本工作沿用 CaMN 提出的训练样本构造方式, 将长序列动作数据切割为固定长度的短片段 (segment) 作为训练样本。该策略具有两个优点: 一方面, 固定长度片段能够显式限制训练时的时间范围, 从而稳定模型的时序建模与优化过程; 另一方面, 片段化表示便于在片段内部进行滑动窗口展开, 使训练流程与实时推理阶段的逐帧生成方式保持一致。

4.2.1 固定长度片段定义

设原始动作序列为 $\{\mathbf{v}_t^B\}_{t=1}^T$, 以及对应的多模态输入特征序列 $\{\mathbf{z}_t^A, \mathbf{z}_t^F, \mathbf{z}_t^H\}_{t=1}^T$ 。我们从长序列中截取长度为 L 的连续片段作为训练样本, 其中片段长度由历史上下文长度 N 与片段内生成步数 M 共同决定:

$$L = N + M. \quad (4.8)$$

在本文设定中, 沿用 CaMN 的片段长度配置取 $L = 34$ 帧, 同时选取历史上下文长度 $N = 16$ 帧, 因此对应的片段内自回归生成步数为 $M = 18$ 帧。

对于第 k 个训练片段, 其时间范围为 $[s_k, s_k + L - 1]$, 片段内的动作与多模态输入分别表示为:

$$\mathbf{V}_k^B = (\mathbf{v}_{s_k}^B, \mathbf{v}_{s_k+1}^B, \dots, \mathbf{v}_{s_k+L-1}^B), \quad (4.9)$$

$$\mathbf{Z}_k = \left(\{\mathbf{z}_{s_k:t}^A\}_{t=s_k}^{s_k+L-1}, \{\mathbf{z}_{s_k:t}^F\}_{t=s_k}^{s_k+L-1}, \{\mathbf{z}_{s_k:t}^H\}_{t=s_k}^{s_k+L-1} \right). \quad (4.10)$$

其中 \mathbf{V}_k^B 将进一步划分为两部分：前 N 帧作为片段的历史上下文（亦即前置动作帧，用于缓冲与提供因果条件），后 M 帧为片段内需要逐帧生成并参与训练目标计算的部分。该划分将于第4.3节中用于描述滑动窗口展开过程与片段级损失计算策略。

4.2.2 重叠切割与样本覆盖

为提高训练样本覆盖率并增强模型对不同对齐位置的鲁棒性，我们采用带重叠的滑动切割方式从长序列中提取片段。具体而言，片段起始位置 s_k 按固定步长 Δ 滑动：

$$s_k = 1 + (k - 1)\Delta, \quad (4.11)$$

从而得到一组相互重叠的训练片段 $\{\mathbf{V}_k^B, \mathbf{Z}_k\}$ 。在本文实现中，沿用 CaMN 的设置取 $\Delta = 10$ 帧，以在样本数量与数据冗余之间取得平衡。

需要强调的是，固定长度片段仅用于训练样本组织与片段级展开，并不意味着模型在推理阶段以整段方式生成输出。相反，模型在推理与训练中均以单步预测器形式逐帧运行，并在片段内部通过滑动窗口展开形成长度为 M 的生成序列。下一节将进一步给出片段内部的滑动窗口展开机制，并说明如何将单步预测结果拼接为片段级输出。

4.3 片段内部的滑动窗口展开策略

上一节定义了训练样本以固定长度片段组织：每个片段长度为 $L = N + M$ ，其中前 N 帧为历史上下文，后 M 帧为需要逐帧生成的目标区间。本节进一步描述片段内部的滑动窗口展开（sliding-window rollout）策略：该策略在训练与推理阶段保持一致，以单步因果预测器（第4.1节）为基本计算单元，逐帧生成长度为 M 的动作序列并拼接得到片段级输出。在本文设定中， $L = 34$ ， $N = 16$ ，因此 $M = 18$ ，窗口长度为 $N+1 = 17$ 。

4.3.1 前置动作帧与 Warm-up 阶段

在严格因果的实时生成中，模型在时刻 t 预测动作时不能访问未来输入，并且其输出需要在片段之间保持连续。为此，我们将每个片段的前 N 帧动作视为前置动作帧（prefix motion frames），其作用是为后续生成提供短期运动上下文，并避免片段边界处出现突兀断裂。需要强调的是，前置动作帧属于可观测条件信息而非生成目标：在训练阶段它们直接来自真实动作序列，在推理阶段则来自系统缓存的历史动作（或

上一片段的生成结果)。

在片段开始时刻 s_k , 我们首先执行 Warm-up 阶段: 将片段的前 N 帧真实动作 $\{\mathbf{v}_{s_k}, \dots, \mathbf{v}_{s_k+N-1}\}$ 写入历史动作缓存 \mathcal{H} , 并同步读取对应的多模态输入特征。与常见的逐帧 RNN 初始化不同, 本文在 Warm-up 阶段不执行任何前向计算, 即不更新 LSTM 状态, 仅完成历史动作上下文的缓冲准备。该设计使得模型的生成阶段从一个明确的历史缓冲状态开始, 并避免在 Warm-up 阶段引入不必要的计算与误差传播。

由于 Warm-up 阶段不执行前向计算, 生成阶段的初始 LSTM 状态由预设初始化值给出。本文采用零初始化作为 $(\mathbf{h}_{t-1}, \mathbf{c}_{t-1})$ 的初值。

4.3.2 滑动窗口展开与逐帧自回归生成

在 Warm-up 阶段结束后, 我们进入生成阶段: 模型在片段内部进行 M 步滑动窗口展开, 每一步生成 1 帧动作, 并将预测结果写回历史动作缓存以供下一步使用 (autoregressive feedback)。设片段内生成阶段的第 m 步对应全局时间 $t = s_k + N + m$ (其中 $m = 0, 1, \dots, M - 1$), 历史动作缓存 \mathcal{H}_{t-1} 包含最近 N 帧可用动作序列:

$$\mathcal{H}_{t-1} = (\tilde{\mathbf{v}}_{t-N}^B, \dots, \tilde{\mathbf{v}}_{t-1}^B), \quad (4.12)$$

其中 $\tilde{\mathbf{v}}$ 表示当前可用的动作帧: 在生成开始时它包含真实前置动作帧, 在生成推进过程中则逐渐由模型预测结果覆盖。

在时间步 t , 我们构造长度为 $N+1$ 的因果上下文窗口, 组合当前可观测的多模态输入特征与历史动作缓存, 形成融合输入序列 \mathbf{Z}_t^{fuse} (定义见式 (4.2)):

$$\mathbf{Z}_t^{fuse} = (z_{t-N}^{fuse}, \dots, z_t^{fuse}), \quad (4.13)$$

其中历史动作序列部分采用 $(\tilde{\mathbf{v}}_{t-N}^B, \dots, \tilde{\mathbf{v}}_{t-1}^B, \mathbf{0})$ 进行对齐, 末帧使用占位符以保证严格因果性。

随后, 单步因果预测器以 \mathbf{Z}_t^{fuse} 与跨步 LSTM 状态作为输入, 输出当前帧动作预测 $\hat{\mathbf{v}}_t^B$ 并更新状态 (窗口内形式化表达见第4.1.4节):

$$\hat{\mathbf{v}}_t^B, (\mathbf{h}_t, \mathbf{c}_t) = f_\theta(\mathbf{Z}_t^{fuse}, \mathbf{h}_{t-1}, \mathbf{c}_{t-1}). \quad (4.14)$$

生成结果 $\hat{\mathbf{v}}_t^B$ 会被写回历史动作缓存, 从而更新 \mathcal{H}_t 并用于下一时间步预测:

$$\mathcal{H}_t = (\tilde{\mathbf{v}}_{t-N+1}^B, \dots, \tilde{\mathbf{v}}_{t-1}^B, \hat{\mathbf{v}}_t^B). \quad (4.15)$$

通过上述逐帧自回归展开, 我们得到片段内生成区间的预测序列:

$$\hat{\mathbf{V}}_k^{gen} = (\hat{\mathbf{v}}_{s_k+N}^B, \hat{\mathbf{v}}_{s_k+N+1}^B, \dots, \hat{\mathbf{v}}_{s_k+L-1}^B), \quad (4.16)$$

其长度为 M , 对应片段的后 M 帧。

4.3.3 拼接生成序列与片段级输出

由于滑动窗口展开在每一步仅生成 1 帧动作, 片段级输出通过对 M 步预测结果进行时间维拼接获得。片段级生成结果 $\hat{\mathbf{V}}_k^{gen}$ 与真实动作片段 \mathbf{V}_k^B 在时间上对齐, 其中前 N 帧为可观测上下文, 后 M 帧为模型生成输出。因此, 后续训练目标的定义将以 $\hat{\mathbf{V}}_k^{gen}$ 为核心对象, 并与真实片段中对应的生成区间进行比较。下一节将进一步给出片段级监督损失与对抗训练目标的定义, 并说明为何在损失计算与判别器输入中需要移除前置动作帧的部分。

4.4 片段级训练目标: 监督损失与时序平滑约束

在训练阶段, 给定来自多模态语音动作数据集 (如 BEAT) 的配对样本序列:

$$(\mathbf{z}_t^A, \mathbf{z}_t^F, \mathbf{z}_t^H, \mathbf{v}_t^B), \quad (4.17)$$

模型的学习目标是在严格因果约束下生成自然、流畅且与语音节奏相匹配的上半身动作序列。与传统离线序列预测不同, 本文采用第4.3节所述的滑动窗口展开策略: 模型在每个片段内部自回归地逐帧生成 M 帧动作, 并将其拼接为片段级生成序列。

4.4.1 片段级生成序列与损失计算范围

对于第 k 个训练片段, 其长度为 $L = N + M$ (第4.2节), 其中前 N 帧为前置动作帧 (prefix frames), 后 M 帧为模型需要生成的目标区间。根据滑动窗口展开过程 (式(4.16)), 模型得到片段生成区间的预测序列:

$$\hat{\mathbf{g}}_k = (\hat{\mathbf{v}}_{s_k+N}^B, \hat{\mathbf{v}}_{s_k+N+1}^B, \dots, \hat{\mathbf{v}}_{s_k+L-1}^B) \in \mathbb{R}^{M \times d}, \quad (4.18)$$

其中 d 表示动作表示的维度 (例如 6D 旋转参数的拼接维度)。对应的真实动作序列为:

$$\mathbf{g}_k = (\mathbf{v}_{s_k+N}^B, \mathbf{v}_{s_k+N+1}^B, \dots, \mathbf{v}_{s_k+L-1}^B) \in \mathbb{R}^{M \times d}. \quad (4.19)$$

