上海科技大学
ShanghaiTech University

硕士学位论文

基于语音素材的骨骼动作数据生成

作者姓名：杨惠

指导教师：孙露 助理教授

上海科技大学信息科学与技术学院

学位类别：工学硕士

一级学科：计算机科学与技术

学校/学院名称：上海科技大学信息科学与技术学院

2023 年 6 月

Speech-driven Skeletal Animation Generation

A thesis submitted to

ShanghaiTech University

in partial fulfillment of the requirement

for the degree of

Master of Science in Engineering

in Computer Science and Technology

By

Yang Hui

Supervisor: Assistant Professor Sun Lu

School of Information Science and Technology

ShanghaiTech University

June, 2023

上海科技大学

研究生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。尽我所知，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研究成果。对论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明或致谢。

作者签名：

日　　期：

上海科技大学

学位论文授权使用声明

本人完全了解并同意遵守上海科技大学有关保存和使用学位论文的规定，即上海科技大学有权保留送交学位论文的副本，允许该论文被查阅，可以按照学术研究公开原则和保护知识产权的原则公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存、汇编本学位论文。

涉密及延迟公开的学位论文在解密或延迟期后适用本声明。

作者签名：　　导师签名：

日　　期：　　日　　期：

摘 要

为了实现更加真实和自然的虚拟人交互，本文探索了语音驱动的骨骼动画生成，目标是使得虚拟人在说话时能够自主的表达一些身体动作，从而丰富虚拟人的表达能力，提高虚拟人交互的自然度和用户的沉浸感。

现有的方法主要是通过采用编码器解码器结构实现音频生成动作序列，用生成动作序列的质量、多样性、语义相关性评估方法性能。为了提高生成动作序列的质量，之前的研究者们引入了判别器，通过训练生成对抗网络使得生成器生成的动作序列越来越真实。为了提高动作的语义性和动作的多样性，一些方法引入了音频转录的文本和动作风格，与音频一起作为编码器的输入。然而，根据最新的语音生成手势比赛结果显示，现有的方法在动作质量和语义相关性上仍然表现较差，绝大部分方法的评分仍然远远低于动作捕捉系统。此外，现有的方法主要是结合用户主观打分和客观评估公式对生成的动作序列进行评估。但是主观评分难以复现实验结果，同时每个方法里使用的客观评估公式无统一标准，从而导致不同方法间难以比较。

针对上述问题，本文做了如下两项工作：

第一，本文探讨了针对虚拟人骨骼动画的数据驱动方法，其中包括使用动作捕捉数据和视频生成数据。在探讨这些方法的基础上，本文明确了驱动流程、动画数据表示以及骨骼动画存储等相关知识。

第二，鉴于使用动作捕捉和视频生成等方法获取骨骼动画数据都有诸多限制，例如需要演员在录制设备系统中做出指定动作，这极大地限制了虚拟人的可交互性。在这种情况下，本文聚焦于使用语音生成骨骼动画的研究。针对现有方法在动作质量和语义相关性方法存在的缺陷，本文提出了一种新的语音驱动的骨骼动画生成模型。该模型通过引入动作质量判别器提高动作序列的真实性，在使用说话人信息指定动作风格的同时提高了动作序列的多样性，并通过引入语义判别器提高了生成动作序列的语义信息。此外，语义判别器还可用作评估动作语义相关性的客观评估方法。在与其他方法的对比实验中，本文提出的模型在使用均方误差作为评价动作质量的客观公式时，相对于最佳方法获得了 14.71% 的性能提升。使用本文提出的语义判别器作为评价动作语义相关性的客观评估

公式时，本文提出的模型相对于最佳方法获得了 7%的性能提升。在主观评估方面，本文提出的模型生成的动作帧序列也取得了不错的视觉效果。

关键词：语音驱动的手势生成，骨骼动画，虚拟人，语义判别器

Abstract

In order to achieve realistic virtual human interaction, this paper explores the speech-driven gesture generation. This study aims to enable vitual agents to exhibit body gestures while speaking. This will enhance their expressive capabilities, improve virtual agent interactions, and augment user immersion.

Existing methods for generating gesture from audio mainly use encoder-decoder structures. These methods evaluate the performance of generated action sequences using metrics such as sequence quality, diversity and semantic relevance. In order to improve the quality of generated action sequences, previous researchers introduce discriminators and train generative adversarial networks to make the generated action sequences realistic. To improve the semantics and diversity of actions, some works introduce motion style and transcribe text from audio as inputs to the encoder along with audio. However, the results of the speech-driven gesture generation competition show that existing methods still perform poorly in terms of action quality and semantic relevance, compared to motion capture systems. In addition, existing methods mainly evaluate the generated action sequences by combining subjective ratings from users with objective evaluation metrics. However, subjective ratings are difficult to reproduce in experimental results, and the objective evaluation metrics used in existing methods have no unified standards, making it difficult to compare different methods.

To address the above problems, this paper focuses on the following two tasks:

First, this paper investigates data-driven methods for virtual human skeletal animation based on motion capture data and video generation data. Based on the exploration of these methods, this paper clarifies the relevant knowledge of driving processes, animation data representation and skeleton animation storage.

Second, this paper proposes to use speech to generate skeleton animation. This approach overcomes the limitations of motion capture that restrict actors to specific actions within a recording system and thus improves the interactivity of virtual characters. To address the drawbacks of current methods in action quality and semantic relevance, this

paper presents a novel model for generating skeleton animation with speech. In terms of action quality, the model improves the realism of the action sequence by introducing an action quality discriminator. As for action diversity, the model utilizes speaker information to specify the action style and enhances the diversity of the action sequence. In terms of semantic relevance, the model improves the semantic information of the generated action sequence by introducing a semantic discriminator. Additionally, the semantic discriminator can objectively evaluate the semantic relevance of the action. In comparative experiments with other methods, our proposed model outperformed the second best method by 14.71% in terms of action quality. In addition, in terms of the objective evaluation metric based on the proposed semantic discriminator, our model achieved a 7% performance improvement over the second best method in comparative experiments. In terms of subjective evaluation, the proposed model also achieved competitive results.

Key Words: Speech-driven Gesture Generation, Skeletal Animation, Virtual Humans，Semantic Discriminator

目 录

图形列表

表格列表

符号列表

符号	说明
f_a	音频帧数
f_m	动作帧数
f_t	文本长度
t_m	动作时间
t_a	音频时间
n_a	音频的采样点个数
d_a	音频的特征维度
d_m	动作的特征维度
L_{rec}	重建损失
L_{kl}	KL 散度
$L_{quality}$	动作质量的判别损失
L_{sync}	动作与语音相关性的感知损失
pose6d	6D 的旋转表示
fps	动作帧率
sr	音频采样率
bs	样本数量
mean_pose	动作数据的均值
max_pose	动作数据的最大值

第 1 章　引言

1.1　研究背景及意义

虚拟人是指由计算机程序或人工智能代理构建的，能够与人类进行交互并表现出某些人类特征和行为的虚拟实体。它们可以应用于多个领域，如游戏、动画、虚拟现实、人机交互、医学模拟等。虚拟人在游戏中可以作为角色，增强游戏的真实感和互动性；在动画中可以替代实际演员进行拍摄，降低成本和风险；在虚拟现实中可以创建更加真实的虚拟环境，提升用户体验；在人机交互中可以实现自然语言交互、表情识别等功能，提高用户满意度；在医学模拟中可以进行手术模拟、病理模拟等，提高医学教育和临床实践水平。

随着虚拟人技术的不断发展，人们对于虚拟人的逼真度和互动性的要求也越来越高。生成虚拟人骨骼动画正是改善这一问题的有效方法。因此，如图 1.1 所示，我们分别探索了通过动作捕捉系统、视频和语音生成骨骼动画数据并驱动虚拟人的方式。然而，由于通过动捕系统和视频驱动虚拟人在应用时仍存在一定的限制，如需要精密的设备、演员等，本文的重心将放在语音驱动虚拟人上面。不同于命令式地通过语音要求虚拟人完成特定任务，本文主要研究虚拟人语音协同的动作生成，即使得虚拟人在说话时，能自动产生与语音内容相符合的动作，从而增强虚拟人的互动性和真实性。

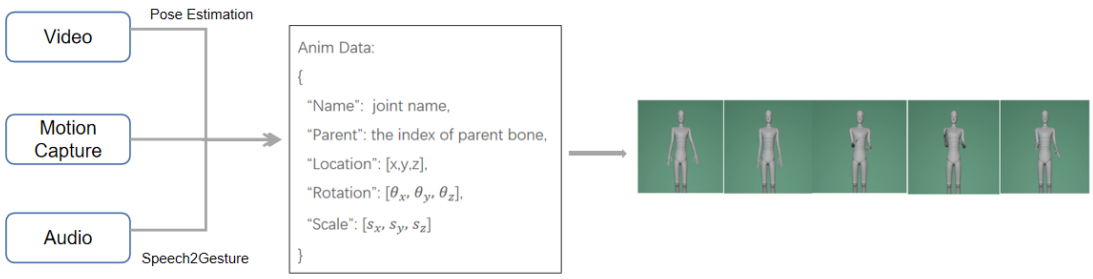


图 1.1　虚拟人骨骼动画生成

Figure 1.1　Skeleton animation generation of virtual human

1.2 研究内容

本论文将采用深度学习作为主要的理论和方法，利用深度神经网络模型实现通过语音生成虚拟人的动作。除此之外，本论文还就虚拟人的驱动方法，语音生成手势的评估指标等方面进行了探讨。

本文的研究内容主要有以下四个方面。

1. 本文详细整理了该领域的发展现状，以及涉及用到的基础知识。目前在该领域内尚无类似文献综述之类的出版物，这并不利于新接触该领域的人快速了解相关知识。本文期待通过整理这些知识，为之后有志于在语音生成手势方向的研究者提供快速了解领域概况的通道。

2. 本文分别探索了通过动作捕捉系统和视频生成动画数据，驱动虚拟人的流程，指出了两种方法存在的限制，从而提出了语音驱动虚拟人的方法。

3. 本文提出了一个语音生成手势的方法，分别就动作质量、动作多样性、语音相关性上三个方面做了不同的创新尝试。为了提高质量，引入了动作质量判别器，采用训练生成对抗网络（GAN）(Goodfellow 等, 2020) 的方式训练动作生成器和判别器，以使得动作质量越来越真实。为了提高动作多样性，在输入中引入了说话人信息，通过在动作序列生成时根据说话人信息指定动作风格，以实现音频序列和动作序列一对多的映射关系。为了提高动作与语音的相关性，提出了语义判别器，通过已有的训练数据构建音频动作相关的数据对和音频动作不相关的数据对，通过孪生网络 (Bromley 等, 1993) 训练了语义判别器。在训练动作生成器时，固定语义判别器的网络权重，将语义判别器的结果作为感知损失项以提高动作的语义性。此外，手势生成领域此前主要依靠人力判断动作的语义相关性，并无客观评估方法，而本文提出的语义判别器能作为评估工具判断动作的语义相关性。

4. 本文就动作质量、多样性、语义相关性等方面总结整理了之前研究者采用过的客观评估公式，并在相关比赛结果上实验验证了这些公式的有效性。

1.3 组织结构

本文主要分为 6 章。

第 1 章概述了本文选题的背景及意义，介绍了本文的相关研究内容及本文

的组织结构。

第 2 章总结了语音生成手势及相关领域的研究现状，揭示了领域内现存的瓶颈和问题，从数据集、评估指标、音频特征、动作表示多个方面归纳整理了前人用到的技术。

第 3 章分别讲述了使用动作捕捉数据和视频驱动虚拟人的流程，以及这两种驱动方式存在问题，并通过这两种驱动方式总结了动作表示、动画存储、动画补全等相关知识点。

第 4 章讲述了本文提出的语音生成手势的模型及训练、推理时的详细步骤。

第 5 章展示了相关实验结果。

第 6 章总结了本文的内容，并指出了该领域未来的发展趋势。

第 2 章 相关研究现状

2.1 跨模态序列生成

跨模态序列生成（Cross-modal Sequence Generation）是指从一个模态（例如图像、视频、声音）中生成另一个模态（例如自然语言文本）的任务。具体来说，它涉及到从一个模态中提取出有意义的信息，然后将这些信息转化为另一个模态所需要的序列数据。在相关领域，如通过音频生成口型 (Prajwal 等, 2020), 通过音频生成面部动画数据 (Karras 等, 2017)(Tian 等, 2019), 及本文涉及的通过音频生成骨骼动画数据，本质上都属于跨模态序列生成任务。

Seq2Seq 模型 (Sutskever 等, 2014) 是实现跨模态序列生成的一种常见方法。它旨在将输入序列映射到输出序列，输入和输出都可以是不定长序列，被广泛应用于语言翻译、问答等自然语言处理任务中。Seq2Seq 模型由编码器和解码器两部分组成。编码器负责处理输入序列并生成隐藏表示。它通过使用循环神经网络 (RNN) 或其变体 (如长短期记忆网络 (LSTM)(Hochreiter 等, 1997) 或门控循环单元 (GRU)(Cho 等, 2014)) 来生成一个固定长度的向量，以概括输入序列。编码器的输出是 RNN 的最终隐藏状态，其中包含有关输入序列的信息。解码器接收编码器的输出并生成输出序列。它也使用 RNN 或其变体，在各个时间步中使用输入序列的编码信息和上个时间步的输出以及隐藏状态作为输入，逐个元素生成输出序列。Seq2Seq 在处理长序列时可能会出现信息丢失或梯度消失的问题。为了解决这个问题, Vaswani 等 (2017) 提出了 Transformer 模型。它使用自注意力机制来捕捉输入序列中的依赖关系，从而避免了信息丢失和梯度消失的问题。同时，Transformer 还可以并行处理输入序列，加速了训练和推理的速度。跨模态 Seq2Seq 是将基本的 Seq2Seq 模型扩展到多模态数据上的一种方法，通常由多个编码器和一个解码器组成。在跨模态序列到序列模型中，模型的输入是多个序列，分别代表不同的模态，每个模态的序列都由一个编码器处理，最终生成一个联合表示。然后，解码器基于这个联合表示生成输出序列。

跨模态序列生成还可使用基于对抗生成网络 (Goodfellow 等, 2020) 的方法和基于自回归模型的方法。对抗生成网络使用两个神经网络模型来协同工作。其中

一个生成器模型生成跨模态序列，而另一个判别器模型则将生成的序列与真实的序列进行比较，以便进行训练和改进。自回归模型利用一种递归的方式来生成跨模态序列，其中每个时间步骤的输出依赖于前面时间步骤的输出，代表模型包括 Transformer(Vaswani 等, 2017) 和 GPT(Radford 等, 2018)。

跨模态方法研究的重点和难点在于如何进行多模态特征融合，多模态数据通常有不同的特征表示方式，如何将这些不同的特征结合起来，使其能够更好地进行跨模态生成是一个重要的问题。此外，跨模态序列生成涉及到时间维度建模的问题，如何保证生成序列的连续性和流畅性也是一个重要的难点。此外，评估跨模态生成的质量也是一个挑战，需要使用多个评价指标，例如 BLEU（(Papineni 等, 2002)）、ROUGE(Lin, 2004)、FID(Heusel 等, 2017) 等。

2.2　语音生成手势

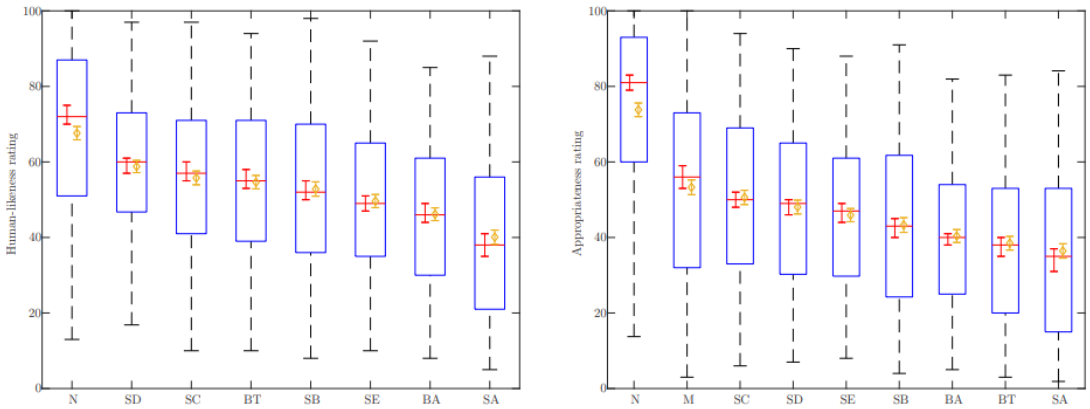
2.2.1　发展情况概述

根据 McNeill (1992) 提出的分类方法，手势可以分为四类：标志性手势、隐喻性手势、指示性手势、节拍性手势。标志性手势是指语义直接相关的手势，例如当我们说“高”时，会伸直手臂来表示高度。隐喻性手势是指用手势来描绘某个事物或场景，例如当我们说“限制”时，用手上下移动来模拟一堵墙。指示性手势则是指用手势来指向特定的目标或空间环境，例如用手指某个地方来引导别人前往。节拍性手势则是与音频有关的手势，例如伴随着音乐节奏做出的舞蹈动作。这些手势类型各有不同的功能和语义，可以用来增加交流的效果和准确性。因此，在生成语音驱动的手势动作时，应将这些手势考虑在内。

Yoon 等 (2022) 指出了现有语音生成手势领域存在的一些问题, 不同方法使用的数据集、动作表示、方法的评价指标都不统一，从而导致这些方法间没法直接比较。基于这个背景，Yoon 等人举办了两场关于语音驱动手势生成的比赛，即 GENEA Challenge 2020(Kucherenko 等, 2021), GENEA Challenge 2022(Yoon 等, 2022)。这两个比赛提供了统一的数据集、动作可视化流程、评估指标，确保了唯一的变量是参赛者提供的方法，从而可以比较这些方法的性能。根据这两个比赛的结果，可以大致了解这个领域的现状和行业发展趋势。

GENEA Challenge 2020 举办于 2020 年，数据集使用了 Trinity Gesture Dataset(Fer-

stl 等, 2018)，移除了数据中的下半身运动和手指运动。该比赛对生成动作序列的真实性、语义相关性两方面进行了评估，采用用户主观投票的方式对不同方法生成的结果进行投票。最终比赛结果如图 2.1 所示，横轴表示方法，纵轴表示得分，动捕数据（图中用 N 表示）的表现在两个方面都是最优的，且其他方法与动捕数据的评分相差较大。2020 年的比赛结果说明 2020 年及之后的一段时间，语音生成手势领域的发展趋势是提高生成动作的真实性和语义相关性。