需要强调的是, 片段前 N 帧前置动作仅用于提供因果历史上下文与片段平滑过渡, 其本身并非生成目标。因此, 本文的监督损失仅在片段生成区间 $\{s_k + N, \dots, s_k + L - 1\}$ 上计算, 不对前置动作帧施加重构约束。该策略与实际推理阶段一致: 前置动作帧作为已知条件用于初始化历史缓存, 而模型输出仅对应后续的生成区间。

4.4.2 总体优化目标

综合考虑空间重构精度、时序平滑性以及动作分布一致性，本文的总体优化目标定义为：

$$\mathcal{L}_{total} = \lambda_r \mathcal{L}_{rec} + \lambda_v \mathcal{L}_{vel} + \lambda_a \mathcal{L}_{acc} + \lambda_{adv} \mathcal{L}_{adv}, \quad (4.20)$$

其中 \mathcal{L}_{rec} 衡量片段生成区间的姿态重构误差， \mathcal{L}_{vel} 与 \mathcal{L}_{acc} 分别约束速度与加速度的连续性， \mathcal{L}_{adv} 表示对抗训练损失，将在第4.5节中进一步介绍。该组合设计在自回归展开过程中能够有效缓解高频抖动与速度漂移问题，并提升生成序列的整体动态一致性。

4.4.3 姿态重构与时序平滑损失

为同时保证空间重构精度与时间连续性，我们采用基于 Huber 误差的重构损失形式，并分别作用于姿态、速度与加速度信号。给定任意预测序列 $\hat{\mathbf{x}}$ 及其对应的真实序列 \mathbf{x} ，基础误差项定义为：

$$\mathcal{L}_{Huber}(\mathbf{x}, \hat{\mathbf{x}}) = \beta \cdot \text{SmoothL1}\left(\frac{\mathbf{x}}{\beta}, \frac{\hat{\mathbf{x}}}{\beta}\right), \quad (4.21)$$

其中 SmoothL1(\cdot) 表示平滑 L1 误差， β 为平滑系数，本文中设为 0.1。

在此基础上，片段生成区间的姿态、速度与加速度损失分别定义为：

$$\mathcal{L}_{rec} = \mathcal{L}_{Huber}(\mathbf{g}_k, \hat{\mathbf{g}}_k), \quad (4.22)$$

$$\mathcal{L}_{vel} = \mathcal{L}_{Huber}(\mathbf{g}'_k, \hat{\mathbf{g}}'_k), \quad (4.23)$$

$$\mathcal{L}_{acc} = \mathcal{L}_{Huber}(\mathbf{g}''_k, \hat{\mathbf{g}}''_k), \quad (4.24)$$

其中一阶与二阶时间差分 \mathbf{g}'_k 、 \mathbf{g}''_k 定义为：

$$\mathbf{g}'_{k,t} = \mathbf{g}_{k,t} - \mathbf{g}_{k,t-1}, \quad \mathbf{g}''_{k,t} = \mathbf{g}'_{k,t} - \mathbf{g}'_{k,t-1}, \quad (4.25)$$

预测序列 $\hat{\mathbf{g}}'_k$ 、 $\hat{\mathbf{g}}''_k$ 同理定义。

上述多尺度重构约束在自回归预测过程中能够有效缓解高频抖动与速度漂移问题，在保证运动学精度的同时提升生成序列的时间稳定性。需要注意的是，这些损失均作用于片段生成区间的拼接结果 $\hat{\mathbf{g}}_k$ ，而非窗口内部的中间输出；从而使损失计算与实际代码实现保持一致，并避免对重叠窗口重复计入误差。

4.4.4 对抗损失

为进一步提升生成动作的自然度，本文引入基于判别器的对抗训练损失 \mathcal{L}_{adv} 。判别器以序列为输入，从整体动力学分布层面约束生成结果，使其在节奏、加速度与能

量变化等统计特性上与真实表演者保持一致。对抗训练的具体形式、判别器结构以及输入序列的掩码策略（包括移除前置动作帧）将在第4.5节中详细说明。

4.4.5 损失权重设置

各损失项的权重系数在实验中设定为 $\lambda_r = 5 \times 10^2$, $\lambda_v = 10^3$, $\lambda_a = 10^3$, $\lambda_{adv} = 10^{-1}$ 。

4.5 对抗训练：序列级判别器与前置动作掩码

尽管第4.4节的监督损失能够约束生成序列在逐帧空间误差与局部平滑性上的一致性，但仅依赖点对点重构目标往往难以完全刻画真实动作序列的整体动力学分布。为进一步提升生成动作的自然度与分布一致性，本文引入序列级判别器（discriminator），并采用基于二元交叉熵（Binary Cross Entropy, BCE）的对抗训练策略，在片段尺度上约束生成序列与真实序列的统计特性一致。

4.5.1 判别器输入：基于拼接片段的序列级判别

与窗口内部的中间输出不同，本文的判别器直接以片段生成区间的拼接序列为输入。对于第 k 个片段，生成器输出的预测序列 $\hat{\mathbf{g}}_k$ 及其对应的真实序列 \mathbf{g}_k 定义见式(4.18)与式(4.19)，二者均为长度 M 的序列。判别器 $Dis(\cdot)$ 接收一段动作序列，并输出其来自真实数据分布的概率：

$$Dis(\mathbf{g}) \in (0, 1). \quad (4.26)$$

通过序列级输入，判别器能够从整体动力学角度判断生成动作的真实感，从而在节奏、能量变化与运动统计特性等层面提供补充监督信号。

前置动作帧的掩码策略 需要强调的是，本文在对抗训练中同样不将片段前 N 帧前置动作输入判别器。前置动作帧属于可观测上下文条件，其内容在训练阶段为真实动作，在推理阶段为历史缓存或上一片段输出；它们并非模型需要生成的目标。若将其与生成区间一并输入判别器（如 CaMN 的做法），判别器将可能利用“前置帧必然真实”这一捷径特征进行判别，从而弱化对生成区间的监督作用，并将对抗梯度错误地分配到已知上下文上。因此，本文仅对生成区间 $\hat{\mathbf{g}}_k$ 与 \mathbf{g}_k 进行对抗判别，使对抗目标严格作用于模型实际生成的部分，并与第4.4节的监督损失范围保持一致。

4.5.2 基于 BCE 的对抗损失定义

判别器的训练目标是区分真实序列与生成序列，其损失定义为：

$$\mathcal{L}_D = -\mathbb{E}_{g_k} [\log Dis(g_k)] - \mathbb{E}_{\hat{g}_k} [\log (1 - Dis(\hat{g}_k))]. \quad (4.27)$$

生成器则希望其输出被判别器判定为真实，从而对应的对抗损失为：

$$\mathcal{L}_{adv} = -\mathbb{E}_{\hat{g}_k} [\log Dis(\hat{g}_k)]. \quad (4.28)$$

其中 \mathbb{E}_{g_k} 与 $\mathbb{E}_{\hat{g}_k}$ 分别表示对真实序列与生成序列的采样期望。该对抗目标从分布层面鼓励生成序列在整体运动统计特性上接近真实数据，从而补充监督损失在局部误差上的约束。

4.5.3 交替优化策略

在训练过程中，本文采用交替优化（alternating optimization）方式更新生成器与判别器。具体而言，对于每个训练批次，我们首先固定生成器参数 θ ，最小化式 (4.27) 更新判别器参数；随后固定判别器参数，最小化总体损失 \mathcal{L}_{total} （式 (4.20)）更新生成器参数，其中对抗项 \mathcal{L}_{adv} 由式 (4.28) 给出。通过上述训练方式，判别器不断提升对真实与生成序列的区分能力，而生成器则在监督约束与分布约束的共同作用下，逐步生成更加自然且时间一致的动作序列。

4.6 本章小结

本章围绕严格因果的实时手势生成任务，系统阐述了本文的滑动窗口训练与推理策略，并给出了与之配套的片段级优化目标。首先，我们建立了单步因果预测器作为基本计算单元：采用单向 LSTM 以满足严格因果约束，并在每个预测步中结合跨步循环状态与显式历史动作上下文，实现“短期显式条件 + 长期隐式记忆”的双时间尺度建模，从而提升自回归生成过程的稳定性与鲁棒性。随后，我们沿用 CaMN^[3]的固定长度片段切割方法将长序列组织为训练样本，并在片段内部执行滑动窗口展开：通过预热阶段缓冲前置动作帧，再在生成阶段逐帧自回归预测并将输出写回历史缓存，最终拼接生成的所有动作帧得到生成序列。

在训练目标方面，本章将监督损失定义在片段生成区间的拼接输出上，采用姿态、速度与加速度的多尺度 Huber 约束以缓解抖动与速度漂移，并进一步引入序列级判别器以提供分布一致性的对抗监督。与基线模型 CaMN 不同，本文将前置动作帧

视为纯条件上下文而非生成目标，在监督损失与对抗训练中均显式移除其影响，使训练目标严格作用于模型实际生成的区间，从而与实时推理阶段的因果生成流程保持一致。

第5章 实验结果与分析

为评估 FaceCapGes 的生成质量与实时性能，本文从用户评估、客观指标测量与推理效率三个方面展开实验，并在生成质量上与代表性方法进行对比分析。

5.1 训练配置

本文模型 FaceCapGes 基于 PyTorch 实现，所有实验在单张 NVIDIA RTX 4090 GPU 上进行。

本文基于 BEAT 数据集^[3] 进行训练与评估。该数据集包含多模态同步的语音、面部 blendshape 与全身动作信息，以 15 fps 记录多位专业表演者的演讲片段，覆盖多种语义与情绪场景。其标准骨架结构如图 5.1 所示，BEAT 数据集共包含人体中的 47 个关节节点，包括上肢及躯干的三个主要控制点（蓝色区域所示）。下肢关节则保持静态。



图 5.1 BEAT 数据集的骨架拓扑结构与驱动范围
Figure 5.1 Skeleton Topology and Actuated Joint Range in the BEAT Dataset

本文选取表演者 ID 2、4、6、8 的数据进行训练与测试，其中 2、4 为男性，6、8 为女性，确保在性别与说话风格上的分布均衡。训练集与测试集均包含相同的表演者，但使用不同的演讲片段，在预处理阶段已进行严格划分以避免片段交叉。

训练配置 训练时输入窗口的前序帧数设为 $N = 16$ ，预测长度为 $M = 34$ ，训练片段的切割步长为 10 帧。相邻片段因此存在部分重叠，从而在保证充分上下文信息的同时提升数据覆盖率与时间连续性。批大小设为 256。

优化器采用随机梯度下降 (Stochastic Gradient Descent, SGD)^[38]。基础学习率设置为 $\text{lr}_{base} = 2.5 \times 10^{-4}$, 并根据批大小按线性规则缩放为

$$\text{lr}_g = \text{lr}_{base} \cdot \frac{\text{batch_size}}{128}, \quad (5.1)$$

其中 lr_g 为生成器 (手势生成网络) 的实际学习率。

在对抗训练阶段, 判别器使用相同类型的 SGD 优化器, 其学习率按权重系数 w_d 缩放为

$$\text{lr}_d = w_d \cdot \text{lr}_g, \quad (5.2)$$

本文中设定 $w_d = 0.2$ 。

为防止早期训练阶段的不稳定, 对抗项在第 10 个 epoch 后引入, 即前 10 个 epoch 仅优化重构与时序平滑损失, 从第 11 个 epoch 起加入判别器并交替优化生成器与判别器参数。整体训练共 374 个 epoch。

在损失计算中, 前 N 帧的历史窗口仅作为因果上下文输入, 不参与重构与对抗项的误差回传。损失函数采用第 ?? 节所述的复合重构与对抗目标。

姿态表示 所有身体动作均转换为连续可微的 Rot6D 表示, 使用 EMAGE^[17] 中的实现方法, 以避免欧拉角奇异性与四元数的符号不确定性。

运行性能 在实时推理阶段, FaceCapGes 以 15 FPS 的速度驱动虚拟角色。

5.2 实验配置

5.2.1 实验对比模型

我们选取 FaceCapGes 的基线模型 CaMN^[3], 以及扩散模型方法中具有代表性的 DiffSHEG^[15] 作为对比模型。

表 5.1 总结了各模型的输入输出模态特征。其中, * 表示模型结构未显式输入说话人 ID, 采用每位说话人独立训练的设置, 说话人 ID 通过模型参数指定。

值得注意的是, FaceCapGes 是唯一满足严格因果约束的在线模型, 因此在评估时采用逐帧推理并将输出拼接为完整序列, 以模拟实时输入流。

5.2.2 跨模型评估设置

本章所有实验均基于 BEAT 数据集进行, 各模型使用一致的骨架拓扑与姿态表示, 从而保证输出格式可直接对齐与比较。其中 CaMN 与 DiffSHEG 使用其官方公开

表 5.1 对比模型的输入输出模态
Table 5.1 Comparison of Input/Output Modalities of Evaluated Methods

模型	输入模态					未来信息	输出
	音频	面部捕捉	头部姿态	说话人 ID	情绪		
CaMN	✓	✓	✗	✓	✓	✓	身体
DiffSHEG	✓	✗	✗	✓	✗	✓	身体 + 面部
本模型	✓	✓	✓	*	✗	✗	身体

的预训练模型；FaceCapGes 则在相同的数据预处理与骨架设定下由本文训练得到。

对于离线模型（CaMN 与 DiffSHEG），我们将整段演讲作为输入并一次性生成完整动作序列；对于在线模型（FaceCapGes），我们在关闭批处理（Batch=1）的条件下模拟逐帧输入流，并将逐帧输出按时间顺序拼接得到最终序列，以还原实时交互场景下的运行状态。

5.3 用户评估

为验证模型在交互环境中的表现，本文进行了用户主观评估实验，比较 FaceCapGes、CaMN^[3]、DiffSHEG^[15]三个模型在动作自然性、同步性与多样性方面的主观质量。本节首先介绍用户评估系统与实验配置，随后报告主观评价结果与分析。

5.3.1 用户评估系统与实验配置

实验材料与呈现方式 用户评估所使用的手势动画均基于 BEAT 数据集中的测试集语音片段生成，并以 Biovision Hierarchy (BVH) 文件形式保存。BVH 是一种通用的动作捕捉数据格式，通过层级化定义骨骼结构与帧级旋转参数，可直接导入 3D 动画与虚拟人系统。

本文使用的 BVH 文件采用欧拉角旋转表示。由于三种对比模型的输出旋转参数形式不同，因此在导出 BVH 之前，需将输出姿态统一转换为欧拉角表示，以便在后续动画播放中使用统一渲染流程。

本系统当前支持的虚拟人三维模型，需同时具备骨骼绑定，和与 ARKit^[21]兼容的 BlendShape 参数。基于此约束，本实验采用 BEAT 数据集提供的公开的男女两名演讲者三维模型，二者均满足兼容要求，可实现身体与面部的联合驱动。此外，经过 Unity^[39]的 Mecanim^[40]自动骨骼绑定系统匹配，可自动配对模型生成的 BVH 文件中定义的骨骼层级与虚拟人模型的骨骼节点，从而在不依赖手动权重绘制的情况下完

成动作重定向。

播放系统实现 我们基于 Unity 自行编写播放脚本，将各模型生成的 BVH 动画用于驱动虚拟人身体骨骼，同时以面部捕捉序列驱动 BlendShape 表情参数，并同步播放原始语音音频。系统支持同时呈现三种模型生成的动画结果：用户可在同一画面中（左、中、右）并行观察三种手势表现，所有语音与面部表情完全一致，唯一变量为身体动作。该设计使参与者能够直接比较不同模型在动作风格、节奏响应与语音同步性方面的差异。

为确保主观评价的公正性与可重复性，系统在每次实验开始前会随机分配三种模型的位置（左、中、右），界面上不会显示模型名称，从而避免潜在偏向。各测试片段的播放顺序在实验前统一设定，以保证不同参与者之间的样本顺序均衡。实验员在播放系统后台记录当前序列与模型对应关系，以便后续结果统计。

实验界面与设备 用户评估系统提供桌面端与虚拟现实（VR）端两种版本，功能完全一致。VR 版本基于 PICO 设备^[41]实现；桌面版支持多窗口并行播放，方便用户同时对比。如图 5.2 所示，播放界面在两种设备上保持统一布局，播放完成后参与者需通过交互界面对三个模型进行排序打分。

VR 用户在沉浸式环境中逐一观看三段动画；桌面端用户则可在单屏上同时观察全部模型。因此前者注重细节感知与临场性，后者更有利于整体风格与节奏的一致性对比。

实验流程与指导 实验正式开始前，研究人员向参与者说明了三项主观评价标准的含义，确保所有被试对评分维度理解一致：

- **真实性：**整体动作是否自然流畅，是否存在明显的违和感，如朝向异常或突然抖动；
- **同步性：**手势动作与语调、语音节奏是否协调一致；
- **多样性：**手势是否丰富多变，避免长时间静止或重复单一动作。

在实验过程中，VR 版本于线下环境进行，桌面版通过线上远程环境执行。两种形式均保持实时交流通道，研究人员可在参与者提问时即时解释操作或澄清评分标准。在正式评估阶段，参与者可多次重播当前片段，但不能返回查看先前内容，以减少记忆偏差。所有播放条件（Unity 场景内的相机角度、光照参数、音量与分辨率设置）在全部被试环境中保持一致，以确保渲染输出的可比性。

需要说明的是，对于 VR 实验，所有测试均在相同的线下实验室环境中进行，使用同一套 PICO 设备与照明条件；而桌面端实验通过远程方式执行，参与者在各自电脑上运行实验程序。研究人员可通过实时屏幕共享观察其操作流程并保持语音沟通，但无法严格控制其所在房间的光照或环境噪声条件。因此，桌面端实验在观看环境上存在一定差异，但由于任务内容与播放系统完全相同，且实验员在测试中持续指导，可认为该差异对结果的总体影响有限。

实验材料与任务设计 评估样本来自 BEAT 数据集中四位演讲者（ID: 2、4、6、8），其中 2、4 为男性，6、8 为女性。每位演讲者各选取两段平均长度约 1 分钟的语音片段，演讲话题互不重复，共组成 8 段固定视频样本。所有实验均使用相同的 8 段样本，但其呈现顺序在不同被试间经过随机化或平衡化处理，以避免顺序效应。每段视频均包含三种模型生成的动作版本（FaceCapGes、CaMN、DiffSHEG），并在播放时随机分配每个模型的动画在屏幕中的排序。参与者在观看每一片演讲音频后，根据三项主观标准（真实性、同步性、多样性）对三个模型的手势动画表现进行排名评估。

图 5.2 中为用户评估工具的实机界面。画面中共有 3 个虚拟人模型水平分布，在每一片演讲音频播放时，将 3 个生成模型的动画随机分配给 3 个虚拟人的身体骨骼。



图 5.2 用户评估工具实机界面
Figure 5.2 User Study Interface on the Evaluation Device

实验参与者 本实验共邀请 16 名参与者（12 名使用 VR 设备，4 名使用桌面端），涵盖不同性别。所有参与者在实验前均接受了操作说明与校准，并在系统指导下完成评分练习。

为避免呈现顺序对主观印象造成偏差，另设计了采用平衡拉丁方（Balanced Latin Square）顺序的实验版本，使不同参与者观看样本的顺序均衡分布。该版本实验共招募 8 位 VR 用户（4 男 4 女），排序顺序由 HCI 用户评估工具包^[42]自动生成，确保模型与演讲者组合的呈现顺序在全体被试间均匀分布。所有条件保持一致，唯一变量为视频播放顺序。

实验环境说明 为全面验证模型在不同交互场景下的表现稳定性，本次主观评估设置了桌面端与 VR 端两种实验环境，确保覆盖常规屏幕交互与沉浸式交互两类典型应用场景，具体环境配置如下：

- **桌面端环境：** 参与者通过个人电脑或实验室台式机进行评估，实验界面采用三窗口并行布局，参与者可同时观察左、中、右三个区域的虚拟人动作，聚焦于整体动作风格、节奏同步性的直观对比，注意力分布于整个屏幕的动作全局表现。
- **VR 端环境：** 基于 PICO VR 设备^[41]搭建沉浸式评估场景，参与者佩戴 VR 头显后进入虚拟观测空间，虚拟人以 1:1 比例呈现在眼前，观看距离模拟真实人际交流（约 1.2m）。该环境下参与者注意力更易聚焦于虚拟人上半身细节动作，对空间一致性、动作协调感的感知更敏锐。

两种环境的实验流程、评估指标定义及测试样本完全一致，仅通过设备差异构建不同的观察视角与注意力聚焦模式，以验证模型表现的跨设备适配性。

5.3.2 结果与分析

本节综合分析两轮用户评估的统计结果与参与者反馈。所有结果基于 BEAT 测试集中 4 位演讲者（ID:2、4、6、8；2 男 2 女）各 2 段语音片段，共 8 段固定样本。