(a) Human-likeness ratings　　(b) Appropriateness ratings

图 2.1　GENEA Challenge 2020 评估结果 (Kucherenko 等, 2021)

Figure 2.1　Evaluation results of GENEA Challenge 2020(Kucherenko 等, 2021)

GENEA Challenge 2022 举办于 2022 年，数据集使用了 Talking With Hands 16.2M(Lee 等, 2019)。相比于 GENEA Challenge 2020, GENEA Challenge 2022 拓展了数据集，使其包含了手指运动，下半身运动以及多个说话者的动态交互。此外，GENEA Challenge 2022 还改进了用户评分实验, 从而提高了实验结果的可信度。由于音频涉及的语言是英语，作者在 Prolific 平台上筛选了受试者，要求受试者的第一语言必须是英语，且受试者必须居住在英国（UK）、爱尔兰（IE）、美国（USA）、加拿大（CAN）、澳大利亚（AUS）、新西兰（NZ）这 6 个说英语的国家中的一个。最终参与实验的受试者信息如图 2.2 所示。

		gender			location					
		female	male	unknown	UK	IE	USA	CAN	AUS	NZ
Human-likeness	Full-body	60	60	1	110	3	4	2	2	-
	Upper-body	74	75	1	134	4	11	-	1	-
Appropriateness	Full-body	137	107	3	211	10	8	13	3	2
	Upper-body	127	173	4	256	1	35	10	2	-

图 2.2　GENEA Challenge 2022 的受试者信息

Figure 2.2　Voter information for GENEA Challenge 2022

比赛举办方在评估动作序列的类人性（Human-likeness）时，移除了音频，只需要受试者根据渲染展示的动作视频评估生成的动作质量，判断该动作是否像真实人类的动作。如图 2.3 所示，在同一页中给出相同语音，不同方法产生的多个动作，受试者需要对每个视频的动作质量进行评分。评分范围为 0-100，被分成 5 个等级，分别为优秀（Excellent）、良好（Good）、一般（Fair）、较差（Poor）、差（Bad）。每一页共有 8 个视频，且包含动捕数据，用于标定评分。由于动捕数据投影到模型上看着可能不会完全自然，因此不要求最好的模型要评分为 100。

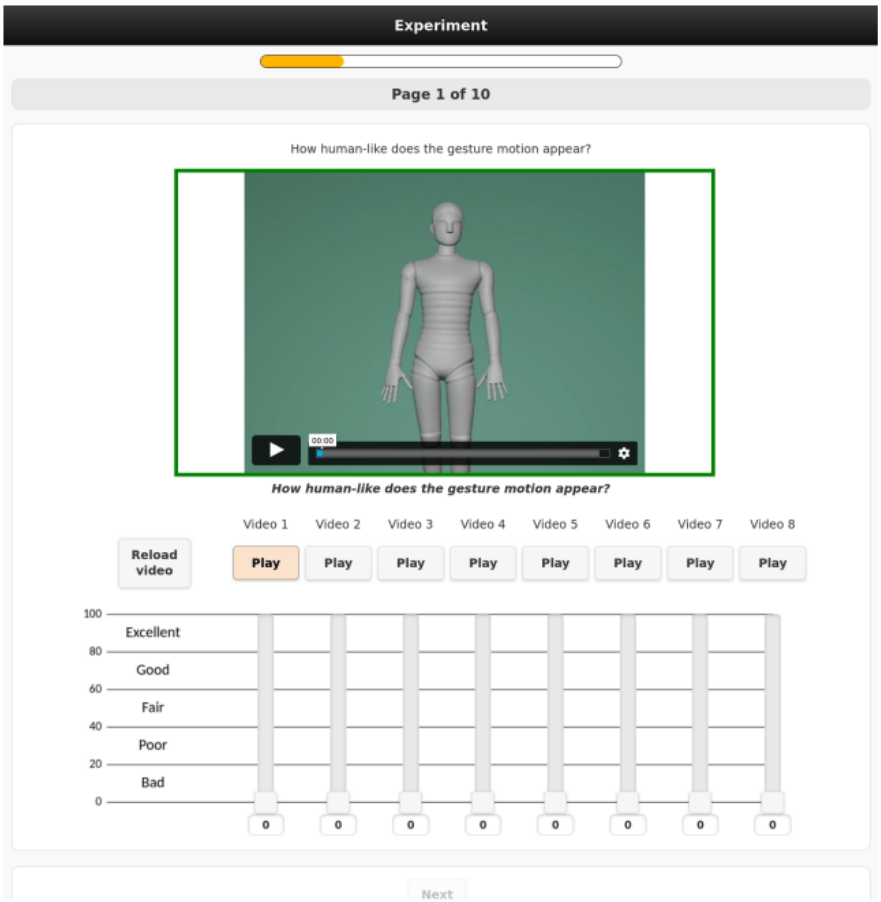


图 2.3　GENEA Challenge 2022 中动作质量的评估界面 (Yoon 等, 2022)

Figure 2.3　Evaluation interface of movement quality in GENEA Challenge 2022(Yoon 等, 2022)

相关性（Appropriateness）用于评估动作和输入语音的联系。GENEA Challenge 2020(Kucherenko 等, 2021) 在评估动作语音相似性时使用了基于多视频并行评分（Human Evaluation of Multiple Videos in Parallel，HEMVIP）的方法，每个视频都包含了音频。受试者被要求忽略动作质量，只对语义相似性进行评分。但是将动捕的运动片段与不相关的语音片段配对时，理论上该视频会获得最低

分，实际结果却是该视频最终获得了第 2 高的评分，说明在这种评估方式下，类人性会影响相关性的评估。GENEA Challenge 2022(Yoon 等, 2022) 改进了音频动作相关性的评估方法，如图 2.4 所示，在每一页中，只展示一对有相同语音的视频，这对视频都来源于同一个方法，确保了两个动作质量相同，但是其中一个动作音频匹配，另一个动作音频不匹配。在测试时，一个方法大约会生成 48 个语音动作片段，通过置换这些动作，直到所有的语音动作对都被置换过以构建不匹配的视频。一共会展示 40 页，受试者要求在每一页根据节奏，语调，意义选出最匹配的，由用户偏好代替打分。

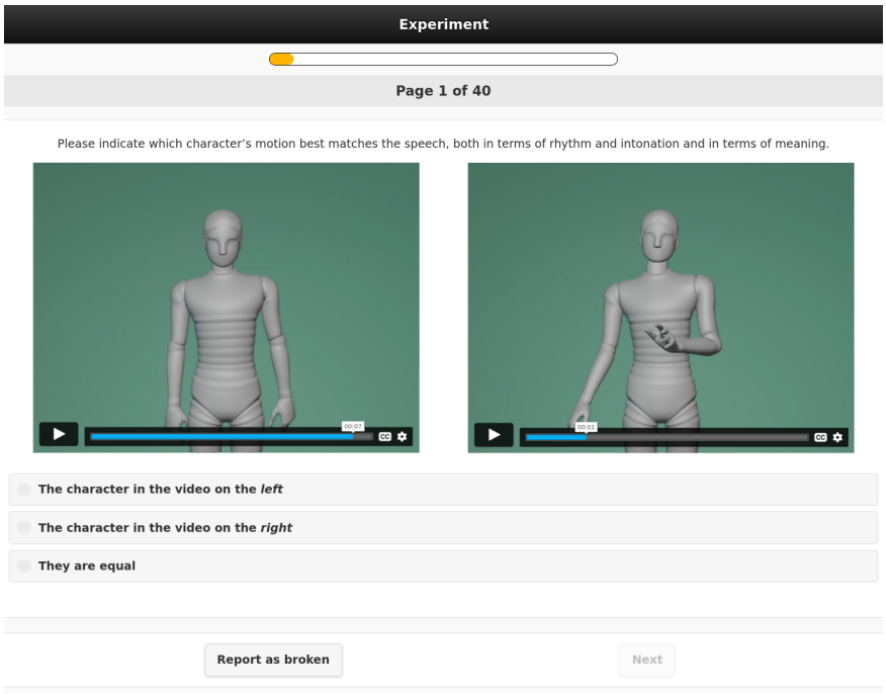


图 2.4　GENEA Challenge 2022 中语义相关性的评估界面 (Yoon 等, 2022)

Figure 2.4　Evaluation interface for speech relevance in GENEA Challenge 2022(Yoon 等, 2022)

为了确保受试者不是随意点击，比赛举办方设置了注意力检查，最终只会采用通过注意力检查的受试者的评分结果。在一些视频播放几秒后会出现注意力检查的信息提示。对于类人性研究而言，屏幕会出现”Attention!You must rate the video NN”, NN 的范围为 (5, 95), 受试者必须滑动滑块到相应的值且值的范围为（NN-3，NN+3），才能通过注意力检查。对于相关性研究而言，屏幕会出现”Attention!Please report this video as broken”, 受试者需要点击按钮”Report as broken”，以通过注意力检查。对于注意力检查失败 2 次及以上的受试者，其对

应的评分将会被移除。

相比于 GENEA Challenge 2020(Kucherenko 等, 2021), GENEA Challenge 2022(Yoon 等, 2022) 的评估结果略有进步。就类人性而言，有方法产生的动作序列超过了动捕数据集，如图 2.5 所示（横轴表示参赛方法，做了匿名化处理，纵轴表示类人性得分），FNA 表示动捕数据 (Lee 等, 2019)，FSA 是参赛的一个方法 GestureMaster(Zhou 等, 2022)，其用户评分的均值已经略微超过了动捕数据的评分均值。就相关性研究而言，如图 2.6 所示，目前还没有方法的评分超越动捕数据，且其他方法的评分与动捕的评分相差较大。

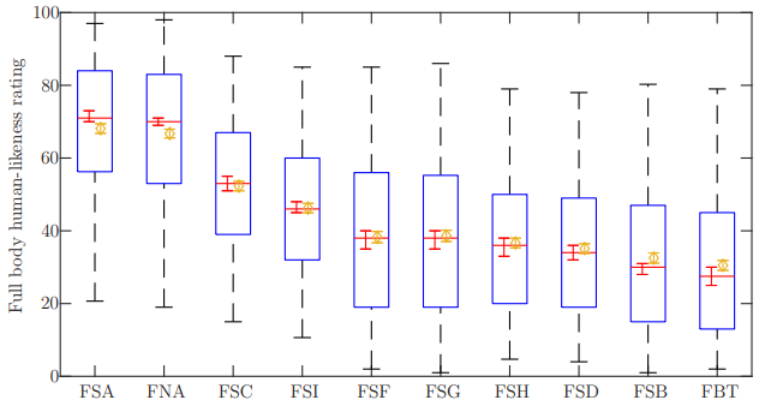


图 2.5　GENEA Challenge 2022 中动作质量的评估结果 (Yoon 等, 2022)

Figure 2.5　Evaluation results of movements quality in GENEA Challenge 2022(Yoon 等, 2022)

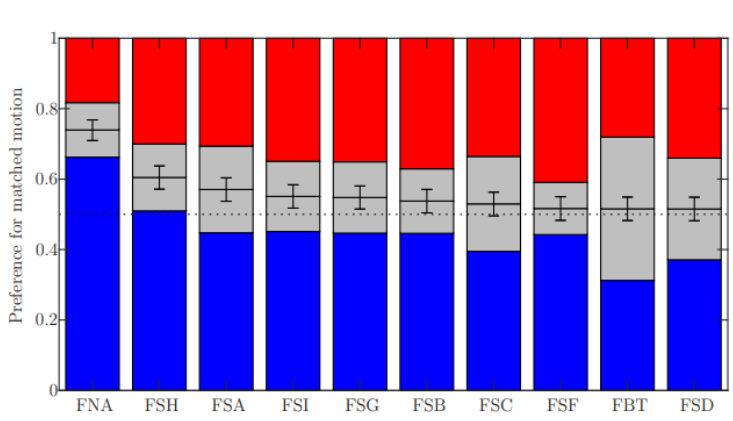


图 2.6　GENEA Challenge 2022 中语义相关性的评估结果 (Yoon 等, 2022)

Figure 2.6　Evaluation results of speech relevance in GENEA Challenge 2022(Yoon 等, 2022)

根据 GENEA Challenge 2022(Yoon 等, 2022) 的结果，我们可以了解到当前

语义生成手势领域所面临的瓶颈包括动作质量和动作与语音的相关性。就动作质量而言，虽然 FSA(Zhou 等, 2022) 的结果超越了动作捕捉系统，但其评分与动捕评分差距不大。此外，根据评分规则，这并不代表动作生成的质量完全自然，因为其距离满分仍有一定差距。FSA(Zhou 等, 2022) 是一个基于运动图的方法，图的节点由动作切片表示，一个动作序列则对应于图中的一条路径。这个方法本质上仍然依赖于动捕数据，且不能产生新的动作。与 FSA(Zhou 等, 2022)、FNA(Lee 等, 2019) 相比，其他方法在动作质量评分上表现差距较大。就语音相关性而言，没有方法的评分超过动捕数据，且其评分远低于动捕系统。因此，提高动作质量和动作与语音的相关性是该领域未来发展的重要趋势之一。此外，GENEA Challenge 2022(Yoon 等, 2022) 在其论文中指出，多人对话和交互也是该领域的另一个发展方向。

除了语义和动作质量的限制，手指运动也是该领域的另一大限制。目前的关键点估计算法并未同时考虑到身体骨骼和手指节点，且大多数视频中手的分辨率很低，很难构建一个同时包含身体运动和手指运动的数据集。即使采用动作捕捉系统，对于手指动画的录制效果也比较差。GENEA Challenge 2022(Yoon 等, 2022) 提供的数据集中包含手指的骨骼动画，但由于手指动画效果较差，很多参赛队伍选择移除手指部分的运动。

2.2.2 数据集

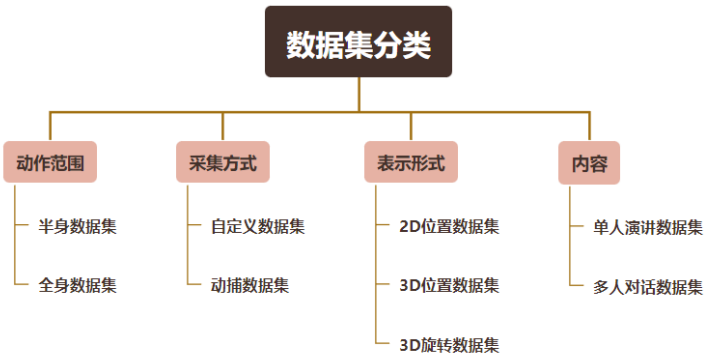


图 2.7 动作数据集分类

Figure 2.7 Classification of action datasets

该领域中现有的数据集并不具备统一性，这种不一致主要表现在动作数据的表示方式、动作表示的范围、数据集的内容、数据采集方式以及数据集的切片

方式等多个方面（如图 2.7）。

2.2.2.1　动作范围

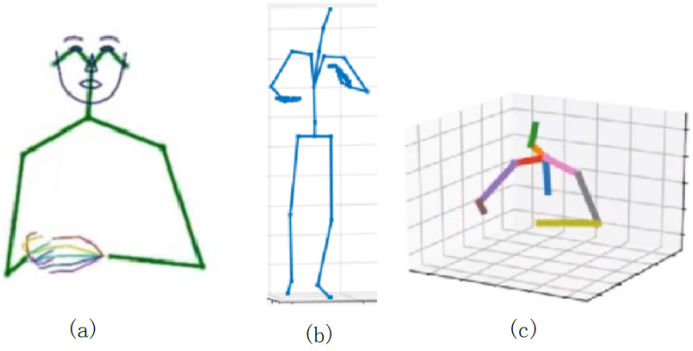


图 2.8　动作数据的关节点范围。(a) 用 2d 位置值表示的上半身动作 (Xu 等, 2022) (b) 用 3d 位置值表示的全身动作 (Kucherenko 等, 2019) (c) 用 3d 旋转值表示的上半身动作 (Yoon 等, 2020)

Figure 2.8　Joint range of action data.(a)Upper body movements represented by 2d position values(Xu 等, 2022) (b)Full body motion represented by 3d position values(Kucherenko 等, 2019) (c)Upper body movements represented by 3d rotation values(Yoon 等, 2020)

根据动作数据采集的范围不同，数据集可分为半身数据集和全身数据集（如图 2.8）。半身数据集通常以臀关节 (Hips) 为分界线，仅考虑说话时上半身的动作，不考虑身体整体的位移。而全身数据集则考虑所有关节点，包括所有关节点的旋转运动和身体整体的移动。尽管手指动作与语义紧密相关，但由于现有设备的限制，无法很好地捕捉手指运动信息，因此是否考虑手指关节点尚未统一规定。

2.2.2.2　采集方式

从数据采集的方式看，数据集可分为自定义数据集和动捕数据集。

自定义数据集是指研究者从网上爬取如脱口秀等视频，并经过切片、人体关节点检测、去抖动等处理步骤得到的数据集。这类数据集中通常使用现有的人体关节点检测算法进行数据处理，以关节点位置的变化来表示动画序列。其中，常用的 2D 人体关节点检测算法包括 OpenPose(Cao 等, 2017), 并被用于构建 Speech2Gesture Dataset(Ginosar 等, 2019), TED Gesture Dataset(Yoon 等, 2019) 等数据集。Speech2Gesture Dataset 包含 49 个 2 维的上半身关节点，由 504 个 YouTube

视频得到。TED Gesture Dataset 也只包含上半身关节点，由 1295 个 TED 视频得到。常用的 3D 人体关节点检测算法有 VideoPose3D(Pavllo 等, 2019)。TED Gesture Dataset(Yoon 等, 2020) 是一个只包含上半身关节点的语音-动作数据集，从 1766 个 TED 视频中处理得到。作者先由 2D 人体姿态检测算法检测得到 2D 人体姿态点，再通过 VideoPose3D 将其提升为 3D 人体姿态点。AIST++(Li, Yang 等, 2021) 是一个音乐-舞蹈数据集，源于 AIST 数据集 (Tsuchida 等, 2019), 提出了另一种将 2D 人体姿态点转换为 3D 人体姿态点的方法。AIST 数据集录制了舞蹈演员在多视角相机下跟随音乐节奏跳舞的视频。AIST++的作者先用 2D 人体姿态检测器获取每个视角下舞蹈动作的 2D 点，并通过光束法平差（Bundle Adjustment）获得相机参数，将 2D 点转为 3D 点。然后通过均方误差（MSE）令 SMPL 模型拟合这些 3D 点，从而得到关节点的旋转值。AIST++数据集包含全身关节，但不包含手指关节。

在处理自定义数据集以及通过模型预测动作序列时，序列抖动是难以避免的问题。解决此问题的方法包括使用滤波器和插值。常用的滤波器包括 Hodrick-Prescott(Hp) 滤波器 (Li, Yang 等, 2021)，Savitzky-Golay 滤波器 (Zhou 等, 2022)。有些研究将多种滤波器结合使用，例如先用 1 欧元滤波器去除小抖动，再用均值滤波器消除大抖动。常用的插值包括 Slerp 插值 (Zhou 等, 2022)(Athanasiou 等, 2022)(Shoemake, 1985)。

动捕数据集是指用动作捕捉设备录制动捕演员在演讲或者对话时的数据集，如 Talking With Hands 16.2M(Lee 等, 2019)，gesture-speech dataset(Takeuchi 等, 2017), Trinity Gesture Dataset(Ferstl 等, 2018) 等语言-手势数据集。Speech2Gesture Dataset(Ginosar 等, 2019), TED Gesture Dataset(Yoon 等, 2019), TED Gesture Dataset(Yoon 等, 2020) 的视频来源于网络，存在着镜头变换，画质不清晰等噪声，因此处理得到的人物动作范围多限制为上半身关节点。不过这种处理方式成本较小，同时也能处理得到更多、更大的数据集。而 AIST++(Li, Yang 等, 2021) 和动捕数据集通过搭建专业设备，采用专业演员得到数据集，这类数据集的噪声相对较小，因此一般得到的人物动作范围为全身关节点。由于 AIST++依赖于现有的关键点检测算法，因此其不含手指关节。

2.2.2.3　表示形式

根据动作数据的表示形式分类，目前已有的数据集可分为“2D 位置”、“3D 位置”、“3D 旋转”三类。其中，“2D 位置”是指使用关节点在图像上的 2D 位置表示其动作状态，如 FreeMo(Xu 等, 2022)。“3D 位置”是指使用关节点在三维坐标系下的位置表示其动作状态，如 DanceNet(Zhuang 等, 2020) 和 TriModal(Yoon 等, 2020)。这两类数据集主要通过爬取网络视频并经过切片、人体关节点检测、去抖动等步骤获得。这种获取方式无需较大的经济成本，有利于作者根据算法需要自定义数据集。然而，在现实生活中，处于同一位置的刚体由于其旋转角度的不同，其实际状态可能不同，因此用旋转角度描述刚体运动能够更准确地描述其在某一时刻的状态，从而在一定程度上避免歧义性。“3D 旋转”是指使用关节点在三维坐标系下的旋转角度表示其动作状态，因为骨骼动画是刚体运动，所以可以根据正向运动学等公式从关节点的旋转角度中获得其 3D 位置。目前已有的“3D 旋转”数据集有 AIST++(Li, Yang 等, 2021)，Talking With Hands 16.2M(Lee 等, 2019)，Gesture-speech dataset(Takeuchi 等, 2017)，Trinity Gesture Dataset(Ferstl 等, 2018)。

AIST++是一种音乐舞蹈数据集，使用多个视角的相机对舞蹈演员进行了拍摄。其处理步骤与“2D 位置”和“3D 位置”数据集的常规处理步骤类似，包括以下步骤：首先从在线网站收集大量讲话视频；对所收集到的视频进行数据切片，只保留有说话人的画面；使用 2D 姿态检测和追踪算法获取视频中说话人骨骼的关节点；核对上一步骤的结果，如有抖动发生，则使用 Hodrick-Prescott 滤波器 (Hodrick 等, 1997) 等去抖动方法消除特征；使用 ffmpeg 提取音频；通过均方误差作为损失项，使得 SMPL 模型 (Loper 等, 2015) 拟合每一帧骨骼各个关节点的位置，从而得到关节点在每一帧的旋转角度。

与 AIST++数据集的获取方式不同，Talking With Hands 16.2 M、Gesture-speech dataset 和 Trinity Gesture Dataset 是通过动作捕捉系统录制动画数据获得的。Taking With Hands 16.2 M 是以双人对话的方式录制动画，包含全身运动和手指运动，每段动画大约长达 10 分钟，总时长约为 50 个小时。该数据集提供由谷歌云自动语音识别工具转录得到的文本、说话人 ID 以及以 BVH 格式提供的动作数据，其涉及的语言是英语。Gesture-speech dataset 是来自以采访形式进行对话的日本人的数据集，包含语音和 BVH 动作文件。Trinity Gesture Dataset 是一个大规

模语音到手势的合成数据集，记录了一个男性英文演讲者谈论很多不同的话题，例如电影、日常活动等。该数据集提供英语语音，动作数据包含 BVH 和 FBX 两种形式。Trinity Gesture Dataset 包括两个子数据集，一个数据集包含 23 个动画，每个动画约长达 10 分钟，总共约 240 分钟；另一个数据集包含 25 个动画，每个动画约长达 10 至 20 分钟，总共约 370 分钟。

2.2.2.4 内容

根据数据集的内容，可以将数据集分为单人演讲和多人对话两类。在处理数据集时，有两种常见的切片方式。一种是通过滑动窗口切割固定长度的动作数据，另一种是根据文本内容切割整句话对应的动作数据 (Saleh, 2022)。据 Saleh (2022) 所述，使用固定窗口大小切割数据可能会刚好切割在某个单词的发音上，导致语义上的不连续。

2.2.3 现存的方法

如图 2.9 所示，本章节将从网络结构、网络模块、算法输入、映射关系和损失函数等方面介绍音频生成手势的各类方法。

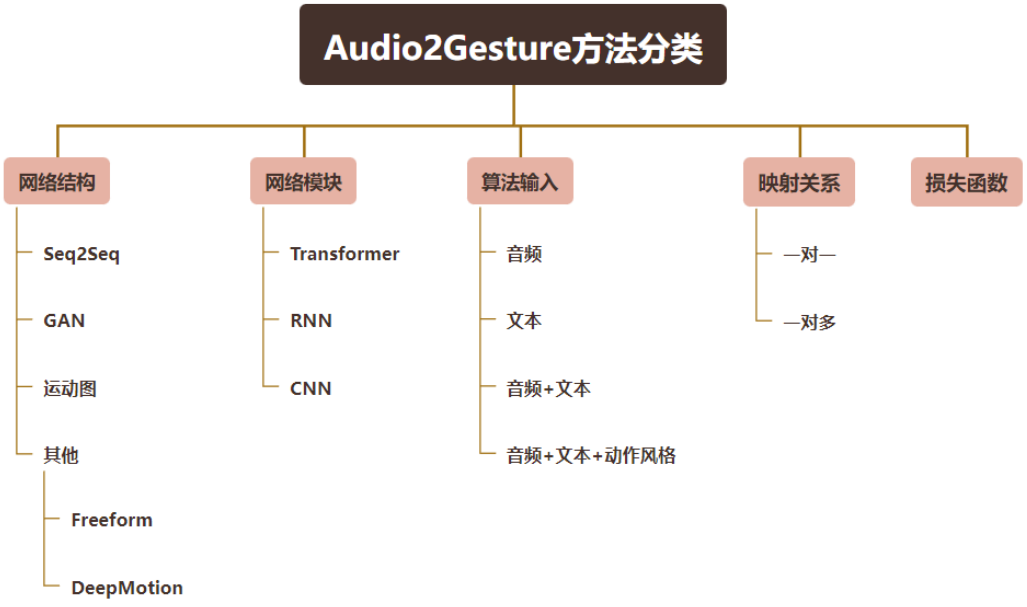


图 2.9 音频生成手势的方法分类

Figure 2.9 Classification of methods for generating gestures from audio

2.2.3.1　网络结构

近年来，通过语音生成动作序列的方法主要基于深度学习技术，其中涉及到的网络结构有 Seq2Seq、GAN、运动图和其他结构。基于 Seq2Seq 的方法 (Yoon 等, 2019)(Kucherenko 等, 2019)(Windle 等, 2022)(Ghorbani 等, 2022)(Saleh, 2022) 为每个输入分配一个网络分支作为该输入的特征编码器，然后将每个网络分支输出的特征融合在一起，作为解码器的输入，解码器再生成动作序列，如 (Yoon 等, 2019)(Kucherenko 等, 2019)(Ghorbani 等, 2022)(Saleh, 2022)(Windle 等, 2022)。基于 GAN 的方法 (Yoon 等, 2020) 将预测手势序列的网络作为生成器，评估动作质量的网络作为判别器，令生成器和判别器之间进行对抗性训练，以提高生成器生成真实手势序列的能力。基于运动图的方法 Kovar 等 (2008)Zhou 等 (2022) 先把动作数据构造成一个有向图，节点用动作切片表示，边表示动作间的变化，边的权重表示两个动作转换所花费的损失。合成的动作序列对应于运动图中的一条最优路径，使得损失最小化，可以使用动态规划思想解决这个问题。GestureMaster(Zhou 等, 2022) 在 GENEA Challenge 2022(Yoon 等, 2022) 中获得了最优的成绩，是目前已知在语音生成手势方向上，效果最真实的方法。但是运动图的方法也存在一些缺陷，Kovar 等 (2008) 所提出方法的缺点是动作间不能产生平滑的过渡, GestureMaster(Zhou 等, 2022) 通过在相邻动作切片间运用 Slerp 插值，使用 Savitzky-Golay 平滑合成的动作序列解决了这个问题。然而运动图的节点是动作切片，源于训练数据集，数据集的规模和准确度都会直接影响结果。如果数据集规模小或者数据集中动作存在抖动等问题，会导致合成的结果也保留这些问题。如果数据集规模大，则构造的运动图大，合成的动作能够更多变，但是会带来极大的时间成本，不利于方法的推广使用。此外，合成的动作序列实质上是运动图中的一条有向路径，所有的元动作都是现有数据集的，并不能产生新的动作。有两个特殊方法没有采用到 Seq2Seq、GAN、运动图这些结构。DeepMotion(Lu 等, 2022) 通过构建条件概率分布模型来生成动作。首先，从训练动作中提取小的单元手势集，并通过 VQ-VAE 模型学到这些动作的离散特征表示。然后，冻结 VQ-VAE 模型，将其作为手势编码器，基于先前的手势、语音和文本特征，使用 Transformer 预测每个单元手势是下一个手势的概率分布。在推理阶段，为了提高生成结果的多样性，不会直接选择概率最高的动作，而是在前 k 个概率最高的动作中随机选择一个。Freeform(Xu 等, 2022) 将音频生成动作分成两个过程。首

先，通过动作分支基于之前的动作帧生成主要的动作。然后，通过语音分支提取音频的动态节奏，并根据节奏生成动作偏移量。最终生成的动作由两个分支的输出直接叠加。

2.2.3.2 网络模块

在语音生成动作领域主要用到的网络模块有 CNNs(Ghorbani 等, 2022), Transformers(Vaswani 等, 2017)(Li 等, 2020)(Li, Yang 等, 2021)(Saleh, 2022)(Lu 等, 2022), RNNs(Saleh, 2022)(Windle 等, 2022)(Chang 等, 2022)。尽管在理论上，RNNs 和 Transformers 更适合生成长序列，但一些研究 (Aksan 等, 2020) 表明，当网络迭代次数增加时，会出现动作冻结或者动作漂移的问题。Kundu 等 (2020) 提出在训练时，将网络的输出作为输入可解决这类问题。RNNs 和 Transformers 使用上一状态的输出作为下一状态的输入，若直接训练，容易导致错误累计，影响模型收敛性。因此，在训练这些网络时通常采用强制教学（Teacher forcing）策略来加速训练，即使用真实标签中上一状态对应的值作为下一状态的输入。但是，在推理阶段，由于没有真实标签可用，只能用网络上一状态的输出，会导致曝光偏差（Exposure Bias），使得模型在训练和推理时的性能表现差距很大，无法阻止错误传递。对于输出离散值的序列生成任务，如机器翻译，通常会采用束搜索（Beam Search）策略，即对词表中每一个单词的预测概率执行搜索，生成多个候选的输出序列。通过这种启发式搜索，可减小模型训练阶段和测试阶段性能的差异。但是在手势生成任务中，模型需要输出一个动作序列，每个动作帧由一系列实值表示，束搜索并不适用于此任务。Saleh (2022) 采用了课程学习策略（Curriculum Learning）来解决这个问题。课程学习策略通过计划采样（Scheduled Sampling）(Bengio 等, 2015) 来避免曝光偏差，它在训练阶段不完全采用真实序列作为下一步的输入，而是通过概率 p 决定当前是使用强制教学策略还是使用模型上一步产生的输出，这个概率 p 会随着迭代次数的增加而逐渐降低。Bengio 等 (2015) 设定了 3 种 p 衰减的方式：线性衰减 (Linear decay), 指数衰减 (Exponential decay)，反向 Sigmoid 衰减 (Inverse sigmoid decay)。Dou 等 (2021) 提出了另一个适用于文字转语音领域（Text To Speech，简称 TTS）的方法，减小目标序列的帧率，即对目标序列进行下采样。由于 TTS 和动作序列生成本质上都是输出实值，故这个也可以参考用于这类方法中。针对 Transformer 训练中使用强制教学遇到的曝光

偏差问题，Mihaylova 等 (2019) 提出了一种不同的解决方法，即先通过解码器使用强制教学策略得到模型预测，然后将真实数据和模型预测混合得到一个新的序列，将其再次输入解码器产生最终预测。

2.2.3.3　算法输入

就算法的输入而言，最初生成人体动作序列的算法要求的输入比较单一，如通过音频生成动作序列 (Kucherenko 等, 2019), 通过文本生成动作序列 (Yoon 等, 2019)。两个方法均通过编码器对输入进行编码得到中间特征，然后传入解码器解码得到动作序列。由于语音和文本都具有一定语义信息，因此除了动作质量外，一般还会要求生成的动作在语义上跟输入有关。然而根据 GENEA Challenge 2022(Yoon 等, 2022) 的结果，目前尚无方法产生的动作能表现出良好的语义信息。一个可能的原因是使用的语音-手势数据集中，一个动作数据对应的语音或者文本片段并没有很强的语义信息。对于 McNeill (1992) 提出的四类手势——标志性手势、隐喻性手势、指示性手势、节拍性手势，跟语义有关的是标志性手势、隐喻性手势和指示性手势，跟音频节奏有关的是节拍性手势。在实际交谈或者演讲中，并不能保证所有手势都是带有语义的手势，从而导致使用的数据集没有强语义性。TEACH(Athanasiou 等, 2022)，TEMOS(Petrovich 等, 2022) 是两个通过文本生成动作序列的算法，其能根据文本描述生成具有指定含义的动作序列，其数据集中的文本和动作具有强烈的对应关系。由于现有语音转录文本工具的存在，如谷歌云自动语音识别工具，语音可以轻易转录成文本，因此有方法开始考虑通过语音和文本生成动作序列 (Lu 等, 2022)(Korzun 等, 2022)，以期待提高生成动作的语义性。Trimodal(Yoon 等, 2020) 提出了不同人的性格不一样，如性格内向的人动作幅度可能较小，性格外向的人动作幅度可能较大，因此不同的人说相同话语时其手势动作可能不一样，因此引入了说话人风格作为输入。在 Trimodal(Yoon 等, 2020) 中，用数字 ID 指定说话人，网络通过 KL 散度学得了一个隐空间用来表示训练数据中所有说话人风格的特征空间，通过说话人标识查阅这个特征表即可得到该说话人对应的风格。动作风格不止有说话人标识这一种表示方式，在 GestureMaster(Zhou 等, 2022), IVI Lab(Chang 等, 2022), Forgerons(Ghorbani 等, 2022), UEA Digital Humans(Windle 等, 2022) 这些方法中，有不同的其他表示。GestureMaster(Zhou 等, 2022) 通过网络获得语音的风格特征，

通过动作的平均速度、平均半径、平均高度表示动作的风格特征。IVI Lab(Chang 等, 2022) 将说话人 ID 表示成了独热向量（One-Hot Vector）。Forgerons(Ghorbani 等, 2022) 在输入时给定一段示例动作切片，风格编码器会从这段切片中学到动作风格，然后传递给生成器。UEA Digital Humans(Windle 等, 2022) 也是通过说话人 ID 指定说话人，每个说话人 ID 都可以查阅到说话人嵌入（Speaker Embedding）作为说话人的风格特征。

2.2.3.4 映射关系

之前的关于音频生成动作数据的工作主要可分为两类。一类是确定性的算法，实现了音频和动作序列一对一的关系，输入相同的音频会生成相同的动作序列 (Ren 等, 2020)(Kucherenko 等, 2019)。另一类是非确定性的算法，尝试构造音频和动作序列间一对多的关系 (Yu 等, 2020)(Li, Yang 等, 2021)(Li 等, 2020)。非确定的算法有以下几种构建方式，通过 VAE 构建一个概率模型 (Qian 等, 2021)(Li, Kang 等, 2021)(Lu 等, 2022)(Ghorbani 等, 2022)(Yang 等, 2022), 通过在输入中加入随机噪声 (Li, Kang 等, 2021), 通过计算输入音频的动态特征确定动作风格 (Zhou 等, 2022)(Xu 等, 2022), 通过在输入时给定一段动作切片决定生成动作的风格 (Ghorbani 等, 2022)(Li, Yang 等, 2021)。

2.2.3.5 损失函数

就损失函数的选取而言。一些方法直接选择计算生成动作序列与目标动作序列的距离，常用的损失函数有平均绝对误差损失（Mean Absolute Error，MAE）(Saleh, 2022)、均方误差（Mean Squared Error，MSE）(Kucherenko 等, 2019) 和 Huber Loss。在使用 MAE 训练神经网络时，一个重要的问题是，其梯度始终很大，这可能导致在使用梯度下降算法进行模型训练时，难以在最小值处停止训练。相比之下，使用 MSE 进行训练时，随着损失值接近最小值，梯度会逐渐减小，因此更加准确。Huber Loss 的公式如式 2.1 所示，当预测偏差小于 δ 时，采用平方误差，当预测偏差大于 δ 时，采用线性误差。相比于最小二乘的线性回归，Huber Loss 降低了对离散点的惩罚程度，因此 Huber Loss 同时具备 MSE 和 MAE 这两种损失函数的优点，是一种更鲁棒的回归损失函数。但 Huber Loss 也存在一个问题，可能需要训练超参数, 这个问题需要不断迭代。FreeForm(Xu 等, 2022)