总体测评结果 如图 5.3 所示，在 16 名参与者的总体评价中，FaceCapGes 在三个维度（真实性、同步性、多样性）上均优于基线模型 CaMN，并在“真实性”维度上略优于离线扩散模型 DiffSHEG。这一结果表明，FaceCapGes 虽在严格的实时因果约束下运行，但仍能保持与非实时生成模型相近的动作自然度与流畅性。

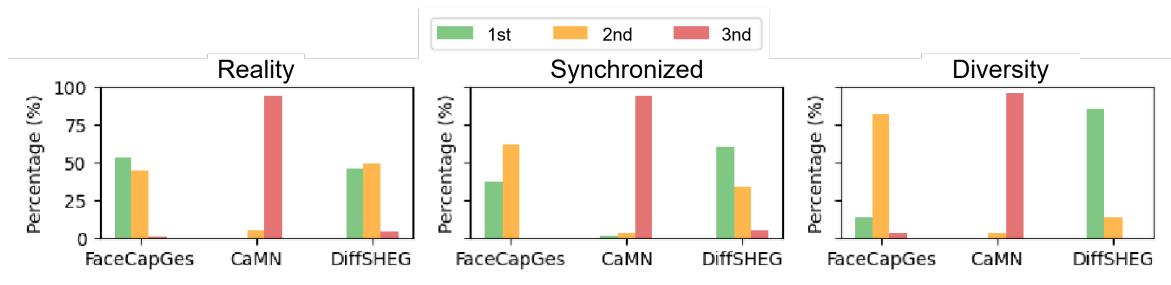


图 5.3 用户评估总体主观排名结果
Figure 5.3 Overall Subjective Ranking Results

平衡拉丁方实验结果 图 5.4 展示了平衡拉丁方设置下 8 位 VR 用户的独立结果，该版本严格控制了模型与演讲者组合的呈现顺序。结果与总体趋势一致，且 FaceCapGes 在“真实性”与“同步性”得到了更好的评价。这表明实验结论在不同顺序条件下保持稳定，进一步验证了模型主观评价结果的鲁棒性。

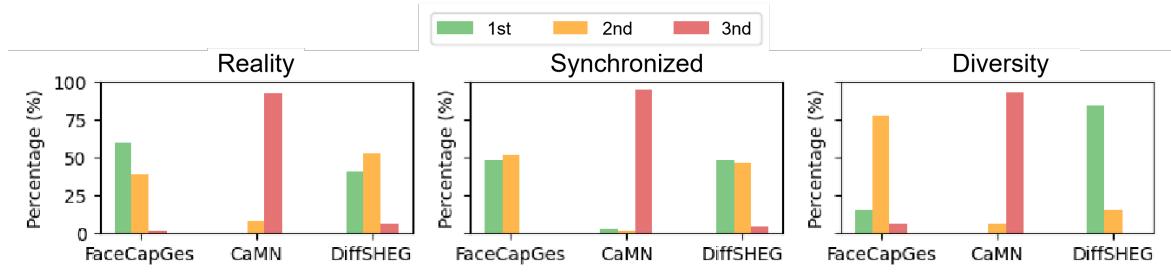


图 5.4 平衡拉丁方设置下的主观排名结果
Figure 5.4 Subjective Ranking Results under Balanced Latin Square Design

用户反馈分析 根据实验后访谈汇总，参与者普遍认为 FaceCapGes 的动作过渡自然、节奏感强，手势响应与语音重音、语调变化更为一致。我们认为，FaceCapGes 融入的头部姿态信息使手势与身体朝向更贴近真实动作，可能是其获得较高真实性评价的重要因素之一。

相比之下，CaMN 在头部与上身动作衔接处常出现僵硬或转向延迟的现象，且头部朝向容易偏离听众方向，从而影响了整体自然度与同步性评分。

对于 DiffSHEG，多数参与者提到其动作丰富度较高，并倾向于将其视为最具多样性的模型。这可能与 DiffSHEG 在生成过程中依赖完整文本输入有关：文本语义边界为动作变化提供了更明确的触发信号，使其更容易生成幅度更大、变化更频繁的动作模式。相比之下，FaceCapGes 主要基于实时语音流与感知模态驱动，缺乏显式语义解析，因此更难实现细粒度的语义级动作响应。

不过，不少参与者也提到 DiffSHEG 在部分片段中存在短暂的手部摆动过快或突然抖动的问题，从而降低了其真实性和同步性的评分。该现象可能与 Axis-Angle 旋转表示在数值空间中存在非连续点或奇异性有关，从而在训练或推理过程中更容易诱发关节角度的突变与抖动。相比之下，本文模型采用 Rot6D^[14] 旋转表示，该表示具有更好的连续性与数值稳定性，有助于缓解由旋转表示引入的突变问题。在本次用户研究中，FaceCapGes 未出现明显的抖动相关反馈，生成动作在关节转动与姿态过渡上保持较为连贯自然。

结果讨论 FaceCapGes 的因果式时间建模与头部姿态融合策略有效提升了局部动作的平滑性与节奏协调，在不依赖未来输入信息的条件下完成逐帧推理，更契合在线实时交互场景的因果性约束。此外，平衡拉丁方版本进一步证明主观结论在不同呈现顺序下的一致性，排除了顺序偏差对结果的显著影响。

综上，用户研究表明 FaceCapGes 在在线实时生成条件下仍能维持与离线模型相近的主观表现，验证了本文提出的多模态融合与时间建模策略的有效性。

5.4 定性分析

我们强烈建议观看附录中的演示视频（附录 A），内容包含 Ground Truth (GT)、本模型 (FaceCapGes)、CaMN 与 DiffSHEG 在不同演讲数据上生成的手势的并排展示，能够直观体现时间对齐性、手势响应性以及头部-身体协调性方面的差异。

为进一步理解模型在实时因果约束下的动态响应行为，我们对每个模型的生成手势中进行了逐帧观察，并选取具有代表性的片段进行案例分析。

5.4.1 生成动作平滑性

如图 5.5 所示，FaceCapGes 在多个片段中均能平滑地响应说话人的语调变化。当语调出现明显上升或下降趋势时，本模型生成的双手高度能够随之连续变化，且动作幅度保持自然，整体运动轨迹连续、关节过渡平稳。该现象与主观用户评估中参与者对 FaceCapGes “动作过渡自然、节奏感强”的反馈一致。

相比之下，CaMN 的生成动作在语调变化较快的片段中表现出一定的迟滞性：双手高度调整通常较为缓慢，且动作幅度变化更趋于保守，导致整体动作轨迹的动态范围较小，视觉上更容易产生“僵硬感”。这一观察与用户反馈中提到的 CaMN “转向延迟与衔接僵硬”现象相一致。

DiffSHEG 的生成动作整体更活跃，在部分片段中能够在更细粒度的时间尺度上产生频繁的手势变化。然而，我们也观察到其在少数时间步出现局部关节的突然加速或短暂抖动，表现为手部轨迹在相邻帧间产生明显跳变或方向快速反转。该现象与主观用户反馈中提到的“偶发抖动”一致，可能与其旋转表示在数值空间中存在非连续点有关，从而使推理阶段更容易产生局部突变。

总体而言，FaceCapGes 通过采用 Rot6D 表示并结合帧级因果时间建模，在保证实时性的同时维持了更高的动作连续性与稳定性，生成结果在视觉平滑性上更接近真实动作序列。

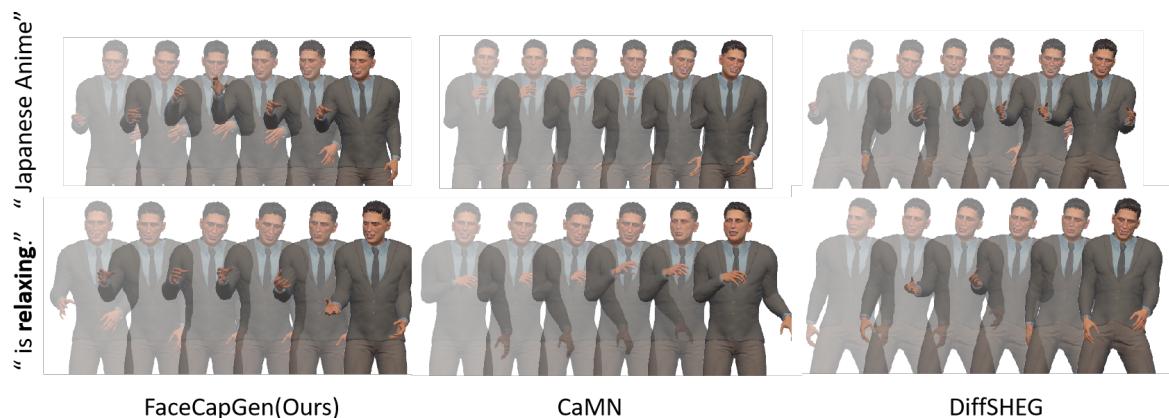


图 5.5 生成动作效果对比
Figure 5.5 Qualitative Comparison of Generated Gestures

5.4.2 头部朝向与手势空间指向一致性

为分析头部姿态输入对生成手势空间指向的影响，我们选取测试集中 GT 头部朝向发生明显偏转的片段，并对比不同模型在该时刻生成动作的朝向一致性表现。这里的“空间指向一致性”指生成手势的主要运动方向是否与角色的头部/身体朝向保持匹配，从而反映模型是否能够利用非语言姿态线索生成更符合交互场景的空间表达。

值得注意的是，CaMN 与 DiffSHEG 在输入端均不显式包含头部姿态信息，因此其生成的手势方向主要依赖语音或文本内容，难以体现真实头部转向带来的空间指向变化。

图 5.6 展示了一个包含明显头部转向变化的片段截图，从左到右依次为 GT、本模型（FaceCapGes）、CaMN 与 DiffSHEG 的生成结果。在该片段中，GT 的身体与头部朝向呈现出明确的空间指向，并且手势多沿着当前头部/躯干朝向展开，体现出面向不同听众时的自然交流习惯。



图 5.6 头部朝向与手势指向一致性的动作对比

Figure 5.6 Comparison of Head Orientation and Gesture Direction Consistency

我们观察到，本模型在多数帧中能够保持与 GT 相近的身体朝向，并使双手动作的空间指向与头部朝向保持一致，例如在头部向右侧转动的阶段，生成的手势也倾向

于朝向相同方向展开。相比之下，CaMN 的动作表现更为僵硬，身体朝向变化较弱且缺乏稳定的空间锚定，手势方向在多帧间呈现随机波动。DiffSHEG 虽能在其可获取的语音信息下生成较连贯的手势节奏与整体动作幅度，但由于其输入不包含头部姿态信号，生成结果难以反映 GT 中由头部转向引起的空间指向变化。

该现象表明，在严格因果与实时输入条件下，头部姿态作为额外的非语言模态能够为手势生成提供空间指向上的中层约束，使动作更自然地与说话者的注意方向和交互对象匹配。这种头-手空间一致性在元宇宙等虚拟交互场景中尤为重要：当用户面向不同方向的听众或对象进行交流时，头部转向与与之匹配的手势能够增强空间合理性与沉浸感，从而提升多方位交流中的可理解性与聆听体验。

5.5 客观评估指标与实现细节

为客观层面评价模型在动作自然性、节奏同步性与多样性等方面的表现，本文在客观评估中采用四项度量：Fréchet Gesture Distance (FGD)^[43]、语义相关动作召回率 (Semantic Relevance Gesture Recall, SRGR)^[3]、节奏对齐度 (Beat Alignment, BA)^[3,44]以及 L1 范数 (L1DIV)。这些指标分别对应生成动作在分布一致性、语音同步性与变化丰富性等不同维度，共同构成对模型质量的综合评估体系。

5.5.1 Fréchet Gesture Distance (FGD)

FGD^[43]用于衡量生成手势分布与真实手势分布之间的统计距离，灵感源自图像生成领域的 Fréchet Inception Distance (FID)。不同于图像任务直接利用 Inception 网络特征，在动作生成领域，特征空间需由单独训练的动作自编码器定义。该自编码器通过重构任务学习手势的潜在表示，使潜在空间具备对运动模式的压缩与区分能力。在该潜在空间中，假设真实分布与生成分布的高维嵌入向量分别为 $\mathcal{N}(\mu_r, \Sigma_r)$ 与 $\mathcal{N}(\mu_g, \Sigma_g)$ ，其中， $\mathcal{N}(\cdot)$ 为高斯分布， μ 与 Σ 分别为嵌入特征的均值向量与协方差矩阵。

此时，FGD 定义为：

$$\text{FGD} = \|\mu_r - \mu_g\|_2^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}). \quad (5.3)$$

其中， $\text{Tr}(\cdot)$ 为矩阵迹运算。

较小的 FGD 值表示生成动作的统计分布更接近真实数据，可反映动作的整体自然度与风格一致性。

FGD 的模型结构与训练配置 本文在每位说话人的训练集上分别训练一组评估用自编码器，以避免跨说话人分布差异对指标的干扰。自编码器输入为以 Rot6D 表示的上半身骨架序列，训练目标为最小化位置与速度的重构误差：

$$\mathcal{L}_{AE} = \lambda_r \|\hat{\mathbf{g}} - \mathbf{g}\|_2^2 + \lambda_v \|\hat{\mathbf{g}}' - \mathbf{g}'\|_2^2, \quad (5.4)$$

其中 $\lambda_r = 1$, $\lambda_v = 0.1$, \mathbf{g}' 为速度序列。

训练配置如下：输入片段长度设为 32 帧，训练片段切割步长设为 10 帧，批大小设为 256，编码器与解码器的隐藏层维度均为 128。优化器采用 SGD^[38]，基础学习率设为 $lr_{base} = 1.2 \times 10^{-4}$ ，并按批大小线性缩放为

$$lr = lr_{base} \cdot \frac{\text{batch_size}}{128}. \quad (5.5)$$

训练过程中仅使用位置与速度重构损失（式（5.4）），不包含加速度或对抗项。自编码器共训练 400 个 epoch。

用于 FGD 评估的骨架感知自编码器 在基线模型 CaMN 的 FGD 评估中，采用了一种基于时间卷积的嵌入式自编码器（embedding-based autoencoder）进行手势特征提取。该结构将每一帧姿态平铺为高维向量输入，并对各关节分量进行独立建模。在实际使用中我们观察到，这类嵌入空间对旋转表示的数值尺度较为敏感。当采用 Rot6D 表示时，不同关节与分量之间的不均衡方差可能在潜在空间中被进一步放大，从而导致协方差估计条件较差，并引发 FGD 数值不稳定的问题。

为提高 FGD 评估的鲁棒性，本文采用了一种骨架拓扑感知的自编码器作为特征提取器。该模型在编码阶段显式引入骨架邻接矩阵 A ，并通过局部卷积实现结构约束。具体而言，第 l 层中关节 i 的特征向量 $\mathbf{h}_i^{(l)} \in \mathbb{R}^{d_l}$ ，其中 d_l 表示第 l 层中每个关节节点的特征维度。通过聚合其邻域 $N(i)$ 内相邻关节 j 的特征 $\mathbf{h}_j^{(l)}$ 进行更新，其中 A_{ij} 表示骨架邻接矩阵中节点 j 到节点 i 的连接权重，即

$$A_{ij} = \begin{cases} 1, & \text{若关节 } i \text{ 与 } j \text{ 在骨架拓扑中直接相连, 或 } i = j; \\ 0, & \text{否则.} \end{cases} \quad (5.6)$$

此外，设 $W^{(l)} \in \mathbb{R}^{d_{l+1} \times d_l}$ 与 $b^{(l)} \in \mathbb{R}^{d_{l+1}}$ 分别为第 l 层的可学习权重矩阵与偏置项。设 $\sigma(\cdot)$ 为非线性激活函数，在本文实现中取为双曲正切函数 $\tanh(\cdot)$ 。上述局部邻域聚合过程可形式化表示为：

$$\mathbf{h}_i^{(l+1)} = \sigma \left(\sum_{j=1}^J A_{ij} W^{(l)} \mathbf{h}_j^{(l)} + b^{(l)} \right), \quad (5.7)$$

这种基于局部邻域的权重共享机制有助于保持人体运动的空间结构一致性，并在特征提取过程中缓解由旋转表示差异带来的数值尺度放大问题。

实验结果表明，该骨架感知自编码器在 Rot6D 下产生更加稳定的潜在分布统计量。在本文的实验设置中，该设计有效提升了 FGD 评估的数值稳定性，并有效避免了在 Rot6D 表示条件下使用嵌入式自编码器时出现的极端 FGD 数值现象。

5.5.2 Semantic Relevance Gesture Recall (SRGR)

SRGR 指标 (Semantic Relevance Gesture Recall)^[3] 用于衡量生成手势在语义相关时间段内与真实手势在数值层面的匹配程度。该指标关注的是语音语义触发的关键手势是否被正确生成，而非整体分布一致性或语音-手势节奏相关性。

与基于分布的评估指标（如 FGD）不同，SRGR 属于基于阈值的逐帧召回率度量，通过统计生成手势在允许误差范围内命中的比例，反映模型在语义相关动作重现方面的准确性。

定义与原理 在本文实现中，SRGR 作用于关节的旋转表示。设真实手势序列与生成手势序列在第 t 帧第 j 个关节的旋转表示分别为 $\mathbf{r}_{t,j}$ 与 $\hat{\mathbf{r}}_{t,j}$ ，其中 $\mathbf{r}_{t,j} \in \mathbb{R}^6$ 为关节的 Rot6D 表示， T 为序列总帧数， J 为关节数量。

在给定旋转表示误差阈值 δ 的条件下，若生成关节旋转与真实关节旋转之间的表示差异满足 $\|\mathbf{r}_{t,j} - \hat{\mathbf{r}}_{t,j}\|_1 < \delta$ ，则认为该关节在该时刻被成功召回。在本文实验中，阈值固定设为 $\delta = 0.5$ 。

SRGR 通过对所有时间帧与关节进行统计，并引入语义相关性权重 λ_t ，定义为：

$$\text{SRGR} = \frac{1}{TJ} \sum_{t=1}^T \sum_{j=1}^J \lambda_t \mathbb{I}(\|\mathbf{r}_{t,j} - \hat{\mathbf{r}}_{t,j}\| < \delta), \quad (5.8)$$

其中 $\mathbb{I}(\cdot)$ 为指示函数。

语义相关性权重 λ_t 用于强调语音中语义显著时间片段（如强调词或语义触发点）对应的手势匹配程度，由 BEAT 数据集提供的语义标注确定，从而使 SRGR 更加关注语义相关手势的召回情况，而非对所有时间片段进行均匀统计。

5.5.3 Beat Alignment (BA)

Beat Alignment (BA) 指标用于衡量语音节拍事件与手势关键动作事件在时间轴上的对齐程度，反映模型在语音-手势时序同步性 (temporal synchronization) 方面的性能。本文采用 BEAT/CaMN 中使用的 BeatAlign 指标^[3]，其原始形式由 Li 等人

提出于 AI Choreographer 工作中^[44]，并可视为音频节拍与动作节拍集合之间的单向 Chamfer 相似度度量。

与 SRGR 基于阈值的逐帧数值匹配不同，BA 关注的是离散事件层面的时间对齐关系，即手势关键动作是否在时间上合理地响应了语音中的重读节拍。

定义与原理 设语音节拍事件集合为 $\mathcal{B}_s = \{t_k^s\}$ ，通过对语音信号的能量相关特征进行起始点检测（onset detection）得到。具体而言，首先基于谱能量变化计算 onset strength 曲线，并在该曲线上进行峰值检测以获得候选语音事件位置。随后，引入 Root Mean Square (RMS) 特征作为短时能量幅度的描述，并在 RMS 曲线上对检测到的起始点进行 backtracking 校正，从而将语音节拍事件定位至能量实际开始上升的位置，以提高时间定位的稳定性与准确性。

手势关键动作事件集合为 $\mathcal{B}_g = \{t_m^g\}$ ，定义为关节运动速度的局部极小值点，对应动作中的停顿或方向变化等显著运动事件。

对于每一个语音节拍事件 t_k^s ，计算其与所有手势关键动作事件之间的最小时间偏差：

$$\Delta t_k = \min_m |t_k^s - t_m^g|. \quad (5.9)$$

随后采用高斯核函数将时间偏差映射为相似度分数，从而得到单个语音节拍的对齐得分。最终 BA 指标定义为：

$$BA = \frac{1}{|\mathcal{B}_s|} \sum_k \exp \left(-\frac{(\Delta t_k)^2}{2\sigma^2} \right), \quad (5.10)$$

其中 σ 为时间尺度参数，本文中取 $\sigma = 0.3$ ，用于控制对齐容忍范围。

该定义可视为从语音节拍集合到手势节拍集合的单向 Chamfer 相似度，当语音节拍与手势关键动作在时间上高度对齐时，BA 值接近 1；反之，当二者时间偏差较大时，BA 值趋近于 0。