指出，若直接使用均方误差等损失函数回归肢体关键点的位置，这种方法由于没有考虑到虚拟形象在执行动作时身体姿势的特点，会导致生成的动作真实感很差，肢体动作严重变形，并且动作和语音的匹配程度不高。因此，通常最终的损失函数会由几部分损失项共同构成。Text2motion(Yoon 等, 2019) 中的损失函数由 L_{mse}、$L_{continuity}$、$L_{variance}$ 三部分构成，$L_{continuity}$ 用于确保动作的连续性，$L_{variance}$ 的值为动作帧方差的负值，用于使网络生成动态的动作。Forgerons(Ghorbani 等, 2022) 的损失函数由 KL 散度和重建损失两部分构成，使用 KL 散度作为一个损失项是因为 Forgerons 中用了 VAE 从动作切片中学习动作风格，重建损失中对关节点位置、旋转角度及其移动速度进行了回归。UEA Digital Humans(Windle 等, 2022) 的损失函数同时考虑了关节点旋转、位置、加速度、速度、位移等方面。

$$L_{\delta}(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2, & \text{for } |y - f(x)| \leqslant \delta \\ \delta(|y - f(x)| - \frac{\delta}{2}), & \text{otherwise} \end{cases} \quad \ldots (2.1)$$

2.2.4　评估指标

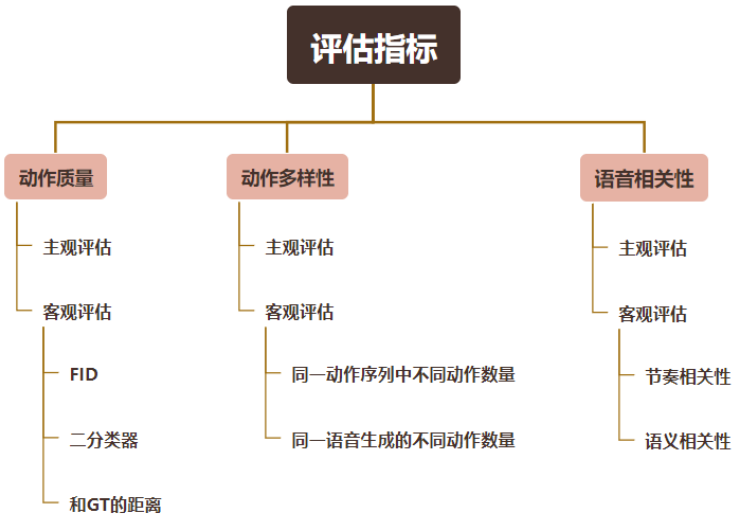


图 2.10　评估指标

Figure 2.10　Evaluation metrics

GENEA Challenge 2022(Yoon 等, 2022) 指出该领域存在的一个问题是评价指标尚未统一，每篇论文都会提出自己的评价指标和客观评估公式，导致不同方法难以进行比较。GENEA Challenge 中通过动作的真实性和动作与语音的相关性来评估动作。现有方法主要从动作质量、多样性、与语音的相关性三个方面评估生

成的手势动作，这三方面的评估方法分为客观评估和主观评估（图 2.10）。现有方法主要会结合两种方式对算法效果进行考察。客观评估主要是通过一些公式或者模型计算生成动作序列的评分，从而判断动作在动作质量、多样性和语音相关性上的表现。客观评估方法的优势在于实验结果易复现、成本低、计算快，但不同论文里选择的衡量公式不一样，现有评估公式面临结果可能不准确、无法只通过某一个评估指标就判断方法好坏等问题。例如，通过距离函数计算生成序列与真实序列的距离来确定动作的质量，当两者距离很小时，说明动作质量很高，但是可能影响动作的多样性。如果某个动作在多样性方面表现很好，得分很高，可能是动作异常或者出现了剧烈抖动等，从而导致其动作质量很低。主观评估主要是通过用户调研来完成，相比于客观评估公式，其结果更真实可信，但劣势在于实验成本高且实验结果不易复现。

动作质量用于评估动作的真实性。Freeform(Xu 等, 2022) 提出了训练一个二分类器的方法，用于从伪造的动作序列中鉴别真实的样本，预测分数越高，说明序列越真。Audio2Gestures(Li, Kang 等, 2021) 提出了两种评估生成手势质量的方法。一种是计算预测关节点位置和真实关节点位置的平均距离，可选择使用 L1 范数或者 L2 范数。另一种方法，先设置一个阈值，若预测关节点和真实关节点的欧式距离小于这个阈值，则认为该关节点预测正确，计算正确关节点的比例。受启发于图像生成任务的评估方法 FID(Frechet Inception Distance), AI Choreographer(Li, Yang 等, 2021)、Trimodal(Yoon 等, 2020)、DAE(Kucherenko 等, 2019) 等方法则通过计算生成动作和真实动作在特征空间的距离来评估运动质量。AI Choreographer(Li, Yang 等, 2021) 分别提取了动作的速度、加速度等动态特征和几何特征。Trimodal (Yoon 等, 2020) 则是基于自编码训练了一个特征提取器。

动作的多样性可从两个层面进行评估。一方面指语音动作具有一对多的映射关系，即一段语音可以生成多个不同的动作。为此，AI Choreographer(Li, Yang 等, 2021)、Freeform(Xu 等, 2022)、Audio2Gestures(Li, Kang 等, 2021) 等方法通过一个音频生成了多个不同的动作序列，并计算这些序列之间的平均距离，距离越大，则说明生成的动作越丰富。另一方面，动作多样性也可以从长序列中不同动作的数量来评估。类似于 RNN 中自回归机制的问题，这种机制存在错误累计的缺点，导致生成的动作逐渐趋向于静态动作。因此，Audio2Gestures(Li, Kang 等,

2021) 将生成的动作序列划分成不同的片段，并通过计算这些片段的平均 L1 距离评估长序列中产生的不同动作数量。

语音相关性既可以体现在动作与音频动态节奏的相关性上，如根据音频的重音做一些节拍性的手势，也可以体现在动作与音频语义的相关性上，如通过某个手势动作表达语言想要表达的含义。在音乐生成舞蹈领域，多用节奏相关性衡量音乐舞蹈是否匹配，如图 2.11 所示，AI Choreographer(Li, Yang 等, 2021) 通过计算动作节拍和音频节拍的匹配度评估两者的相关性。其中动作节拍用速度的局部最小值表示，音频节拍可以使用 Librosa 库 (McFee 等, 2015) 提取。目前尚未在现有出版物中看到用客观评估公式来评估动作的语义相关性。

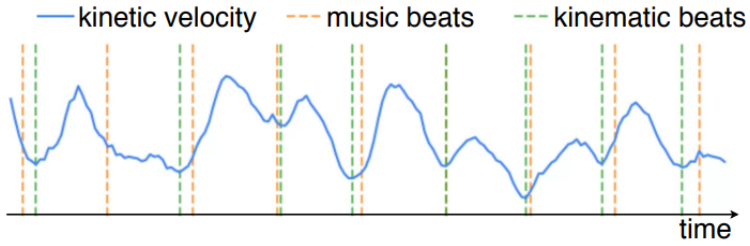


图 2.11　计算动作节拍和音频节拍的距离 (Li, Yang 等, 2021)

Figure 2.11　Distance calculation between motion beats and audio beats(Li, Yang 等, 2021)

对于动作序列的主观评估方法，目前主要依赖于受试者的人工评分。例如，Freeform(Xu 等, 2022) 采用了视觉真实度、语音动作是否同步、动作表现力等指标，让用户选出喜欢的动作序列。Audio2Gestures(Li, Kang 等, 2021) 则通过设置问卷调查的方式，调研用户对动作序列的偏好程度。问卷包含 4 个由不同方法生成的 20s 的视频，音频相同，用户需要根据动作的真实性、动作细节的丰富程度、动作与音频的匹配程度等指标，按照 5 个等级{最佳（best）、好（fine）、一般（not bad）、差（bad）、最差（worst）}对视频进行评分。DAE(Kucherenko 等, 2019) 采用了类似的问卷评估方法，针对动作的自然度、时序一致性和语义一致性设置了 3 个问题，如图 2.12 所示，每个问题的评分范围为 1 至 7, 对应于强烈反对至强烈同意]。此外，GENEA Challenge 在 2020 年和 2022 年的比赛中也采用了用户偏好调查来评估较参赛结果。其中，GENEA Challenge 2022 改进了主观评估的方法，使得调查结果更加真实可信，详见章节 2.2.1。

Scale	Statement (translated from Japanese)
Naturalness	Gesture was natural Gesture was smooth Gesture was comfortable
Time consistency	Gesture timing was matched to speech Gesture speed was matched to speech Gesture pace was matched to speech
Semantic consistency	Gesture was matched to speech content Gesture well described speech content Gesture helped me understand the content

图 2.12　DAE 中动作评估的相关问题 (Kucherenko 等, 2019)

Figure 2.12　Issues related to gesture evaluation in DAE(Kucherenko 等, 2019)

2.2.5　音频特征

原始音频信号是一种连续信号，其在时间和幅度上都是连续的。通过按照指定的采样率对原始信号进行采样，可以得到一个离散信号。然而，离散信号仍包含大量采样点，如 Librosa 库 (McFee 等, 2015) 中默认采样率是 22,050，即 1s 内采样 22,050 个采用点，若音频长达几十秒，则采样点的数量会非常庞大。为了减少音频的冗余信息，现有方法通常会先提取音频的基础特征，将这些基础特征输入到网络中学习。常见的音频特征有梅尔频率倒谱系数 (Mel Frequency Cepstral Coefficients, MFCCs), 频谱图 (Spectrogram), 韵律特征 (Prosodic features)，音高（Pitch）。

2.2.5.1　梅尔频率倒谱系数

梅尔频率倒谱系数是一种用于音频信号处理的特征提取方法。它是一种基于人耳听觉特性的频域特征提取方法，用于语音识别、说话人识别等任务。提取 MFCC 特征的过程如下：先将音频信号进行预加重，分帧和加窗操作；对每一帧的信号进行快速傅里叶变换（Fast Fourier Transform, FFT），将时域信号转换为频域信号；由于人耳对声音的频率分辨率并不是线性的，因此需要使用梅尔滤波器组来模拟人耳的听觉特性；对于每个滤波器输出的能量值，取其对数，得到的结果称为梅尔频率倒谱系数；对于每帧的梅尔频率倒谱系数，进行离散余弦变换（Discrete Cosine Tranform，DCT），得到 MFCCs。MFCC 特征如图 2.13 所示，

横轴表示时间，纵轴表示 MFCC 系数，颜色表示强度。只需指定 MFCC 系数数量，现有的 Librosa 库，python_speech_features 库就能直接提取出音频的 MFCC 特征。Freeform(Xu 等, 2022) 中提取了音频的 MFCC 特征作为动作动态节奏生成分支的输入。

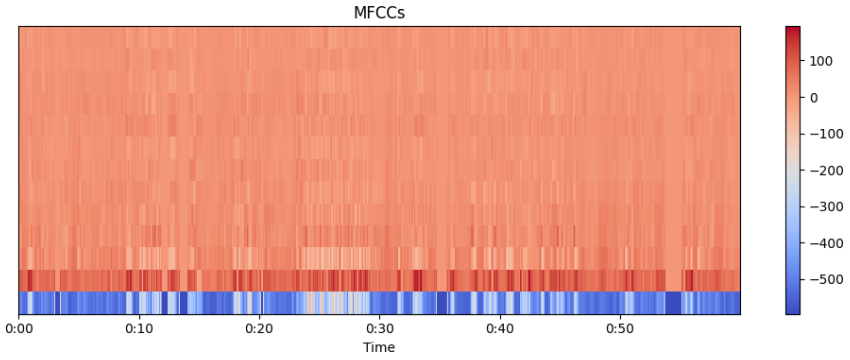


图 2.13　MFCCs 可视化

Figure 2.13　Visualization of MFCCs

2.2.5.2　频谱图

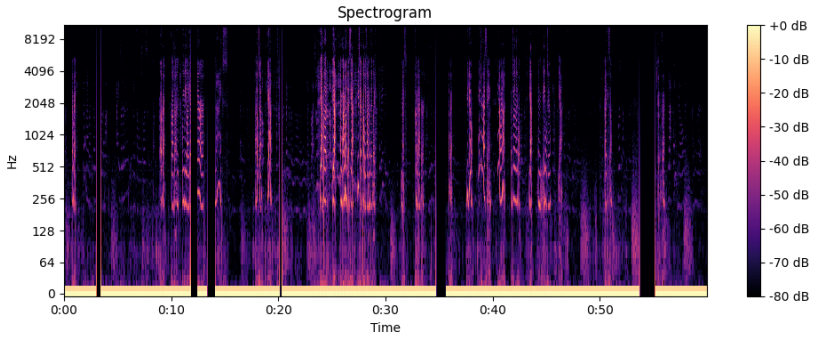


图 2.14　音频频谱图可视化

Figure 2.14　Audio spectrogram visualization

频谱图（Spectrogram）如图 2.14 所示，横轴表示时间，纵轴表示声音的频率，颜色表示对应时间和频率下的信号能量或强度。先将信号在时间轴上切分成一段段小时间窗，对每个时间窗进行傅里叶变换，然后将每个时间窗对应的频谱结果拼接成一个二维图像即可得到频谱图。音素是语音中最基本的语音单位，是语音信号中最小的可以区分语义的音段。频谱图和音素之间存在一定的关系。在语音信号中，不同的音素通常在不同的频率范围内有不同的能量分布。通过

分析语音信号在频域上的能量分布情况，可以帮助识别不同音素的特征。例如，清音的/p/和浊音的/b/在声带振动时产生的谐波分布不同，因此在频谱图中可以看到它们在不同的频率上有不同的能量分布。通过这种方式，可以根据不同频率特征来判断语音信号中所包含的音素。频谱图广泛应用于信号处理、语音识别、音频分析等领域。

2.2.5.3 韵律特征

韵律特征（Prosodic features）是指语音信号中的语调、节奏、语速等特征。常用的韵律特征包括基频、能量、时长、语速等。基频（Fundamental Frequency, F0）指语音信号中的主频率，即声音振动周期的倒数。基频通常用于表达说话人的语调、情感等信息。能量（Energy）指语音信号在频域中的总能量，反映了语音信号的响度大小。时长（Duration）指语音信号的时间长度，反映了发音的持续时间。语速（Speaking Rate）指说话人每分钟说话的单词数或音素数。在音乐信号方面，音频的韵律特征包括色度特征（Chroma）、音频峰值（Peak）、音频节拍（Beat）等。色度特征描述了不同音高在音乐中的相对强度和分布。音频峰值是音频信号的最大幅度值，通常用于描述音频的音量。音频节拍是指音乐中的基本节拍或节奏感，可以用于衡量音乐的速度和节奏特征。音乐信号特征和语音信号特征在一定程度上具有联系，如音频峰值和语音信号的能量都可以用于描述信号的响度或强弱，音频节拍和语速都可以用于描述信号的节奏感和速度，基频和色度特征都涉及音高的计算和分析。

2.2.5.4 音高

音高特征如图 2.15 所示，横轴表示时间，纵轴表示频率，颜色表示音高。音高是指人类感知音乐时，通过听到的声音中辨别出的音调的高低。它是音乐中非常重要的特征之一，决定了音乐的基础和和谐。在音频信号中，音高通常通过基频表示。基频是声音中最显著的频率成分，它是声音的主要周期性变化。除了基频，梅尔频率倒谱系数（MFCCs）也可以用于表示音高。MFCCs 可以将音频信号转换为一组系数，其中每个系数对应着一段时间内的音高、音量和声音的频谱形状。除了 MFCCs，其他如 YIN 算法、自适应谱估计（Adaptive Spectral Estimation，ASE）等方法都可以用于提取音频信号的音高特征。

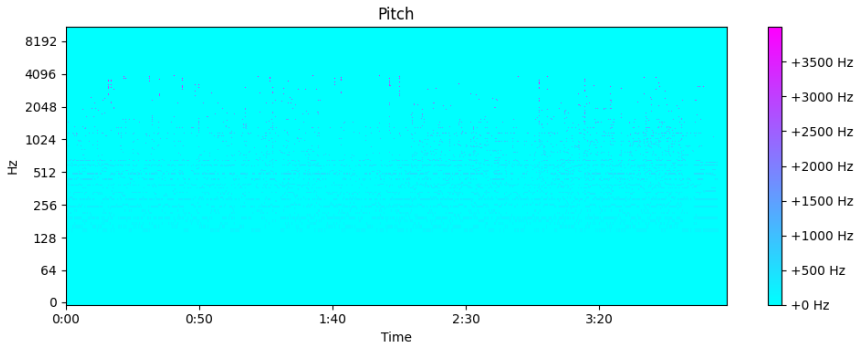


图 2.15　音频的音高特征

Figure 2.15　Pitch of audio

2.2.5.5　音频特征使用

在 Forgerons(Ghorbani 等, 2022) 中，采用 50ms 汉宁窗口和 12.5ms 步幅（Hop Length）提取频谱图，将其投影到梅尔频率刻度，并使用每个通道的对数幅度。在 IVI Lab(Chang 等, 2022) 中，使用梅尔谱图 (Mel-Spectrograms)、梅尔频率倒谱系数 (MFCCs) 和音律特征作为音频的基础特征，其中音律特征包括音频强度 (Intensity)、音高及这它们的导数。而 Freeform(Xu 等, 2022) 只使用 MFCC 作为音频特征。DSI(Saleh, 2022) 认为 MFCC 仅描述了一帧语音上的谱包络，语音信号还包含一些动态信息，因此采用一阶差分和二阶差分作为基础语音特征。UEA Digital Humans(Windle 等, 2022) 使用 PASE(Ravanelli 等, 2020)(The problem-agnostic speech encoder，问题无关语音编码器) 提取音频特征。PASE 是一种基于多个自监督任务的语音表示训练的模型，这些任务从多个角度从语义中提取有用的信息，使得模型可以学习到问题无关的语音特征 (problem-agnostic feature)。相对于传统的语音特征如 MFCC，PASE 的特征可以显著提高模型性能。在 GestureMaster(Zhou 等, 2022) 中，使用音频中的脉冲表示音频节拍。在音乐生成舞蹈领域，AI Choreographer(Li, Yang 等, 2021) 用到了音频包络、MFCC、色度特征 (Chroma)、音频峰值、音频节拍等特征。

DAE(Kucherenko 等, 2019) 做实验验证了梅尔频率倒谱系数、频谱图、韵律特征作为网络输入的效果。实验中，在频谱图中移除了小于 20Hz 或大于 8000Hz 的频率，因为这些频率携带的语义信息较少。韵律特征使用了音高和能量，这些特征中的信息具有较低的比特率，不足以区分任何单词，但对于预测一些有节奏的手势可能会提供信息。最终得到的结果如图 2.16 所示，MFCCs 实现了

最低的 APE, 但是相比于频谱特征产生了更大的加速度和抖动。APE（Average Position Error）计算了预测序列的关键点和真实序列关键点之间的平均欧式距离，如式 2.2，其中 T 表示序列的持续时间，D 表示动作数据的维度，n 表示序列索引。

Model/feature	APE	Acceleration	Jerk
Static mean pose	8.95	0	0
Prosodic	8.56±0.2	0.90±0.03	1.52±0.07
Spectrogram	8.27±0.4	**0.51**±0.07	**0.85**±0.12
Spectr. + Pros.	8.11±0.3	0.57±0.08	0.95±0.12
MFCC	**7.66**±0.2	0.53±0.03	0.91±0.05
MFCC + Pros.	**7.65**±0.2	0.58±0.06	0.97±0.11
Baseline [13] (MFCC)	8.07±0.1	1.50±0.03	2.62±0.05
Ground truth	0	0.38	0.54

图 2.16　MFCC，频谱图，韵律特征比较 (Kucherenko 等, 2019)

Figure 2.16　MFCC, Spectrogram, Prosodic feature comparison

$$APE(g_t^n, \hat{g}_t^n) = \frac{1}{DT}\sum_{t=1}^{T}\sum_{d=1}^{D}\|g_t^n - \hat{g}_t^n\|_2 \quad \ldots (2.2)$$

2.2.6　动作表示

对于不面向 3 维虚拟人模型的 2D 动作序列和 3D 动作序列通常使用关节点位置控制动作变化。然而，在 3D 空间中，只使用物体位置描述刚体的位姿是远远不够的，还需要考虑物体的旋转，如在同一个位置的物体可能有不同的朝向。关节点的旋转常用欧拉角、四元数、旋转矩阵等方法表示。

欧拉角是通过三个旋转角度来表示旋转姿态的方法。这三个旋转角度分别为绕 X 轴的角度（Roll），绕 Y 轴的角度（Pitch）和绕 Z 轴的角度（Yaw），每个角度的范围为 0 到 360 度。不同旋转顺序会导致结果不同，通常有 3 种不同的约定，分别为 XYZ、ZYX 和 YXZ。例如，欧拉角 (α, β, γ) 表示先绕 Z 轴旋转 γ 角度，再绕 Y 轴旋转 β 角度，最后绕 X 轴旋转 α 角度。虽然欧拉角比较直观，易于理解和可视化，但是其存在万向锁等问题，即在某些特殊情况下，两个欧拉角表示的旋转姿态是相同的。

四元数是一种使用四个实数来表示旋转姿态的方法。它们的形式为（w, x, y,

z），其中 w 是一个实数，x、y 和 z 是虚数，可以表示为 $w + xi + yj + zk$。四元数可以通过向量规范化来实现单位四元数的要求，单位四元数可以避免由于舍入误差而引起的旋转变形。四元数可以消除万向锁问题，还可以用于球面线性插值（Spherical linear interpolation, Slerp）(Shoemake, 1985)。

旋转矩阵是一种使用 3×3 矩阵来表示旋转姿态的方法，其中每一行（或每一列）表示旋转后的坐标轴。旋转矩阵可以表示旋转和变换，且旋转顺序不会影响结果，但是矩阵运算相对较慢。旋转矩阵和四元数，欧拉角可以相互转化。例如，对于欧拉角 (α, β, γ)，其对应的旋转矩阵可以表示为式 2.3。对于四元数（w, x, y, z），其对应的旋转矩阵表示为式 2.4。

$$R = R_z(\gamma) * R_y(\beta) * R_x(\alpha) \quad \ldots (2.3)$$

$$R = \begin{bmatrix} 1 - 2y^2 - 2z^2 & 2xy - 2wz & 2xz + 2wy \\ 2xy + 2wz & 1 - 2x^2 - 2z^2 & 2yz - 2wx \\ 2xz - 2wy & 2yz + 2wx & 1 - 2x^2 - 2y^2 \end{bmatrix} \quad \ldots (2.4)$$

由于欧拉角，四元数是不连续的表示，Zhou 等 (2019) 认为不连续的表示会造成监督信号的不连续，不利于神经网络学习。旋转矩阵虽然是连续的表示，但是其含有冗余特征，于是 Zhou 等 (2019) 提出了 6D 的旋转表示，即先将角度的其他表示方式转换成的 3×3 旋转矩阵，再取旋转矩阵的前两列值作为动作数据的表示。

2.3　本章小结

本章介绍了跨模态序列生成和语音生成手势的现状。对于跨模态序列生成，涉及了适用场景、研究方法以及研究的重点和难点。在语音生成手势方面，本章从数据集、音频生成手势的方法、评估指标、音频特征和动作表示等方面进行了介绍和分析。

第 3 章　虚拟人驱动

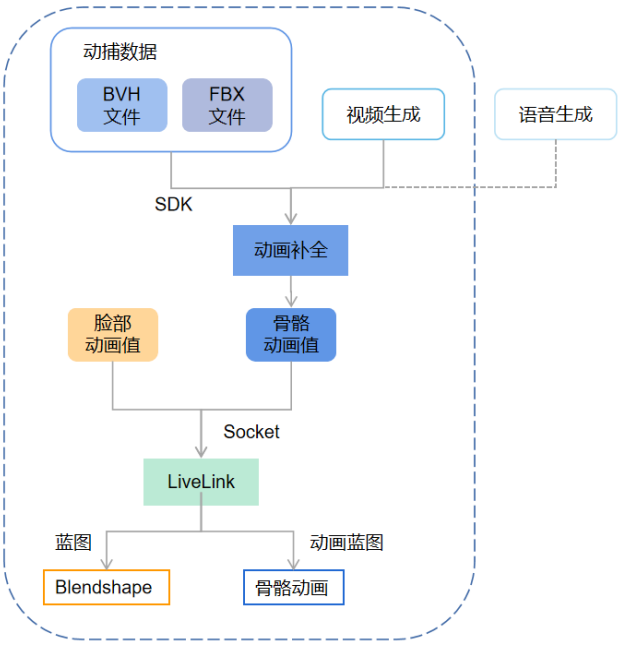


图 3.1　虚拟人驱动流程

Figure 3.1　Virtual human-driven processes

为了提高虚拟人的真实性和可交互性，本文致力于研究虚拟人的动画生成，图 3.1 是本文所实现的虚拟人驱动流程。本章节将主要讲述蓝色虚线圈出的流程，语音生成是为了弥补动捕数据、视频生成的不足而提出的，将在第 4 章讲述。本章节既实现了虚拟人脸部动画（Blendshape）的驱动，又实现了骨骼动画的驱动。其中骨骼动画分别采用了动捕驱动、视频驱动两种方式。在图 3.1 展示的流程中，动捕驱动和视频驱动的流程大致相似，即先获取动画数据，然后进行动画补全获取到完整的骨骼动画值，将动画值通过 Socket 发送给虚幻引擎，虚幻引擎通过 Livelink 插件接收数据并使用动画蓝图将数据赋值给指定虚拟人得到骨骼动画。动捕驱动和视频驱动的主要区别在于骨骼动画的数据来源不一样。动捕驱动的数据源于 Biovision Hierarchy（BVH）文件和 Filmbox（FBX）文件，这两个文件用于存储动捕数据。视频驱动的数据源于视频文件，对其采用姿态估计、关节点检测等算法得到骨骼动画数据。本章节将从脸部驱动、动捕驱动骨骼、视频驱动骨骼三个方面进行介绍。

3.1　脸部驱动

虚拟人的面部表情动画采用了 Blendshape 技术。Blendshape 也称为形状插值动画 (Shape Interpolation Aniamtion)，是一种基于几何变换的技术，通常用于表现角色面部表情变化。在实际的工程实现中，会在虚拟人的面部设置一些关键点，用于控制人物局部的表情变化。在虚幻引擎中，这些关节点的数值范围为 0-1，表示脸部指定区域的变化幅度，如当右眼的值设为 0 时，表示右眼完全睁开（如图 3.2 所示），当值设为 0.5 时，表示右眼半睁开（如下图 3.3 所示），当值设置 1 时，表示右眼完全闭上（如图 3.4 所示）。

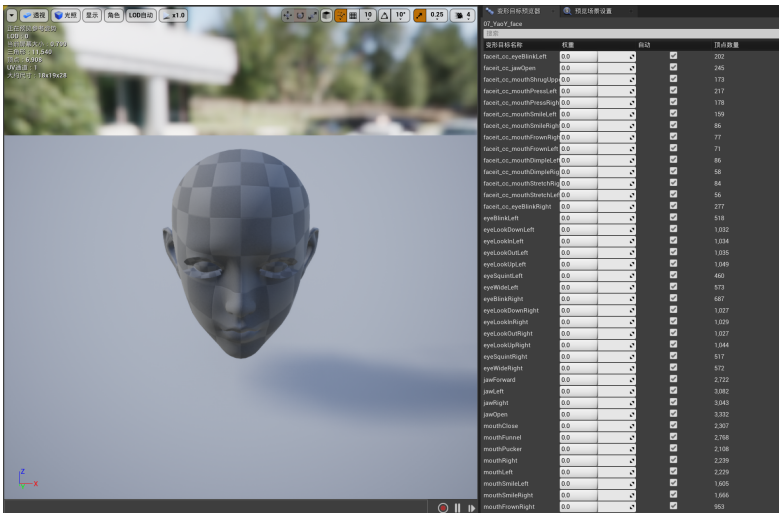


图 3.2　变形目标为 0 时的脸部表情

Figure 3.2　Facial expression when all morph targets are 0

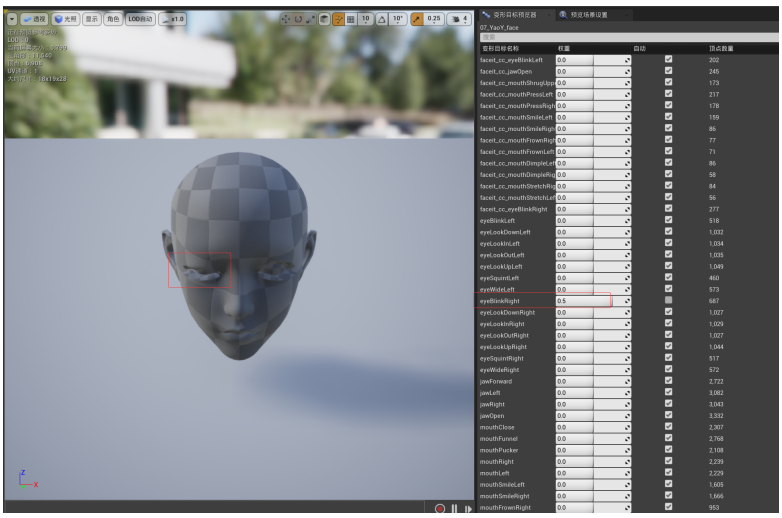


图 3.3　eyeBlinkRight=0.5 时的脸部表情

Figure 3.3　Facial expression when eyeBlinkRight=0.5

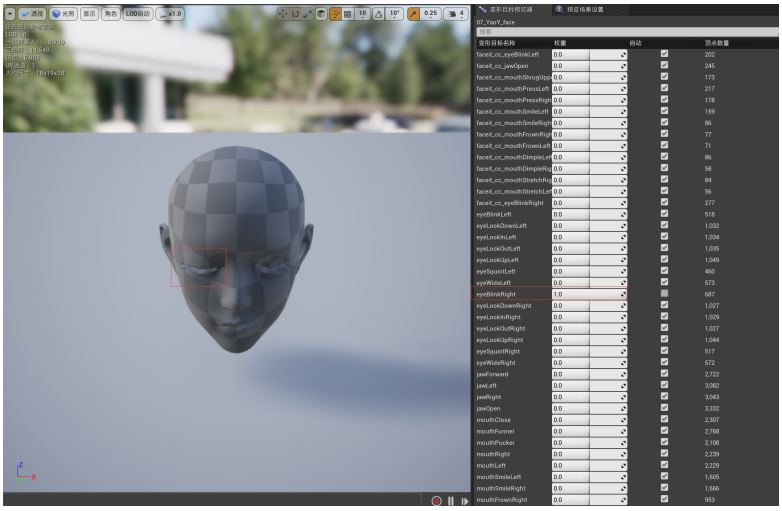


图 3.4　eyeBlinkRight=1 时的脸部表情

Figure 3.4　Facial expression when eyeBlinkRight=1

若要实现脸部动画的编程实现，需要先通过 LiveLink 插件接收到来自其他程序的动画数据，并将这些数据值重定向到指定的变形目标上。虚幻引擎中的 LiveLink 插件是一个用于实时数据捕捉和共享的插件。它的主要作用是将来自外部设备的实时数据直接输入到虚幻引擎中，使得虚幻引擎可以实时地反映出这些数据的变化。虚幻引擎中的蓝图是一种视觉化编程工具，其由多个节点组成，每个节点代表一种操作或函数调用。用于驱动虚拟人面部动画的蓝图如图 3.5 所示。

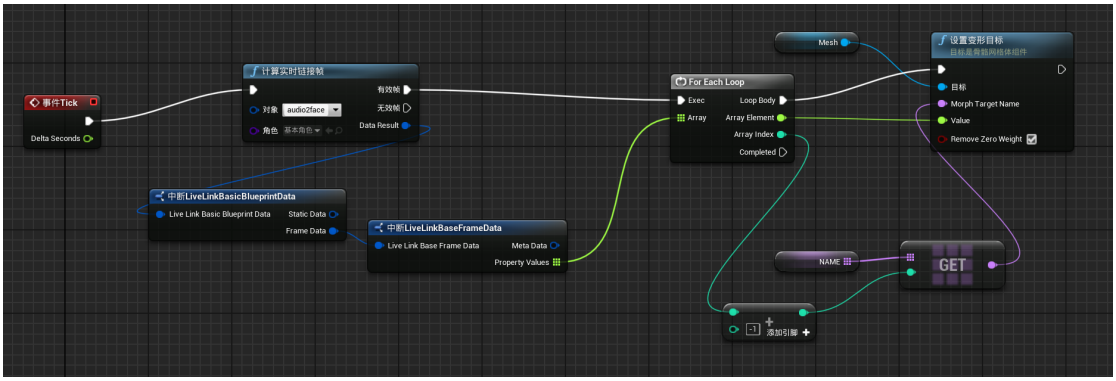


图 3.5　驱动虚拟人面部动画的蓝图

Figure 3.5　A blueprint for driving virtual human facial animation

3.2　动捕驱动骨骼

对虚拟人身体的驱动采用了骨骼动画技术。骨骼动画 (Skeletal Animation) 是一种基于骨骼结构的技术，也是数字动画制作中最常见的技术之一。在虚幻引擎中，虚拟人的骨骼设置如图 3.6 所示，其骨架结构由骨骼树构成，树中的每个节点代表一个关节点，每个关节点都有一个位置值、旋转值和缩放值，这些值决定了这个关节点在当前时刻的状态。例如当更改右臂的旋转值时，其更改后的状态如图 3.7 所示。骨骼动画主要受到各个关节点的动画曲线控制，每个关节点的位置、旋转、缩放的 X、Y、Z 三个分量通常都有一条动画曲线，保存了每个分量在不同时间的动画值，其中 X、Y、Z 分量分别代表三个方向的运动。虚幻引擎在给关节点赋值时，一个时刻只能为每个动画分量指定一个值。因此，在驱动虚拟人骨骼动画的流程中，其关键的一步就是将动画曲线值转换成指定的动画格式（如图 3.8），即指定关节点名称、父节点信息、关节点当前时刻的位置值、旋转值和缩放值。不论是动捕数据驱动、视频驱动还是语音驱动，都是利用了这个思想，即先获取动画曲线值，再转换为指定的格式，通过已实现的流程驱动虚拟人。获取到指定格式的动画数据后，需要将其通过 Socket 发送给虚幻引擎，虚幻引擎通过 LiveLink 插件接收到其他程序发送的动画数据，然后使用动画蓝图将这些动画数据值重定向到指定的关节点中。动画蓝图是虚幻引擎中的一个工具，用于创建和编辑角色动画逻辑。用于驱动虚拟人骨骼动画的动画蓝图如图 3.9 所示。

图 3.6　默认骨架结构

Figure 3.6　Default skeleton structure

图 3.7　$Rotation_z(RightArm) = 40$ 时对应的骨骼状态

Figure 3.7　The bone state when $Rotation_z(RightArm) = 40$

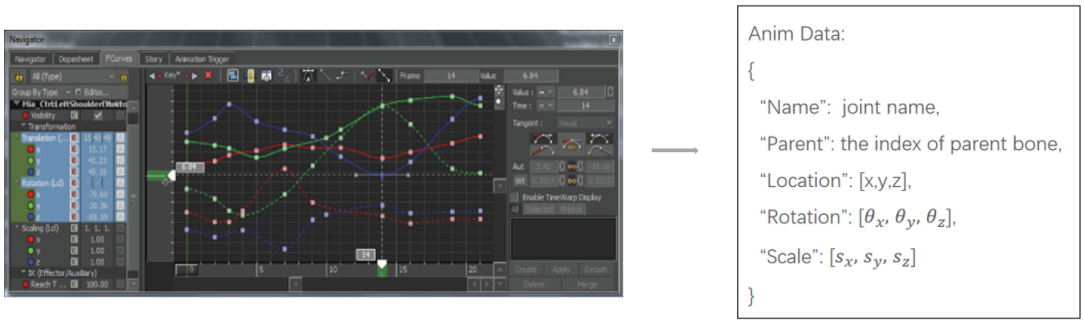


图 3.8　骨骼动画处理流程

Figure 3.8　The process of skeletal animation

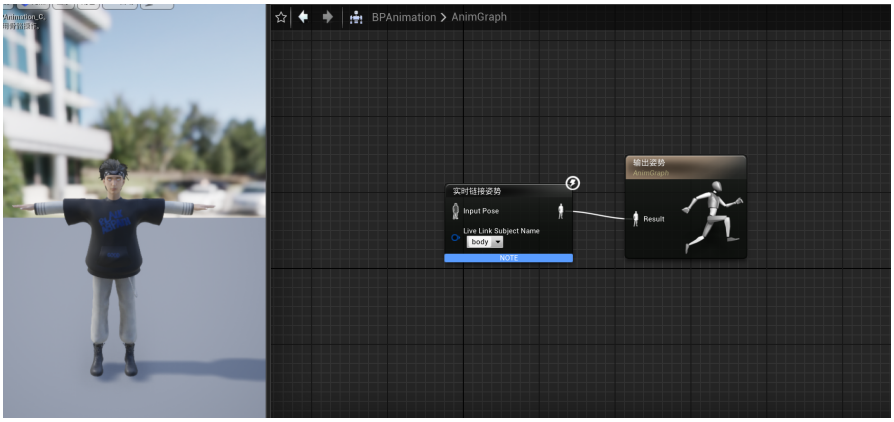


图 3.9　驱动骨骼动画的动画蓝图

Figure 3.9　Animation blueprint that drive skeletal animations

3.2.1　动画值获取

在实际应用中，常用 BVH 文件和 FBX 文件作为存储骨骼动画数据的格式。在动捕驱动流程中，需要通过相关的软件开发工具包（Software Development Kit，SDK）从动画文件中读取到动画值。