本文使用 BA 指标评估模型在语音重读节拍与手势关键动作之间的时间同步能力，该指标与主观观察到的语音-手势同步自然度通常具有较好一致性。

5.5.4 L1 范数

L1 范数 (L1DIV) 指标^[3]用于衡量模型生成手势序列的多样性，即不同生成样本之间在动作空间中的差异程度。该指标反映模型在避免生成结果收敛到平均动作模式 (mode collapse) 的同时，是否能够保持足够丰富的动作变化。

定义与原理 设模型在评估过程中生成 N 个手势序列样本，第 i 个生成样本在第 t 帧第 j 个关节的旋转表示为 $\hat{\mathbf{r}}_{t,j}^{(i)}$ ，其中 $\hat{\mathbf{r}}_{t,j}^{(i)} \in \mathbb{R}^6$ 为关节的 Rot6D 表示， T 为序列总帧数， J 为关节数量。

L1DIV 通过计算不同生成样本之间在所有时间帧与关节上的平均 L1 距离，来刻画生成动作分布的离散程度。其数学形式定义为：

$$\text{L1DIV} = \frac{1}{N(N-1)} \sum_{i < k} \frac{1}{TJ} \sum_{t=1}^T \sum_{j=1}^J \left\| \hat{\mathbf{r}}_{t,j}^{(i)} - \hat{\mathbf{r}}_{t,j}^{(k)} \right\|_1. \quad (5.11)$$

较高的 L1 范数表明模型生成具有较强的多样性，但过高可能意味着动作不稳定或噪声放大。因此，L1 范数通常与 FGD 联合分析：FGD 反映真实度，L1 范数反映丰富度，两者共同平衡模型在自然性—多样性维度上的表现。

5.5.5 评估区域设定与公平性说明

由于本文方法 FaceCapGes 在输入端显式引入了真实的头部姿态作为额外模态，并在输出中生成包含头部旋转的上半身骨骼序列，若直接将头部旋转也计入整体误差，可能会对不使用头部姿态输入的对比方法（如 CaMN 与 DiffSHEG）造成不公平的优势。为保证跨方法的可比性，本文在所有定量评估指标中分别报告两种评估区域：

- (1) 上半身（含头部）：包含头部与上半身全部关节旋转；
- (2) 上半身（不含头部）：在计算指标时移除头部关节的旋转分量，仅统计身体与手臂部分。

具体而言，在计算 FGD、SRGR、BA 与 L1DIV 时，我们将头部关节的旋转维度从指标计算中排除，从而消除头部旋转差异对指标的直接影响。

5.6 定量评估结果

表 5.2 显示，FaceCapGes 在所有指标上均优于 CaMN。值得注意的是，“上半身（含头部）”与“上半身（不含头部）”两种评估区域下的结果趋势基本一致，表明本文方法的性能优势并非仅来源于头部姿态输入或头部输出的额外信息。

此外，表 5.2 还显示，与 DiffSHEG 相比，本模型 FGD 更低，SRGR 相近，且在 L1 范数上表现最优，表明其生成动作具备良好多样性。但这一结论与用户主观评分存在一定出入：DiffSHEG 在多样性上主观排名更高。

这个现象可能来自于 L1 范数的局限性：它主要衡量空间偏离程度，不能直接体

表 5.2 定量评估结果
Table 5.2 Quantitative Evaluation Results

区域	方法	FGD↓	SRGR↑	BA↑	L1DIV↑
上半身（含头部）	CaMN	47.732	0.098	0.845	7.591
	DiffSHEG	<u>26.846</u>	0.109	<u>0.883</u>	<u>11.028</u>
	本模型	23.385	<u>0.107</u>	0.913	13.284
上半身（不含头部）	CaMN	49.437	0.103	0.845	7.327
	DiffSHEG	<u>26.847</u>	0.110	<u>0.883</u>	<u>10.990</u>
	本模型	23.384	<u>0.107</u>	0.913	13.330

现动作的颗粒度，或用户感知上的丰富性。虽然我们的模型在动作结构上更丰富，但用户普遍认为 DiffSHEG 的动作更活跃，在合理的时机做出了更多吸引注意的动作。

5.7 消融实验分析

我们围绕两个核心设计展开消融：

- (1) 头部姿态输入是否能提供有效空间与时序线索；
- (2) 帧级自回归生成与滑动窗口训练是否优于片段级一次性解码。

为此，我们构造了三种变体：移除头部姿态输入、移除帧级生成策略，以及基线 CaMN。

所有消融实验均基于 BEAT 数据集的第 2 位说话人进行，训练与测试划分与主实验保持一致。

表 5.3 展示了各模块在四个指标上的结果。

表 5.3 消融实验结果
Table 5.3 Ablation Study Results

区域	变体	FGD↓	SRGR↑	BA↑	L1DIV↑
上半身（含头部）	CaMN	32.870	0.111	0.858	7.214
	移除头部姿态	19.591	0.123	0.916	10.642
	移除帧级生成	21.592	0.125	0.892	10.456
	FaceCapGes（本模型）	19.290	0.123	0.918	10.871

头部姿态输入的贡献 对比“移除头部姿态”与完整模型可以发现，尽管两者在 SRGR 上差异不大（均为 0.123），但完整模型在 FGD、BA 与 L1DIV 上均取得更优表现，尤其在动作分布一致性（FGD）与多样性（L1DIV）上呈现出稳定收益。这表明头部姿

态作为非语言模态提供了额外的空间方向与交互焦点线索，能够帮助模型在生成手势时保持更一致的身体朝向与动作指向，从而提升动作自然度与表达丰富性。

需要注意的是，该提升幅度相对温和，原因可能在于语音与面部表情已包含较强的节奏与情绪提示，头部姿态主要补充空间层面的约束，因此其收益更集中地反映在分布与多样性相关指标上。

帧级自回归生成的优势 值得注意的是，“移除帧级生成”版本允许利用未来上下文并采用双向 LSTM，但其整体表现仍不及帧级版本，尤其在 FGD 上出现较为明显的退化（FGD 从 19.290 上升至 21.592）。

一种合理解释是，该变体在训练阶段以独立窗口为单位进行优化，而在测试阶段采用整段演讲作为长序列输入并一次性生成完整动作序列。这种训练与推理的序列长度范围不一致，可能导致模型在长序列推理时的隐状态传播方式偏离训练分布，从而使生成动作在嵌入空间中的统计特性偏离真实数据分布，表现为生成分布与真实动作分布的距离增大（FGD 上升）。

尽管本模型同样采用窗口化训练，但在每个窗口内通过滑动窗口展开与纯自回归预测进行多步生成，并对每步输出的误差累计取平均作为优化目标，使模型在训练阶段即暴露于自身预测分布并学习局部动力学的稳定性。因此训练目标与在线推理过程保持一致，对推理阶段序列长度变化所引入的分布偏移具有更强鲁棒性。

与基线模型 CaMN 的差异 相比基线 CaMN，本模型在所有指标上均显著提升，其中 FGD 从 32.870 降至 19.290，表明生成分布更接近真实动作；同时 BA 与 L1DIV 的提高说明动作更平衡且更具表达多样性。这进一步验证了本文引入的因果时序建模、头部模态补充与滑动窗口训练策略对于实时交互场景下的手势生成具有有效作用。

此外，基线 CaMN 采用欧拉角而本模型采用 Rot6D 表示，该表示差异亦可能部分解释性能提升幅度较大的原因。

5.8 性能评估

5.8.1 单帧推理性能

FaceCapGes 作为端到端实时手势生成框架的核心计算模块，其性能评估聚焦于单帧输入，单帧输出的核心推理流程，即模型接收当前帧的语音、面部表情与头部姿态多模态输入，实时输出对应帧的上半身手势骨骼动画。

为模拟真实应用中的实时输入流场景，性能测试基于 BEAT 数据集的测试集展开，关闭批处理机制，确保每帧数据均独立输入模型进行推理，还原逐帧处理的实际运行状态。测试过程中，我们将测试集中总计 93015 帧的多模态输入数据传入模型，记录从首帧输入到末帧输出的总推理耗时，通过总时长与测试帧数的比值计算平均单帧处理时间。测试时使用的硬件配置为单张 RTX4090 硬件。

测试结果如表 5.4 所示，模型平均单帧处理时间为 6.07 毫秒，具备良好的实时响应能力。

表 5.4 推理速度评估结果
Table 5.4 Inference Speed Evaluation Results

指标	时间
测试帧数	93015 (f)
推理总时长	504 (s)
平均单帧时间	6.07E-03 (s/f)

5.8.2 端到端计算链路延迟

结合第 2.2 节所述的端到端框架流程，系统端到端延迟可按时间顺序拆解为数据采集、特征提取、模型推理与结果返回四个核心阶段。各阶段性能消耗及瓶颈分析如下：

- **数据采集阶段：**依赖设备端传感器实时捕获多模态信号，主要耗时来源于 ARKit 面部与头部姿态追踪。根据官方文档标注，在 iOS 设备上 ARKit 的目标追踪帧率为 60 FPS^[2]，对应输入更新周期约为 16.7 ms。该阶段耗时由设备端硬件算力与系统负载决定。
- **特征提取阶段：**将原始传感器数据转换为模型可识别的结构化特征（语音 Mel 频谱、面部 BlendShape 系数、头部 Rot6D 旋转参数）。测试原型中采用 Librosa 对测试集音频进行离线特征提取，93015 帧数据的总计算耗时约 61 秒，对应的平均单帧计算成本约为 0.66 ms。需要注意的是，该统计不包含实时 I/O 与缓冲管理等系统开销，但可用于估计音频特征提取的计算量级。实际部署时可替换为 PyAudio 等实时流式提取工具，因此该阶段预计不构成主要性能瓶颈。
- **模型推理阶段：**为端到端流程的核心计算环节，结合第 5.8.1 节的模型推理性能测试，在单张 RTX4090 硬件、无批处理（Batch=1）配置下，单帧推理耗时为 6.07 ms。该阶段耗时与硬件算力强相关，是由神经网络结构设计决定的主要性能变量。

- **结果返回阶段：**将模型输出的骨骼姿态参数传输至渲染引擎，耗时可忽略（通常低于 0.1 ms），不构成性能瓶颈。

在本地推理设置下，若将一次响应链路定义为单帧采集完成后立即进入后续计算，则计算链路延迟可近似表示为：

$$t_{\text{e2e}} \approx t_{\text{ARKit}} + t_{\text{feat}} + t_{\text{infer}} + t_{\text{return}}, \quad (5.12)$$