BVH 是一种用于存储骨骼动画数据的文件格式，最初是为了存储运动捕捉（Motion Capture）数据而设计的。BVH 文件以文本格式存储，包含了骨骼的层次结构和每个关键帧的旋转角度。在 BVH 文件中，骨骼的层次结构是通过缩进来表示的，每个骨骼都有其名称、父骨骼的名称和旋转角度。BVH 文件的优点是文件大小相对较小，易于使用文本编辑器进行编辑，也方便程序读取和解析。但它的缺点是不支持存储网格信息和贴图等其他模型数据，也不支持动画的蒙皮权重 (Skin Weight) 等高级功能。

```
HIERARCHY
ROOT Hips
{
   OFFSET 11.883895874023438 83.49174880981445 -17.31802749633789
   CHANNELS 6 Xposition Yposition Zposition Zrotation Xrotation Yrotation
   JOINT RightUpLeg
   {
      OFFSET -11.883895874023438 0.0 0.0
      CHANNELS 3 Zrotation Xrotation Yrotation
      JOINT RightLeg
      {
         OFFSET 0.0 -40.196685791015625 0.0
         CHANNELS 3 Zrotation Xrotation Yrotation
         JOINT RightFoot
         {
            OFFSET 0.0 -43.29506301879883 0.0
            CHANNELS 3 Zrotation Xrotation Yrotation
            End Site
            {
               OFFSET 0.0 0.0 17.31802749633789
            }
         }
      }
   }
}
MOTION
Frames: 199
Frame Time: 0.03333333333333333
-0.0 83.83618927001953 0.0 -1.668219474048889 -4.831458422548677 -40.14683288491714 7.
-0.025091553106904 83.83618927001953 0.20021668076515198 -1.794467368246954 -5.051
-0.07304687798023224 83.83618927001953 0.1479644775390625 -1.888470238871281 -5.1394
-0.09923095256090164 83.83618927001953 0.03290100023150444 -1.9200611405688468 -5.14
-0.095367431640625 83.83618927001953 0.06017303839325905 -1.9651039453757755 -5.1693
-0.11008300632238388 83.83618927001953 -0.09192809462547302 -2.05890507893023 -5.205
```

图 3.10　BVH 文件示例

Figure 3.10　Example BVH file

BVH 文件格式如图 3.10 所示。一个 BVH 文件中通常含有 HIERARCHY，ROOT，JOINT，OFFSET，CHANNELS，MOTION，FRAMES，FRAME TIME 等关键字。HIERARCHY 用于指定动作数据的骨骼结构。ROOT 用于定义骨骼

的根节点，即整个骨架结构的起点。JOINT 用于定义骨骼的节点，可以有多个 JOINT 组成骨架结构。OFFSET 指定了当前骨骼节点相对于其父节点的位移量，以 XYZ 轴为单位。CHANNELS 定义了每个骨骼节点上可以应用的运动通道，包括旋转和位移，用来描述骨骼节点的运动。MOTION 部分包含了动作数据，用于描述骨骼节点随时间变化的旋转和位移信息。FRAMES 指定了动作数据的帧数。FRAME TIME 指定了每一帧的时间长度。

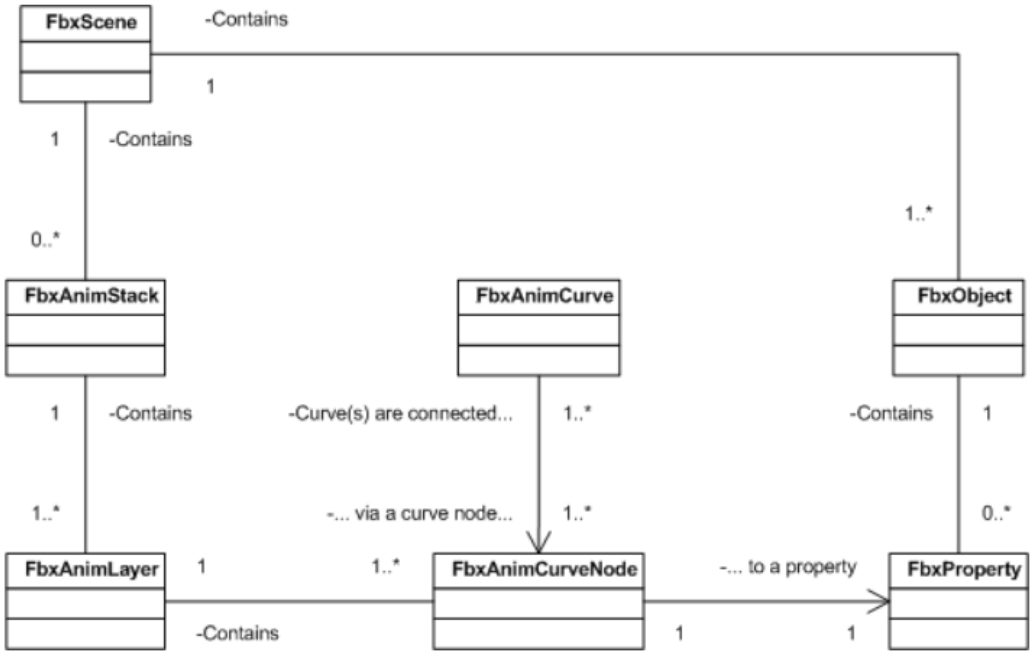


图 3.11　FBX 中动画元素关系 (Autodesk, n.d.)

Figure 3.11　Animation element relationship in FBX(Autodesk, n.d.)

FBX 是由 Autodesk 开发的一种文件格式，以二进制格式存储，可以存储骨骼动画、形变动画、关键帧动画等多种类型的动画数据。相比于 BVH 文件，但的文件大小相对较大，不便于使用文本编辑器进行编辑和查看。FBX 文件采用树状结构存储虚拟人的骨骼结构。除了根关节点外，每个关节点都有一个父级关节点，两个相邻的关节点形成一块骨骼。关节点的基本信息包括关节点名称、父级关节点 ID、关节点 3D 位置、关节点 3D 旋转值和关节点 3D 缩放值。FBX 骨骼链上存储着每个关节点的默认值，这些值只有一帧，不会产生动画效果。在 FBX 中，动画数据由多个层次结构组成，包括 FbxScene、FbxAnimStack、Curve Node 和 LclTranslation、LclRotation、LclScaling，这些层次结构的关系如图 3.11 所示。FbxScene 代表一个场景，包含多个节点，如模型、灯光、相机等，

其包括一个或多个 FbxAnimStack。FbxAnimStack 代表一个完整的动画时间线，一个 FbxAnimStack 包含多个 FbxAnimLayer。FbxAnimLayer 代表一层动画，在 FbxAnimLayer 中，可以添加、编辑、删除动画曲线，以及设置曲线的权重、静态值等。Curve Node 代表一个被动画控制的属性，例如对象的位置、旋转、缩放等。每个 Curve Node 都与一个属性（例如 LclTranslation、LclRotation、LclScaling）相关联，它们描述了动画曲线如何影响这些属性的值。LclTranslation、LclRotation、LclScaling 分别代表对象的位置、旋转、缩放属性。这些属性通常被动画控制，可以在 FbxAnimLayer 中添加动画曲线来控制它们的值。Autodesk 为 FBX 文件提供了开源的 SDK，其为这些层次结构提供了代码实现，通过相关函数接口，可以从 FBX 文件中读取动画数据，进而通过我们的驱动流程发送给虚幻引擎，驱动虚拟人。

3.2.2　动画补全

在驱动虚拟人时，期待获取的完整数据如图 3.12 所示，包含每个关节点的名称、父节点信息、关节点的位置、旋转角度、缩放信息。每个骨骼按照骨骼结构完全展开的顺序自上而下编号，因此每个关节点都对应着一个唯一的索引。父节点信息即父节点对应索引。这些信息存在冗余，由于运动学等概念的存在，可以根据一些已知的关节点信息补全其余的数据。在工程实现中，为了节省存储的数据信息，部分导出的 FBX 文件只有根节点的全局位置，和所有节点的旋转信息，缺失每个节点需要用到的 3D 位置信息，需要对数据进行补全。

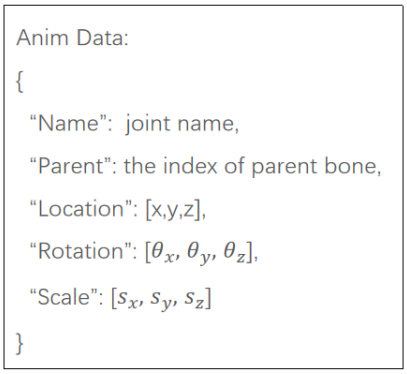


图 3.12　驱动虚拟人的动画数据格式

Figure 3.12　Animation data format for driving avatars

正向运动学是机器人学中的一个重要问题，它是将机器人各个关节的状态信

息转换为机器人末端执行器的位置和姿态信息的过程。正向运动学方程如式 3.1 所示，T 表示机器人末端执行器在基座坐标系下的位姿矩阵，$T_0^1, T_1^2, \ldots, T_{n-1}^n, T_n^E$ 分别表示机器人相邻关节之间的转换矩阵，n 为机器人的自由度，E 为机器人末端执行器坐标系。

$$T = T_0^1 T_1^2 \ldots T_{n-1}^n T_n^E \quad \ldots (3.1)$$

一般来说，虚拟人的坐标系可以分为全局坐标系和本地坐标系两种。全局坐标系是虚拟环境中的固定参考系，通常用来描述虚拟场景中的物体位置和方向。在全局坐标系中，物体的位置和方向是相对于虚拟环境的原点和方向而言的。虚拟人的根节点通常使用全局坐标系来描述其位置和方向。若对虚拟人使用正向运动学公式，最终得到的值也是相对于全局坐标系的值。在虚拟人模型中，每个节点通常都有一个本地坐标系，用来描述该节点的位置和方向。在这种情况下，虚拟人中的节点通常使用本地坐标系来描述其相对父节点的位置和方向。除根节点外，图 3.12 涉及的相关数值信息就是使用的本地坐标系的值。因此，若要采用 3.2 章节的数据驱动方式，不能直接使用正向运动学公式。

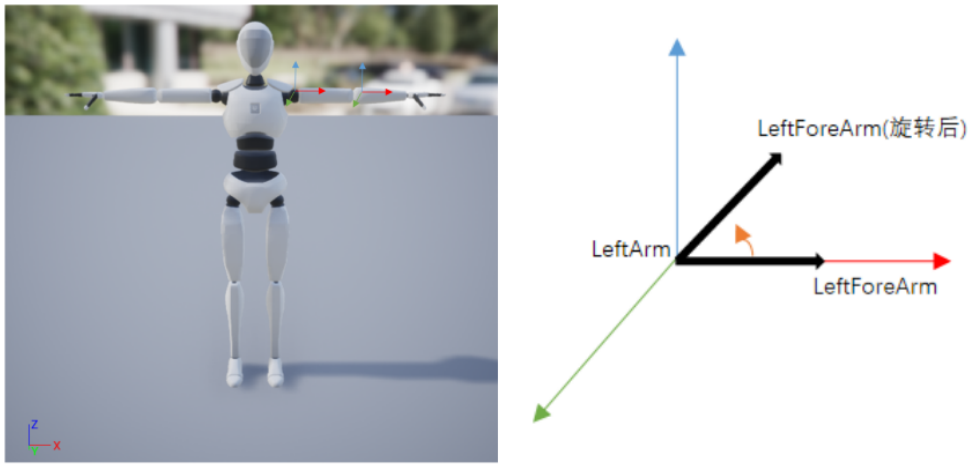


图 3.13　关节点坐标系

Figure 3.13　Joint coordinate system

FBX 使用树状结构存储骨骼链，每个节点代表一个关节点，连接两个节点的边表示一块骨骼。T-pose 状态表示虚拟人模型所有关节点旋转角度为 0 时的状态，通过 T-pose 可以获取每个关节点在旋转角度为 0 时的位置。将每块骨骼视作一个向量，由父关节点指向子关节点。以父节点为原点建立坐标轴，坐标轴方向是骨骼在 T-pose 状态时的方向，如图 3.13 所示。将旋转角度转换成旋转矩

阵，如式 3.2，通过将 T-pose 状态的坐标向量乘以旋转矩阵，可以得到旋转后的坐标向量，从而得到旋转后关节点的坐标位置。默认的缩放值可以设置为 1。因此，动作序列的表示方式可以简化为每一帧根节点的位置，以及各个节点在当前时刻的旋转角度。

$$p' = Rp \quad \ldots (3.2)$$

3.3　视频驱动骨骼

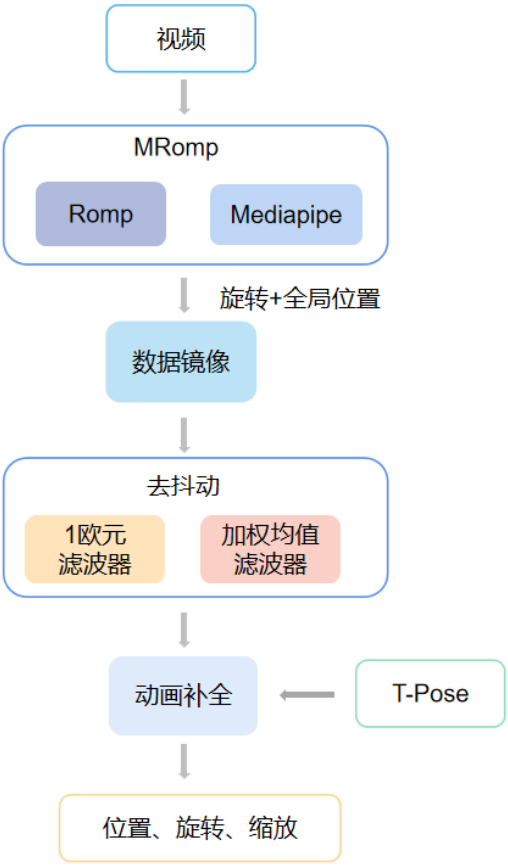


图 3.14　视频生成动画值流程

Figure 3.14　The process of generating animation value from video

用动作捕捉系统驱动虚拟人物的动作存在一些弊端。第一个缺点是其成本高，动作捕捉系统需要高质量的摄像头、传感器和计算机等硬件设备，这些设备的成本较高，同时还需要专业的技术人员来安装和维护。其次，其具有一定的局限性，动作捕捉系统通常只能捕捉已知的运动，例如人体的基本动作或特定的运

动，但是在现实世界中，人的运动是非常复杂和多样化的，有些运动可能无法被准确捕捉。此外，由于动捕系统通常需要一定的时间来处理数据并生成动画，这意味着虚拟人物的动作可能会有延迟，无法实现实时反馈。

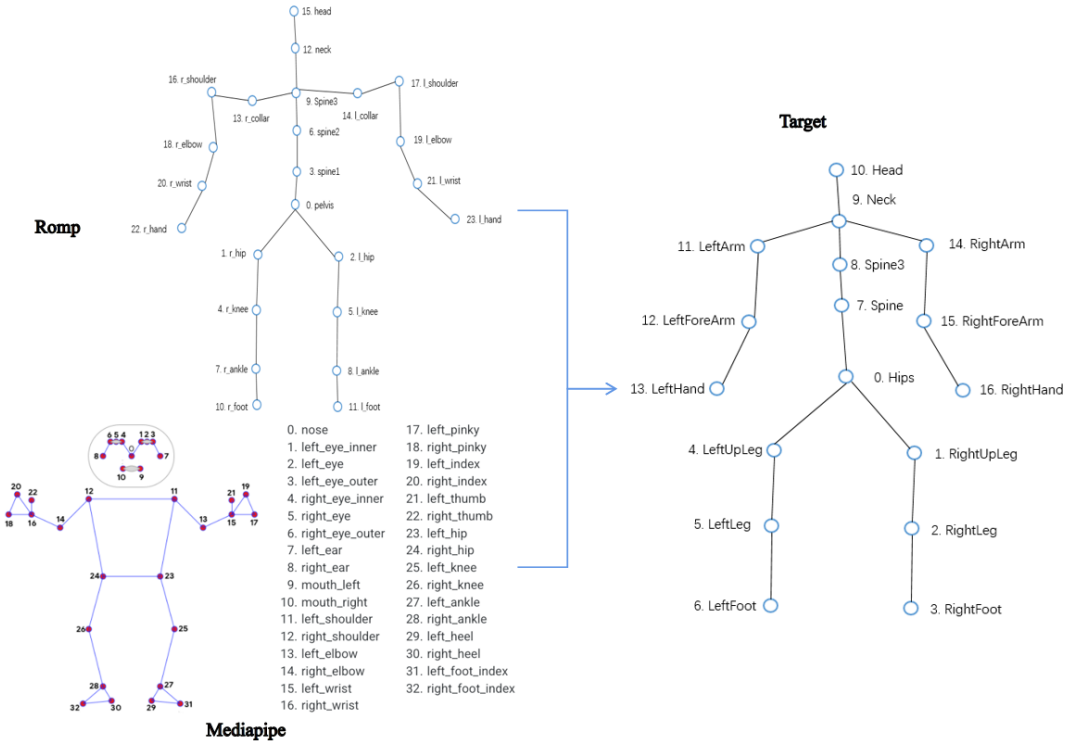


图 3.15　骨骼重定位

Figure 3.15　Bone retargeting

在工程实现中，为了解决动捕系统存在的上述问题，我们尝试过通过视频驱动虚拟人。根据视频估计骨骼动画值的流程如图 3.14 所示，先通过提出的 MRomp 模型从视频中估计出人物的旋转角度和全局位置信息，然后对动画值分别进行数据镜像、去抖动、数据补全操作，即可得到完整的动画值。MRomp 模型结合了 Romp(Sun 等, 2021) 和 Mediapipe(Lugaresi 等, 2019) 两个模型，使用 Romp 估计得到各个关节点的旋转，使用 Mediapipe 估计得到视频中人物的全局位置信息和全局方向信息。如图 3.15 所示，Romp、Mediapipe 和目标驱动的虚拟人骨骼结构均不一样，所以需进行骨骼重定位，即将 Romp 和 Mediapipe 的相关值按照目标虚拟人骨骼结构转换。只需要 Mediapipe 提供根节点的旋转和位置信息，但是 Mediapipe 所使用的骨骼结构并没有直接定义根节点，在实现中，我们按照式 3.3 获取到目标骨骼根节点（Hips）的位置，即虚拟人的全局位置，依次按照式 3.4，式 3.5，式 3.6，式 3.7，式 3.8，式 3.9，式 3.10 获得根节点关于 X，Y，Z