其中 $t_{\text{ARKit}} \approx 16.7 \text{ ms}$, $t_{\text{feat}} < 1 \text{ ms}$, $t_{\text{infer}} = 6.07 \text{ ms}$, $t_{\text{return}} < 0.1 \text{ ms}$ 。因此在该设置下，计算链路理论延迟可视为 25 ms 以下。

需要注意的是，该估计未计入音频特征提取的窗口缓存与渲染端同步可能引入的额外等待，实际交互延迟还将受渲染刷新周期影响。

远程推理部署的额外开销 在移动端等算力受限的平台上，模型推理可部署于远程服务器并通过网络进行输入输出传输。此时端到端延迟需进一步加上网络往返与序列化开销：

$$t_{\text{e2e}}^{\text{remote}} \approx t_{\text{ARKit}} + t_{\text{feat}} + t_{\text{net}} + t_{\text{infer}} + t_{\text{return}}, \quad (5.13)$$

其中 t_{net} 表示网络传输与通信开销，其大小由网络条件与系统实现决定，本文不在此展开测量。

5.8.3 系统更新率

在本实验设置下，模型在单张 RTX4090 上的单帧推理时间为 6.07 ms，对应的理论推理吞吐约为 165 FPS，说明模型推理阶段在当前硬件条件下不会成为实时交互中的性能瓶颈。需要强调的是，该数值仅反映神经网络推理吞吐能力，并不等同于系统端到端更新率。

然而，端到端系统的实际更新率仍将受到输入采集频率与渲染刷新率的共同约束。根据官方文档，ARKit 面部与头部追踪通常以 60 FPS 为目标更新率^[2]，因此在采用 ARKit 作为面部与头部输入来源的应用场景中，系统可获得的相关输入更新频率理论上限可视为 60 Hz，相应地，系统的有效输出更新率亦不会超过 60 FPS。在远程推理部署下，网络传输与同步开销可能进一步降低实际更新率。

5.8.4 帧率设定的可扩展性

本文实现中采用 15 FPS 的动作时间采样率进行训练与输出，主要原因在于与 BEAT 数据集预处理及基线模型的设置保持一致，以便进行公平对比。该采样率选择

并不构成模型的固有限制。

但需要注意的是，本文模型的输入模态不包含帧间时间间隔 Δt 的显式信息，因此训练过程中隐式假设固定的离散时间步长（即 $\Delta t = 1/15\text{ s}$ ）。当部署环境的输入采样率或输出刷新率发生变化（例如 ARKit 为 60 FPS）时，若直接使用 15 FPS 训练的模型进行逐帧推理，模型可能会对运动速度与节奏尺度产生偏差，从而影响手势动态表现。

因此，在目标运行帧率与训练帧率不一致的情况下，更稳妥的做法是对数据集进行对应帧率的重采样并重新训练模型，以确保训练时域与实际系统时域一致，从而使模型学习到正确的动力学时间尺度。未来也可进一步探索将 Δt 作为显式条件输入，提高模型对不同采样率输入的鲁棒性。

在实际部署中，系统可依据端到端管线中的主要瓶颈选择合适的目标动作帧率：当推理计算为瓶颈（如移动端或低算力设备）时，可维持较低帧率以确保每帧按时生成；当输入采样率与计算资源允许（例如更高刷新率的追踪）时，可采用更高帧率训练版本以获得更细粒度的时间响应与动作细节表达。

5.9 本章小结

本章围绕 FaceCapGes 的生成质量与实时性能开展了系统评估。

在主观评估方面，本章结合两轮用户评估对模型在真实感、同步性与多样性三个维度进行了分析，结果显示本方法整体优于基线模型 CaMN，并在真实感维度上与离线模型表现相近。结合用户反馈与访谈，本章进一步讨论了头部姿态信息在提升动作朝向一致性与真实感方面的潜在贡献。此外，通过对生成结果的定性分析，我们观察到本方法能够随头部朝向变化生成方向一致的手势动作，从而增强交互场景下的空间合理性。

在客观评估方面，本文采用 FGD、SRGR、BA 与 L1DIV 等指标，分别从动作分布一致性、语音节奏对齐、动作平衡性与多样性等角度对比了 CaMN 与 DiffSHEG 等方法，并通过“含头部/不含头部”两种评估区域设定保证指标对比的公平性。在此基础上，本章进一步通过消融实验分析了头部姿态输入与帧级自回归生成策略对性能提升的贡献，验证了各模块设计的有效性。

最后，本章对模型推理速度、端到端计算链路延迟与系统更新率进行了评估，表明 FaceCapGes 在保持较高生成质量的同时具备实时交互所需的低延迟响应能力。

第6章 结论

6.1 本文工作总结

本文提出了 FaceCapGes，一种基于 CaMN 框架的帧级实时语音驱动手势生成模型。与依赖完整时间上下文或未来信息的离线方法不同，FaceCapGes 能够仅利用实时音频信号、面部 blendshape 权重以及头部姿态信息，在无动作捕捉设备的条件下驱动虚拟角色生成自然、连贯的手势动作。

在模型设计上，本文融合了 LSTM、MLP 与对抗学习机制，构建了级联式网络结构，并采用滑动窗口的自回归训练策略，从而在保证生成质量的同时实现低延迟的在线推理。本文首次将头部姿态作为终端输入模态系统性地引入实时手势生成任务，有效提升了生成手势在节奏协调性与整体自然度方面的表现。

综合客观评估与主观用户研究结果表明，FaceCapGes 在生成质量上显著优于 CaMN 等基线方法，并在保持实时交互能力的前提下，在多项指标上达到与主流离线方法相当的水平。此外，模型采用模块化设计，能够部署于兼容 ARKit 的轻量级设备之上，验证了其在实际交互场景中的可行性与应用潜力。

6.2 未来工作展望

6.2.1 高层语义信息

当前模型主要关注语音声学特征与运动感知模态对手势生成的影响，尚未显式引入语言层面的语义理解与表达意图建模。未来可结合实时语音识别与增量式语义解析技术，引入语篇结构、强调意图或对话功能等高层语义信息，以丰富手势在交互场景中的表达能力。在不破坏实时性的前提下，探索对有延迟但可修正的语义假设的鲁棒利用方式，将有助于提升生成手势在语义层面的准确性与一致性。

6.2.2 面向未来趋势的预测性训练目标

从建模目标的角度来看，当前 FaceCapGes 的训练过程主要以当前时间步手势姿态的重建误差为优化目标，即在给定历史与当前多模态输入的条件下，监督模型对当前手势的预测精度。然而，该学习目标并未对未来时间段内手势节奏与结构变化施加显式约束，使模型对历史信息的利用更多服务于当前帧生成，而非对即将发生的动作

变化进行前瞻性建模。

未来的研究可在现有框架基础上，引入针对手势未来趋势的预测性监督信号，尤其是充分挖掘头部与面部动态中所蕴含的准备性线索。与直接预测未来具体手势姿态不同，该方向更侧重于对抽象化时序属性的建模，例如未来短时间窗口内的手势起始概率、运动能量变化或强调强度等。这类趋势性变量具有时间平滑、语义明确且可提前出现的特点，适合作为实时系统中的前瞻性约束。

通过在训练阶段同时优化当前手势生成与未来趋势预测两个目标，模型有望学习到更具时间结构性的中间表示，从而在不引入额外模态或显著增加系统延迟的前提下，实现对手势节奏的提前准备与更稳定的时序对齐。

6.3 本章小结

本章对本文提出的实时手势生成方法 FaceCapGes 进行了总结，回顾了模型在结构设计、多模态融合以及实时推理方面的主要贡献。通过在无需动作捕捉设备的条件下实现高质量、低延迟的手势生成，本文工作验证了基于感知模态驱动虚拟角色表达的可行性与有效性。

同时，本章从语义建模与时间结构学习两个角度，对未来可能的研究方向进行了展望。相关扩展有望在保持实时性的前提下，进一步提升生成手势在表达意图、节奏一致性与应用适应性方面的表现，为面向自然人机交互的虚拟角色系统提供更坚实的技术基础。

参考文献

- [1] KARTYNNIK Y, ABLAVATSKI A, GRISHCHENKO I, et al. Real-time facial surface geometry from monocular video on mobile gpus[J]. arXiv:1907.06724, 2019.
- [2] NYISZTOR K. Introduction to Augmented Reality with ARKit[EB/OL]. 2019. <https://www.pluralsight.com/resources/blog/guides/introduction-to-augmented-reality-with-arkit>.
- [3] LIU H, ZHU Z, IWAMOTO N, et al. BEAT: A Large-Scale Semantic and Emotional Multi-Modal Dataset for Conversational Gestures Synthesis[J]. arXiv preprint arXiv:2203.05297, 2022.
- [4] KENDON A. Gesture: Visible Action as Utterance[M]. Cambridge, UK: Cambridge University Press, 2004.
- [5] MCNEILL D. Hand and Mind: What Gestures Reveal about Thought[M]. Chicago, IL: University of Chicago Press, 1992.
- [6] BAARS S, ANDEWEG B. ‘Wapper wat meer met je handen’ [J/OL]. Tijdschrift voor Taalbeheersing, 2019, 41(1): 3-17. <https://www.aup-online.com/content/journals/10.5117/TVT2019.1.001.BAAR>. DOI: <https://doi.org/10.5117/TVT2019.1.001.BAAR>.
- [7] WAGNER P, MALISZ Z, KOPP S. Gesture and speech in interaction: An overview [J/OL]. Speech Communication, 2014, 57: 209-232. DOI: 10.1016/j.specom.2013.09.008.
- [8] HADAR U, BUTTERWORTH B. Iconic gestures, imagery, and word retrieval in speech[J]. Semiotica, 1989, 75(1/2): 63-83.
- [9] ESTEVE-GIBERT N, PRIETO P, PONS X, et al. The timing of head movements: The role of prosodic heads and edges[J/OL]. The Journal of the Acoustical Society of America, 2017, 141(6): 4727-4739. DOI: 10.1121/1.4986649.
- [10] MAHMOOD N, GHORBANI N, TROJE N F, et al. AMASS: Archive of Motion Capture as Surface Shapes[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). 2019: 5442-5451.
- [11] LOPER M, MAHMOOD N, ROMERO J, et al. SMPL: A Skinned Multi-Person Linear Model[J]. ACM Transactions on Graphics (TOG), 2015, 34(6): 248:1-248:16.