轴的方向向量，对 X_dir，Y_dir，Z_dir 单位化，然后将其按照 XYZ 列顺序构造成 3×3 的旋转矩阵，再将旋转矩阵依次转换成四元数、欧拉角即可。如图 3.15，Romp 使用的骨骼结构其左右骨骼的定义与目标骨骼相反，因此若直接使用模型生成的动画值驱动虚拟人，会导致最终得到的数据是镜像的，如检测视频中人物左手的动作会展示在虚拟人右手上。为了解决镜像问题，在实际使用动画值时，绕 X 轴旋转的角度保持不变，对于绕 Y、Z 轴旋转的角度取相反值。若直接使用前几步得到的动画数据，虚拟人会出现动作抖动的问题，本方法为每一条动画曲线分别构造滤波器，先使用了 1 欧元滤波器 (Casiez 等, 2012) 消除一些小的抖动，再使用加权均值滤波器消除大的抖动。其中使用到的加权均值滤波器的公式如式 3.11 所示，每输入一帧，对该帧及该帧前两帧的动画值进行加权求和，计算得到的值作为当前帧骨骼的平移值或者旋转值。数据补全操作见章节 3.2.2。视频驱动结果如图 3.16 和图 3.17 所示。

$$pos_{Hips} = \frac{pos_{left\_hip} + pos_{right\_hip}}{2} \quad \ldots (3.3)$$

$$pos_{Neck} = \frac{pos_{left\_shoulder} + pos_{right\_shoulder}}{2} \quad \ldots (3.4)$$

$$pos_{Spine} = \frac{pos_{Hips} + pos_{Neck} - pos_{Hips}}{3} \quad \ldots (3.5)$$

$$pos_{LeftUpLeg} = pos_{left\_hip} \quad \ldots (3.6)$$

$$pos_{RightUpLeg} = pos_{right\_hip} \quad \ldots (3.7)$$

$$X\_dir = pos_{LeftUpLeg} - pos_{RightUpLeg} \quad \ldots (3.8)$$

$$Y\_dir = pos_{Spine} - pos_{Hips} \quad \ldots (3.9)$$

$$Z\_dir = X\_dir * Y\_dir \quad \ldots (3.10)$$

$$y(n) = \frac{3x(n) + 2x(n-1) + x(n-2)}{3+2+1} \quad \ldots (3.11)$$

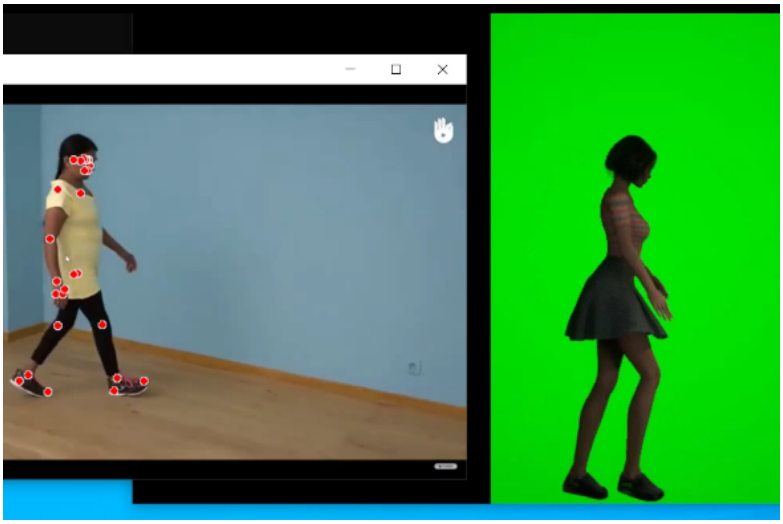


图 3.16　视频驱动结果 1

Figure 3.16　The result 1 of video driven

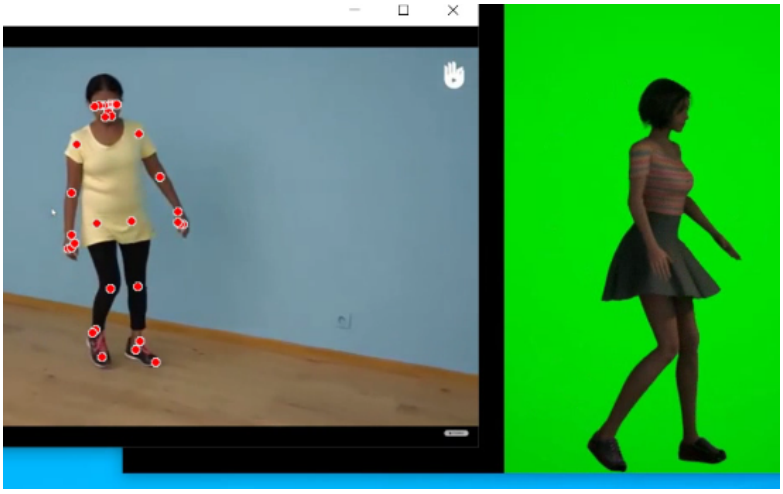


图 3.17　视频驱动结果 2

Figure 3.17　The result 2 of video driven

3.4　本章小结

本章实现了虚拟人驱动流程，其中包括脸部动画的驱动流程、动捕驱动骨骼动画的流程以及视频驱动骨骼动画的流程。通过采用动捕驱动技术，我们明确了虚拟人骨骼动画的驱动流程。然而，动捕驱动是从 BVH 文件或 FBX 文件中获取

动画数据的，这意味着本文设计的动捕驱动流程是离线的，即动画数据需要预先录制并导出到 BVH 或 FBX 文件中，从而限制了虚拟人的交互性。为了解决这个问题，本章节还实现了视频驱动技术，通过接入视频流，可以估计人物的动画数据，从而驱动虚拟人。虽然视频驱动在一定程度上提高了虚拟人的交互性，但仍需要一个动捕演员来操纵虚拟人的动作，在一定程度上限制了虚拟人的应用范围。此外，无论是动捕驱动还是视频驱动，都需要高质量的摄像头和传感器，这会增加成本和复杂性。

通过语音生成虚拟人的动作不但可以避免前两种驱动方式出现的问题，还可以提高交互的自然度和可信度，使得虚拟人更加真实、生动，从而增强用户的体验和参与度。在提高沟通效率方面，语音生成虚拟人动作可以将语音转化为可视化的行动，帮助用户更加准确地理解虚拟人的反应和回应。在增强用户体验方面，它可以增加交互的趣味性和真实感，从而使用户更加沉浸在交互过程中。在提高交互的自然度方面，它可以使得虚拟人的反应更加自然、真实，从而使得交互更加自然流畅，减少了用户的认知负担和交互的阻碍。此外，语音生成虚拟人动作可以适用于多种场景，例如教育、娱乐、客服等领域，可以提供更加直观、生动的解决方案。在之后的章节，本文将研究如何生成与语音同时发生的手势动作 (Co-speech Gesture)，即人在说话时的肢体动作。这些手势通常与语音内容相关联，可以表达说话者的情感、意图、强调等信息，其与语音一起传达了更加丰富的信息。通过手势，说话者可以强调特定的词汇或句子，也可以表达肯定、否定、疑问等语气，从而使得交流更加直观和生动。

第 4 章 基于语音素材的骨骼动作数据生成

4.1 方法

4.1.1 总体结构概述

现有方法生成的动作序列存在动作静态、动作抖动、不连续、与语音的相关性不高等问题。本文针对这些问题分别提出了解决方法，并组合这些方法设计了语音生成手势的网络模型。针对动作静态的问题，本文使用了动作质量判别器和构建动作风格空间等方法。通过令动作质量判别器和动作生成器进行对抗训练，使得生成的动作序列越来越真实。本文通过在模型的输入中引入说话人信息来指定动作风格，从而使得生成动作序列具有一定的多样性，相同的音频和文本能够产生不同的动作序列。针对动作抖动、不连续等问题，本文采用了 6D 的旋转表示作为动作的表示形式，以消除由于欧拉角等不连续表示带来的动作剧变等问题。此外，本文还采用了 Slerp 插值对生成的序列进行后处理，以消除生成序列的抖动问题。针对语音相关性不高等问题，本文采用了孪生网络结构设计了语义判别器，用于提取音频特征和动作特征，通过计算两个特征的距离得到两者的相关性。此外，在跨模态方法的研究方面，如何融合不同模态数据的特征也是一项具有挑战性的任务。因此，为了探索多模态数据的融合方式对生成序列的影响，本文设计了两种不同的编码器结构，即时间融合编码器和特征融合编码器，分别令两个模态的特征在时间维度融合和特征维度融合。

本文的研究目标是设计一个网络模型，能够生成与语音驱动的动作序列，以丰富虚拟人物的表达能力。为了使虚拟人的表达更接近真实人类，生成的动作序列需要达到一定的质量、多样性和语义相关性要求。在动作质量方面，生成的动作序列应该是真实自然的。在动作多样性方面，同一个音频在不同场景下应该生成不同的动作序列，即不同说话人对应的动作序列应该不同。在语义相关性方面，动作序列的节奏和含义应该与音频一致。

网络的定义如式 4.1 所示，输入音频、文本和说话人信息，生成其对应的动作序列。音频是一段语音信号，文本可由语音信号转录得来，说话人信息用说话人 ID 表示。如图 4.1 所示，网络总体架构采用编码器-解码器结构。编码器负责

提供音频特征和文本特征的联合表示，解码器将联合表示、前一步产生的动作序列、说话人信息作为输入，基于此解码生成新的动作序列。

$$motion = model(audio, text, speaker) \quad \ldots (4.1)$$

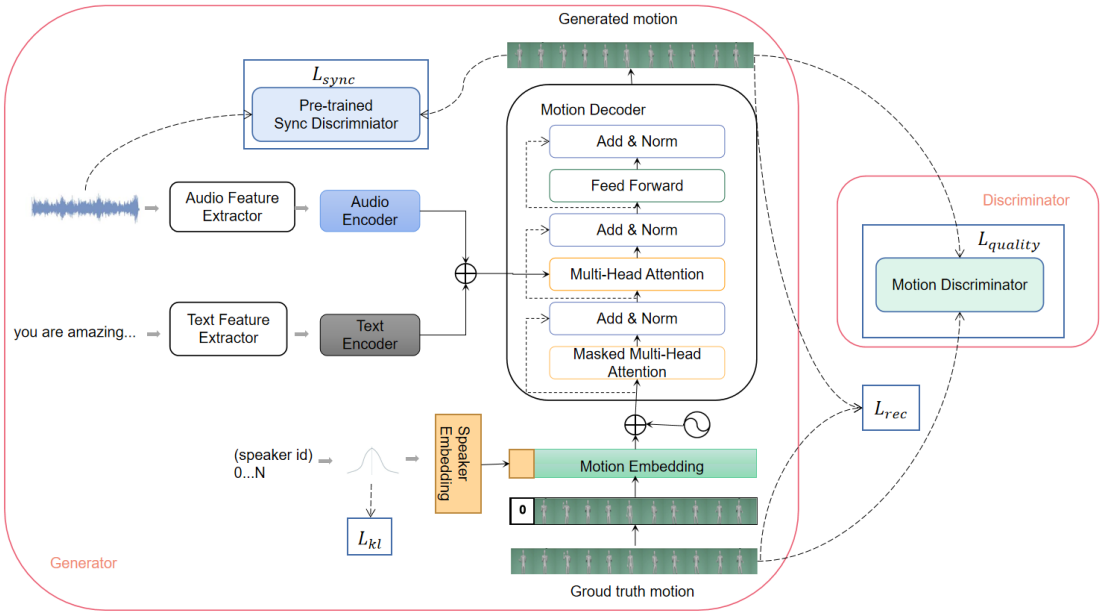


图 4.1　本文提出的网络总体结构

Figure 4.1　The network structure proposed in this paper

对于语音分支，首先通过一个音频特征提取器提取音频的基础特征，如梅尔频率倒谱系数、频谱图、韵律特征等，本文使用预先训练好的 PASE+网络 (Ravanelli 等, 2020) 提取音频特征。一旦得到了音频的基础特征，就可以通过音频编码器将其转化为高维特征。对于文本分支，首先使用一个文本特征提取器提取文本的词向量，然后通过文本编码器将其转化为高维特征。接着，将音频特征和文本特征进行融合，生成一个联合表示，并将其作为解码器的输入。

音频编码器提取出的音频特征维度为（bs, f_a, d_a），文本编码器提取出的文本特征维度为（bs, f_t，d_t），三个维度分别表示序列个数、序列长度和序列的特征维度。SpeechTemplates(Qian 等, 2021) 使得两个模态在序列个数和序列长度上一致，将两个特征在特征维度上叠加，得到两个模态的联合表示，其维度为（bs, $f_a, d_a + d_t$），其中 $f_a = f_t$。Teach(Athanasiou 等, 2022) 使得两个模态在序列个数和序列特征维度一致，将两个特征在序列长度上叠加，得到的联合表示维度为（bs，$f_a + f_t$，d_a），其中 $d_a = d_t$。本文借鉴了这两种思想，分别实现了两套不同

的音频编码器和文本编码器，以尝试两种特征融合方式。关于两个方法的效果对比，将在实验章节描述。

动作解码器使用 TransformerDecoder 生成动作序列。在训练网络时，采用了强制教学策略，因此训练时 TransformerDecoder 需要输入真实的动作序列。Seq2Seq 网络中解码器是从<bos>（开始符号）预测第一帧，然后根据前一帧不断预测下一帧，Transformer 本质上也是一个 Seq2Seq 网络，只是在训练时通过并行生成下一帧来提高处理效率。因此，在输入真实的动作序列前，需要将动作序列向右偏移一位，对于空出的第 0 帧，用<bos>填补。在代码中，将零向量作为<bos>符号。在动作序列经过线性层映射后，用说话人风格作为<bos>符号。

4.1.2 时间融合编码器

时间融合编码器是在时间维度上组合音频和文本特征的编码器结构，参考了文章 Teach(Athanasiou 等, 2022)。编码器结构如下图 4.2 所示，音频编码器和文本编码器均采用了 Transformer 编码器，期望每个序列都能在多头注意力机制下学习到一些高度概括该序列的特征。

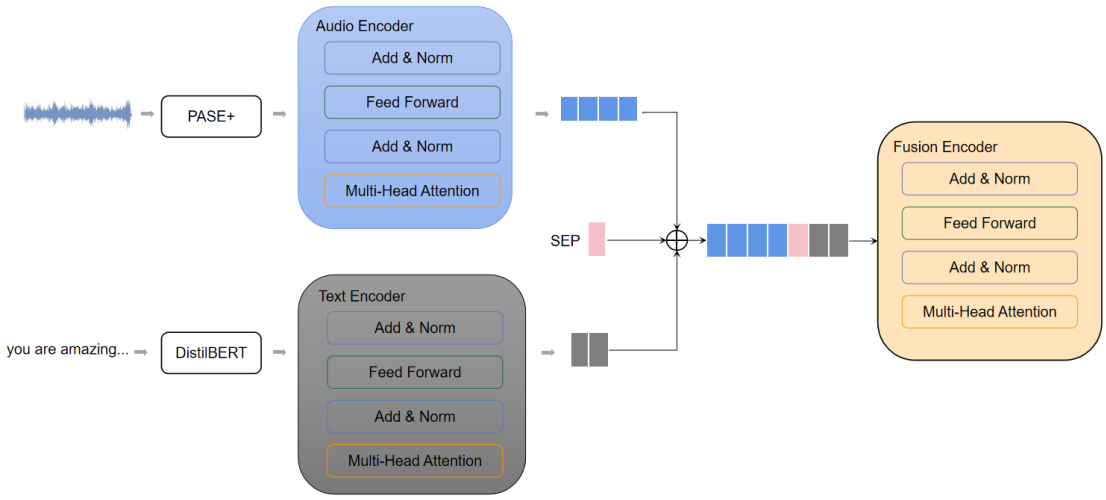


图 4.2 时间融合编码器

Figure 4.2 Time fusion encoder

针对音频的处理，本文使用固定权重的 PASE+网络提取音频的基础特征，然后使用音频编码器学习提取音频的高维特征，其形状为（f_a, bs, d）。如果动作切片为 300 帧，则对应的音频帧数为 1000 帧。在本文中，使用的 PASE+网络提取的音频特征维度为 256 维，在经过音频编码器后得到的特征维度为 512 维。针对文

本的处理，使用 DistilBERT 处理整个句子以获得文本特征，得到的词向量表示维度为（$f_t, bs, 768$）。不同切片对应的单词数量不同，因此每个句子的长度也不同。DistilBERT(Sanh 等, 2019) 返回的文本特征是经过填充后的，同时，DistilBERT 还会返回填充掩码（Padding Mask）。在文本编码器处理这些词向量时，需要将填充掩码传递给编码器，以便编码器了解填充位置，并在处理文本时忽略这些填充处。将音频编码器和文本编码器输出的音频特征和文本特征按时间维度进行组合。为了使网络能够区分这两种模态，借鉴于 Teach(Athanasiou 等, 2022) 方法，在两个模态之间添加了一个分隔符。分隔符也作为网络参数参与学习。经过这一步，最终得到的特征维度为（f_a+1+f_t, bs, d），其中 d 是编码器的维度，在实验中被设置为 512。为了实现两个模态的特征更好的融合，需要将时间维度上组合后的特征传入一个融合编码器进行学习。融合编码器是一个可训练的模型，其输入是由音频编码器、文本编码器输出的音频特征、文本特征在时间维度上组合后的特征，输出为融合后的特征。融合编码器的输出即为整个编码器的输出，也是动作解码器的输入。编码器涉及的实验参数设置如表 4.1 所示。

表 4.1　时间融合编码器的参数设置

参数	维度	说明
d_a	256	PASE+提取的音频特征维度
d_t	768	DistilBERT 提取的文本特征维度
d	512	Transformer 中的特征维度
ff_size	1024	Transformer 中全连接层维度
num_heads	4	Transformer 中多头注意力的头数
num_layers	2	Transformer 中编码器包含的网络层数