-
- [12] GLEICHER M. Retargetting Motion to New Characters[C]//Proceedings of the 25th Annual Conference on Computer Graphics & Interactive Techniques (SIGGRAPH '98). ACM, 1998: 33-42.
 - [13] MARTINELLI G, GARAU N, BISAGNO N, et al. Skeleton-Aware Motion Retargeting Using Masked Pose Modeling[C]//European Conference on Computer Vision (ECCV) Workshops, LNCS 15624. Springer, 2024: 287-303.
 - [14] ZHOU Y, BARNES C, LU J, et al. On the Continuity of Rotation Representations in Neural Networks[J]. arXiv preprint arXiv:1812.07035v4, 2020.
 - [15] CHEN J, LIU Y, WANG J, et al. DiffSHEG: A Diffusion-Based Approach for Real-Time Speech-driven Holistic 3D Expression and Gesture Generation[C]//CVPR. 2024.
 - [16] FU C, WANG Y, ZHANG J, et al. MambaGesture: Enhancing Co-Speech Gesture Generation with Mamba and Disentangled Multi-Modality Fusion[EB/OL]. 2024. <https://arxiv.org/abs/2407.19976>. arXiv: 2407.19976 [cs.HC].
 - [17] LIU H, ZHU Z, BECHERINI G, et al. Emage: Towards unified holistic co-speech gesture generation via expressive masked audio gesture modeling[J]. arXiv:2401.00374, 2024.
 - [18] AO T, ZHANG Z, LIU L. GestureDiffuCLIP: Gesture Diffusion Model with CLIP Latents[J/OL]. ACM Trans. Graph., DOI: 10.1145/3592097.
 - [19] CHHATRE K, DANĚČEK R, ATHANASIOU N, et al. AMUSE: Emotional Speech-driven 3D Body Animation via Disentangled Latent Diffusion[C/OL]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2024: 1942-1953. <https://amuse.is.tue.mpg.de>.
 - [20] EKMAN P, FRIESEN W V. Facial Action Coding System: A Technique for the Measurement of Facial Movement[M]. Palo Alto, California: Consulting Psychologists Press, 1978.
 - [21] ARKit in iOS - Tracking and Visualizing Faces. 2024. https://developer.apple.com/documentation/arkit/arkit_in_ios/content_anchors/tracking_and_visualizing_faces.
 - [22] OZEL M. ARKit to FACS Cheat Sheet[EB/OL]. 2022. <https://melindaozel.com/arkit-to-facs-cheat-sheet/>.
 - [23] AMOS B, LUDWICZUK B, SATYANARAYANAN M. OpenFace: A general-

- purpose face recognition library with mobile applications[R]. CMU-CS-16-118, CMU School of Computer Science, 2016.
- [24] GINOSAR S, BAR A, KOHAVI G, et al. Learning Individual Styles of Conversational Gesture[C/OL]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019: 3497-3506. https://openaccess.thecvf.com/content_CVPR_2019/papers/Ginosar_Learning_Individual_Styles_of_Conversational_Gesture_CVPR_2019_paper.pdf.
- [25] ALEXANDERSON S, HENTER G E, KUCHERENKO T, et al. Style-Controllable Speech-Driven Gesture Synthesis Using Normalising Flows[C/OL]//Computer Graphics Forum (Proc. Eurographics). 2020: 487-496. <https://people.kth.se/~ghe/pubs/pdf/alexanderson2020style.pdf>.
- [26] KUCHERENKO T, BESKOW J, KJELLSTRÖM H, et al. Moving Fast and Slow: Analysis of Representations and Post-processing in Speech-Driven Gesture Generation[J/OL]. arXiv preprint arXiv:2107.12305, 2021. <https://people.kth.se/~ghe/pubs/pdf/kucherenko2021moving.pdf>.
- [27] KUCHERENKO T, HABERNAL I, BESKOW J, et al. Multimodal Analysis of the Predictability of Hand-Gesture Properties[J/OL]. Frontiers in Computer Science, 2021, 3: 10. https://web.cs.ucdavis.edu/~neff/papers/kucherenko2021hand_property_predictability_final.pdf.
- [28] CASSELL J, VILHJÁLMSSEN H H, BICKMORE T. BEAT: the Behavior Expression Animation Toolkit[C/OL]//SIGGRAPH '01: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. 2001: 477-486. <http://doi.org/10.1145/383259.383315>. DOI: 10.1145/383259.383315.
- [29] HUANG C M, MUTLU B. Robot behavior toolkit: generating effective social behaviors for robots[C/OL]//HRI '12: Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction. 2012: 25-32. <https://doi.org/10.1145/2157689.2157694>. DOI: 10.1145/2157689.2157694.
- [30] KIPP M. Gesture generation by imitation: from human behavior to computer character animation[C/OL]//. 2005. <https://api.semanticscholar.org/CorpusID:26271318>.
- [31] ZHANG Z, AO T, ZHANG Y, et al. Semantic Gesticulator: Semantics-Aware Co-Speech Gesture Synthesis[J]. ACM Transactions on Graphics (TOG), 2024, 43(4):

- 1-17.
- [32] ZHU L, LIU X, LIU X, et al. Taming Diffusion Models for Audio-Driven Co-Speech Gesture Generation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 10544-10553.
- [33] YANG S, WU Z, LI M, et al. DiffuseStyleGesture: Stylized Audio-Driven Co-Speech Gesture Generation with Diffusion Models[C/OL]//Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23. 2023: 5860-5868. <https://doi.org/10.24963/ijcai.2023/650>. DOI: 10.24963/ijcai.2023/650.
- [34] HOGUE S, ZHANG C, DARUGER H, et al. DiffTED: One-shot Audio-driven TED Talk Video Generation with Diffusion-based Co-speech Gestures[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. 2024: 1922-1931.
- [35] DEICHLER A, MEHTA S, ALEXANDERSON S, et al. Diffusion-Based Co-Speech Gesture Generation Using Joint Text and Audio Representation[C/OL]//ICMI '23: Proceedings of the 25th International Conference on Multimodal Interaction. 2023: 755-762. <https://doi.org/10.1145/3577190.3616117>. DOI: 10.1145/3577190.3616117.
- [36] ALEXANDERSON S, KUCHERENKO T, HENTER G E, et al. DiffGesture: Diffusion-based Co-Speech Gesture Generation[C/OL]//Proceedings of the 23rd ACM International Conference on Intelligent Virtual Agents (IVA 2023). 2023: 1-8. <https://dl.acm.org/doi/10.1145/3570945.3607282>. DOI: 10.1145/3570945.3607282.
- [37] YOON Y, KIM S, LEE J, et al. Speech Gesture Generation from the Trimodal Context of Text, Audio, and Speaker Identity[C/OL]//Proceedings of the 2020 ACM International Conference on Multimodal Interaction (ICMI). 2020: 22-30. <https://dl.acm.org/doi/10.1145/3382507.3418838>. DOI: 10.1145/3382507.3418838.
- [38] BOTTOU L. Large-Scale Machine Learning with Stochastic Gradient Descent[C/OL] //International Conference on Computational Statistics. 2010. <https://api.semanticscholar.org/CorpusID:115963355>.
- [39] Unity Editor [Computer Software][A/OL]. Unity Technologies. <https://unity.com/>.
- [40] Mechanim Animation System [Computer Software Module][A/OL]. Unity Technologies. <https://docs.unity3d.com/Manual/AnimationOverview.html>.

-
- [41] PICO 4 All-in-One VR Headset [Hardware Device][A/OL]. ByteDance Inc. <https://www.pico-interactive.com/>.
 - [42] SCHWIND V, RESCH S, SEHRT J. The HCI User Studies Toolkit: Supporting Study Designing and Planning for Undergraduates and Novice Researchers in Human-Computer Interaction[C/OL]//Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems (CHI EA '23). ACM, 2023: 7. DOI: 10.1145/3544549.3585890.
 - [43] YOON Y, CHA B, LEE J H, et al. Speech gesture generation from the TRIMODAL context of text, audio, and speaker identity[J]. arXiv:2009.02119, 2020.
 - [44] LI R, YANG S, ROSS D A, et al. AI Choreographer: Music Conditioned 3D Dance Generation With AIST++[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). 2021: 13401-13412.

附录 A 手势生成对比视频

为直观展示不同模型在语音驱动手势生成任务中的表现，本文提供了基于 BEAT 数据集的可视化对比结果。具体而言，在说话人 2、4、6、8 的测试语音上，分别对本文模型 FaceCapGes、以及 CaMN^[3] 与 DiffSHEG^[15] 生成的手势序列进行了对比展示。

对应的生成结果视频可通过以下链接访问：

- **Gesture Generation Comparison Videos:**

[https://drive.google.com/drive/folders/1TmPqF-e6eHwSsMCQ2JNyPgGdMzKII_DbW?
usp=sharing](https://drive.google.com/drive/folders/1TmPqF-e6eHwSsMCQ2JNyPgGdMzKII_DbW?usp=sharing)

附录 B 代码与实现资源

本文模型 FaceCapGes 的完整训练与推理代码已开源，以便于复现本文中的实验结果与评估指标。代码仓库地址如下：

- **FaceCapGes GitHub Repository:**

<https://github.com/IORGestureTeam/FaceCapGes>

致 谢

硕士阶段的学习与研究经历对我而言是一段宝贵且难忘的时光。在此期间，我不仅系统地学习了图形学相关知识，也在不断的探索与实践中逐渐明确了自己的研究兴趣与方向。

首先，衷心感谢我的家人。感谢你们始终如一的关心与支持，在我面对压力与挑战时给予我理解与鼓励，让我能够专注于学业与研究，顺利完成硕士阶段的学习。

其次，特别感谢杨老师在课程教学与实验室项目中对我的培养与指导。老师带领我进入图形学领域并逐步深入，使我对图形学有了更深刻的理解，也让我最终能够在这一方向完成研究并发表论文。我将铭记这段经历，并期待将所学继续应用于未来的项目与工作之中。

同时，感谢实验室的各位同仁。感谢你们在日常交流中营造的良好氛围，以及在研究过程中对我实验流程、用户调研等工作的指导与协助。你们的建议与帮助不仅提升了我的研究效率，也让我更深刻体会到科研合作的价值。

最后，感谢在硕士课程中与我共同学习、互相支持的朋友们。与志同道合的同伴并肩前行、共同钻研图形学的深入内容，使我在学习过程中受益良多，也让这段时光更加充实而珍贵。

最后，衷心祝愿学校的老师与同学们身体健康、工作学习顺利、生活愉快。

学术论文和科研成果目录

学术论文

- [1] Hanaizumi J, Shang C, Yang X. FaceCapGes: Real-Time Frame-by-Frame Gesture Generation from Audio, Facial Capture, and Head Pose[C]. Computer Graphics International (CGI 2025), 2025.