4.1.3　特征融合编码器

参考 SpeechTemplates(Qian 等, 2021)，这组编码器实现了将音频特征和文本特征在特征维度上进行融合，旨在通过在时间维度上对齐音频特征、文本特征和动作特征，从而提升网络性能并生成更高质量的动作序列。

网络结构如图 4.3 所示。使用 PASE+提取音频的基础特征，训练时，PASE+网络的权重被固定，不会更新参数。一个 300 帧的动作切片对应音频切片的 PASE

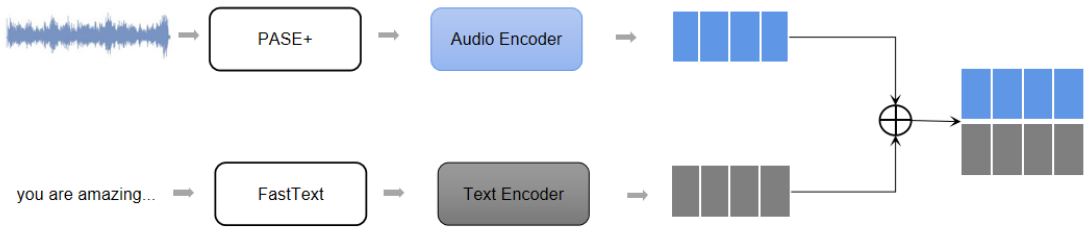


图 4.3　特征融合编码器

Figure 4.3　Feature fusion encoder

特征长度为 1000 帧。为了使音频特征能和文本、动作特征对齐，音频编码器需要对音频下采样到 300 帧。音频编码器采用了 SpeechTemplates(Qian 等, 2021) 中的音频编码器结构，将音频特征视为单通道图像，即音频的 PASE 特征先后经过转置、扩维，使其由 (bs, f_a, 256) 变成了 (bs，1，256，f_a)，然后将转换后的特征依次经过卷积网络，并插值得到指定帧数的音频特征。我们将音频帧数设置成与动作帧数 f_m 一样，即经过编码器后，最终得到音频特征维度为（bs，f_m，256）。对于文本数据，使用 FastText(Bojanowski 等, 2017) 提取单词的词向量，每个单词可以得到一个 300 维的向量表示。为了与动作帧对齐以便与网络训练，将文本特征的长度设置为与动作切片的长度相同，即期望获得一个大小为（f_m，300）的文本切片特征。原始数据提供了每个单词的开始时间和结束时间，将这两个时间分别映射为文本切片中的开始索引和结束索引，在这段时间内的文本特征用单词的 FastText 特征代替，否则用 0 向量代替。假设第 i 个切片的开始时间是 s_t^i，结束时间是 e_t^i，切片长度是 f_m，这段切片中某个单词的开始时间是 s，结束时间是 t，则该单词在切片中对应的开始索引 s_{id} 和结束索引 e_{id} 分别通过式 4.2，式 4.3 计算。本研究采用了 Trimodal(Yoon 等, 2020) 提出的文本编码器。该编码器使用了时序卷积网络（TCN）(Bai 等, 2018) 从词向量中提取 32 维的特征向量，共使用了 4 层时序卷积网络。最终，文本编码器输出的文本特征维度为（bs, f_m，32）。将音频特征，文本特征在特征维度上组合，得到维度为（bs, f_m，288）的联合表示，其作为编码器的输出。编码器涉及的部分实验参数设置如表 4.2 所示。

$$s_{id} = \max(0, \frac{s - s_t^i}{e_t^i - s_t^i} \times f_m) \quad \ldots (4.2)$$

$$e_{id} = \min(n_m^i, \frac{e - s_t^i}{e_t^i - s_t^i} \times f_m) \quad \ldots (4.3)$$

表 4.2　特征融合编码器的参数设置

参数	维度	说明
d_a	256	PASE+提取的音频特征维度
d_t	300	FastText 提取的文本特征维度
d	256	音频编码器中设置的输出音频特征维度
num_frames	300	音频编码器中设置的音频目标采样帧数
text_latent_dim	32	文本编码器设置的输出文本特征维度
num_levels	4	文本编码器中设置的 TCN 层数

4.1.4　动作解码器和说话人风格的建模

由于不同说话人的动作风格具有差异，本文采用了 TriModal(Yoon 等, 2020)提出的网络结构（见图 4.4），通过学习说话人的风格特征空间来建模不同说话人的动作风格。在数据集中共有 17 个说话人，每个说话人在网络中用数字标号标识。首先，通过词嵌入层（Embedding）将说话人 ID 转换成固有的向量表示。接着，通过线性层学习得到该说话人在风格特征空间中的均值和标准差，以模拟该分布。其中，均值表示分布的中心位置，在图 4.4 中用 mean 表示，标准差表示分布的离散程度，在图 4.4 中用 stddev 表示。然后，从学得的分布中采样得到该说话人对应的动作风格。

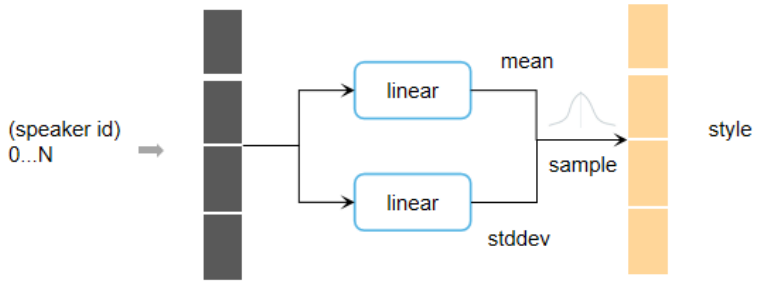


图 4.4　动作风格学习

Figure 4.4　Motion style learning

动作解码器采用了 Transformer 解码器结构，其结构如图 4.5 所示。动作解码器接收编码器输出的上下文向量、动作风格特征和动作序列作为输入。其中，动作序列需要向右偏移一位，对于空出的第 0 帧，用 0 向量填充。动作序列需要

经过线性层映射以得到跟解码器一致的特征维度。映射后，动作序列的第 0 帧用说话人的动作风格向量代替，以实现动作序列和说话人风格的联合表示。联合表示特征经过位置编码后，输入到动作解码器。动作解码器由掩码多头注意力模块、多头注意力模块和前馈网络模块组成。解码器涉及的实验参数设置如表 4.3 所示。

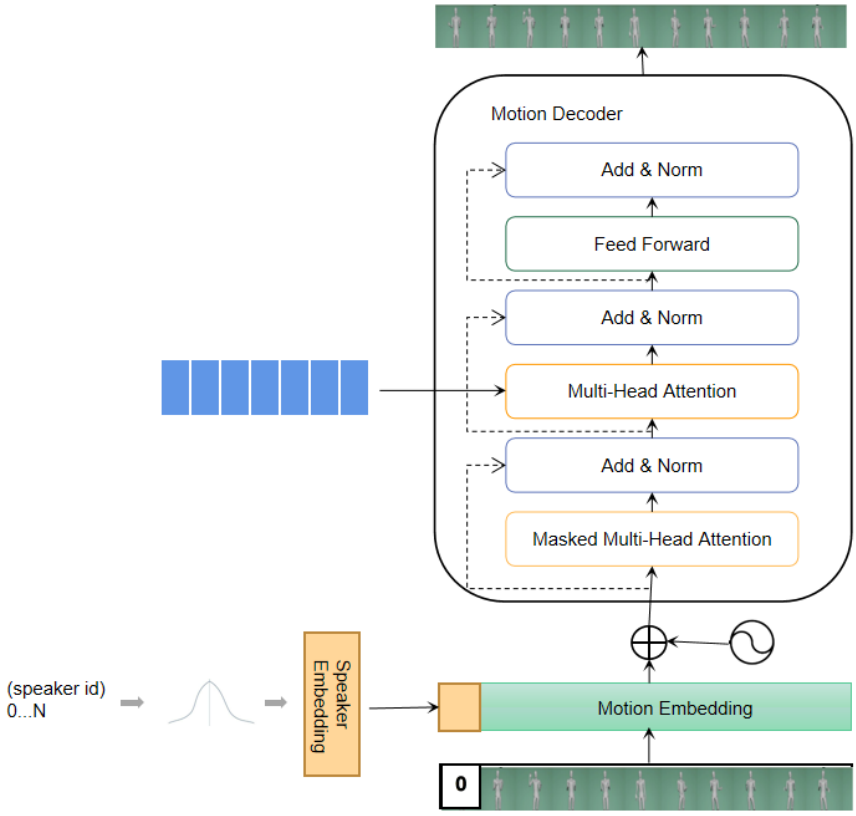


图 4.5　动作解码器结构

Figure 4.5　Motion decoder structure

4.1.5　损失函数设计

网络的损失函数由几个损失项构成（式 4.4），包括重建损失、KL 散度、以及用于监督动作质量的判别损失和用于监督动作语音相关性的感知损失。β，γ，λ 分别表示各项损失的权重系数。

$$L = L_{rec} + \beta \cdot L_{kl} + \gamma \cdot L_{quality} + \lambda \cdot L_{sync} \quad \ldots (4.4)$$

对于真实的动作序列 m 和生成的动作序列 $\hat{m}$，用均方误差（Mean Squared

表 4.3　解码器的参数设置

参数	维度	说明
d_m	333	动作表示维度
d	512	Transformer 中的特征维度
ff_size	1024	Transformer 中全连接层维度
num_heads	4	Transformer 中多头注意力的头数
num_layers	4	Transformer 中解码器包含的网络层数

Error, MSE）作为其重建损失项，其重建损失为式 4.5。

$$L_{rec} = \frac{1}{n}\sum_{i=1}^{n}(m_i - \hat{m}_i)^2 \quad \ldots (4.5)$$

KL 散度（Kullback-Leibler divergence）是用来衡量两个概率分布之间差异的度量方式。通过 KL 散度为每个说话人学得一个风格特征空间，通过均值 μ 和标准差 σ 来描述这个分布（用 P 表示）。将 KL 散度的参考分布设置为标准正太分布（用 Q 表示），其对应的计算公式为式 4.6 和式 4.7。

$$L_{kl} = D_{KL}(P\|Q) = \sum_{i=1}^{n} P(i)\log\frac{P(i)}{Q(i)} \quad \ldots (4.6)$$

$$Q = N(0,1) \quad \ldots (4.7)$$

质量判别器用于对生成的动作序列进行分类，判断其是高质量动作序列还是低质量动作序列，其网络结构设计如图 4.6 所示。质量判别器采用了一个由两个分支构成的网络结构，分别用于提取动作特征和动作速度特征。动作的速度由相邻两帧动作序列的帧差得来。动作特征和动作速度特征都是通过 Transformer 中的多头注意力机制模块和前馈网络模块提取的，最终得到维度为（f_m，bs，d）的特征表示。在提取特征后，通过全局平均池化层（Global Average Pooling, GAP）提取出序列数据的全局信息，缩减时间维度，然后将池化后的全局动作特征和全局速度特征在特征维度上组合，作为 Softmax 层的输入，最终得到动作序列分别属于两个类别的概率。训练质量判别器时使用的损失函数为二元交叉熵损失函数（Binary Cross-Entropy Loss，BCE），将动作质量高的标签设为 1，动作质量低

的标签设为 0，其计算公式为式 4.8，其中 y 表示真实标签，$\hat{y}$ 表示模型将动作序列分类为 1 的概率。在训练动作序列生成器时，期待生成的动作序列都是高质量的，所以将真实标签设置为 1，$L_{quality}$ 的计算公式为式 4.9。动作质量判别器涉及的实验参数设置如表 4.4 所示。

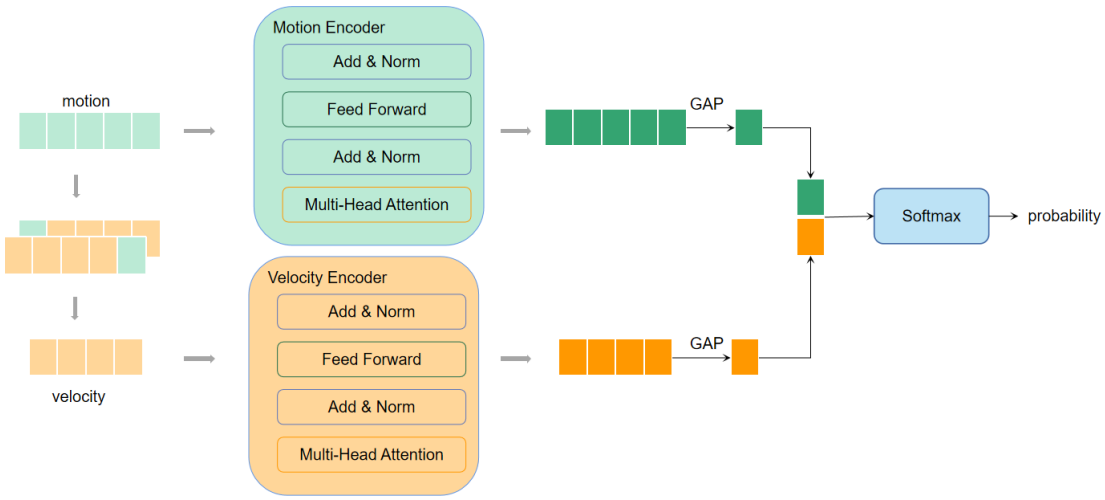


图 4.6　质量判别器结构

Figure 4.6　Quality discriminator structure

$$BCE(y,\hat{y}) = -ylog(\hat{y}) - (1-y)log(1-\hat{y}) \quad \ldots (4.8)$$

$$L_{quality} = BCE(1,\hat{y}) \quad \ldots (4.9)$$

表 4.4　动作质量判别器的参数设置

参数	维度	说明
d_m	333	动作表示维度
dropout	0.1	神经元被丢弃的概率
d	512	Transformer 中的特征维度
ff_size	1024	Transformer 中全连接层维度
num_heads	4	Transformer 中多头注意力的头数
num_layers	4	Transformer 中解码器包含的网络层数

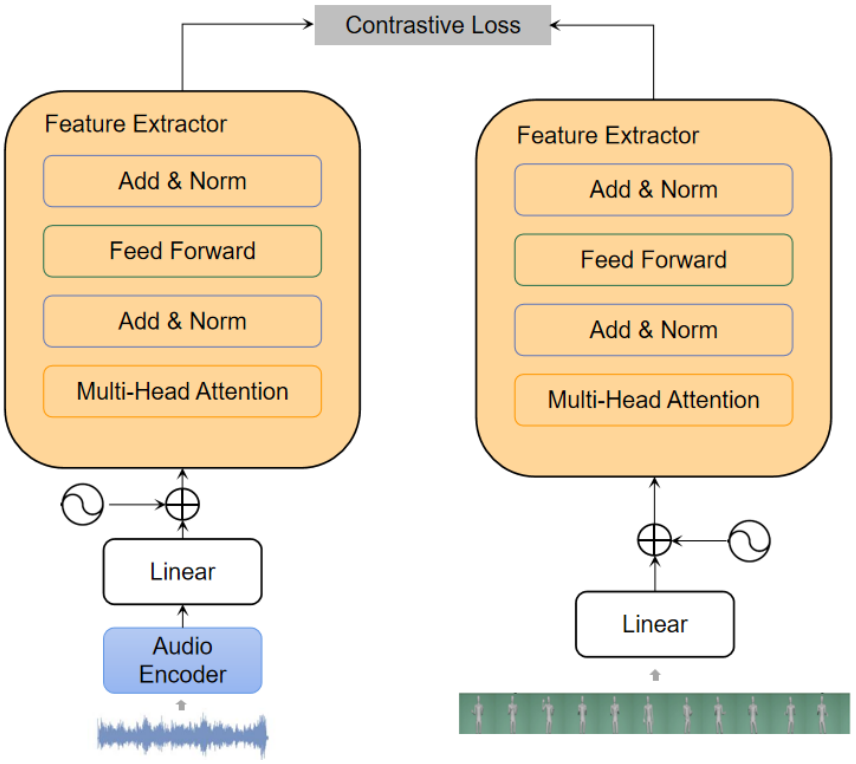


图 4.7　语义判别器结构

Figure 4.7　Semantic discriminator structure

Wav2Lip(Prajwal 等, 2020) 和 SyncNet(Chung 等, 2017) 通过将同步的音频口型错位一定的帧数，以构建不同步的音频口型数据集，并使用孪生网络从数据集中学习。通过该方式，使得同步的音频口型特征距离相近，不同步的音频口型特征距离较远。受启发于这个思想，本文拟通过这种方式学习一个语义判别器，进而提高动作和音频的相关性。GENEA Challenge 2022 的结果表明，用户普遍认为运动捕捉数据在语义相关性方面表现最好，而其他生成方法在相关性方面与运动捕捉数据存在显著差距。因此，在本文中，将动捕数据视为语义相关的数据。通过将原数据与音频错位一定的帧数，构造语义不相关的动作数据。具体而言，将动作数据整体向右偏移 10 秒，相当于将音频与动作错位了 300 帧。语义判别器设计的网络结构如图 4.7 所示。网络总体上采用孪生网络结构，使用了相同的特征提取器提取音频特征、动作特征。特征提取器采用了 Transformer 的多头注意力模块和前馈网络模块，并使用了 Speech2Templates(Qian 等, 2021) 的音频编码器结构将音频特征下采样到与动作序列帧数相同的长度。损失函数采用了对比损失（Contrastive Loss），它的计算公式如式 4.10，其中 y 表示相似性标

签，d 表示样本对之间的距离，margin 表示控制距离的参数，用于控制相似和不相似样本之间的距离。用 y=1 表示样本对相似，y=0 表示样本对不相似，d 可以选择使用欧式距离或者余弦距离，来表示音频特征和动作特征间的距离。模型通过使用对比损失，最终可以使得相似的数据点在特征空间中更加接近，不相似的数据点在特征空间中更加远离。在训练动作生成器时，需要固定该语义判别器的权重，用其分别提取音频和生成动作序列的特征。L_{sync} 也采用对比损失（式 4.11），期待生成的动作序列与音频是相关的，所以令 y=1，表示音频和动作是相似的，从而通过反向传播使得音频和动作序列在特征空间中越来越接近。

$$L_{contrastive}(y,d) = yd^2 + (1-y)max(margin-d,0)^2 \quad \ldots (4.10)$$

$$L_{sync} = L_{contrastive}(1,d) \quad \ldots (4.11)$$

4.2 训练

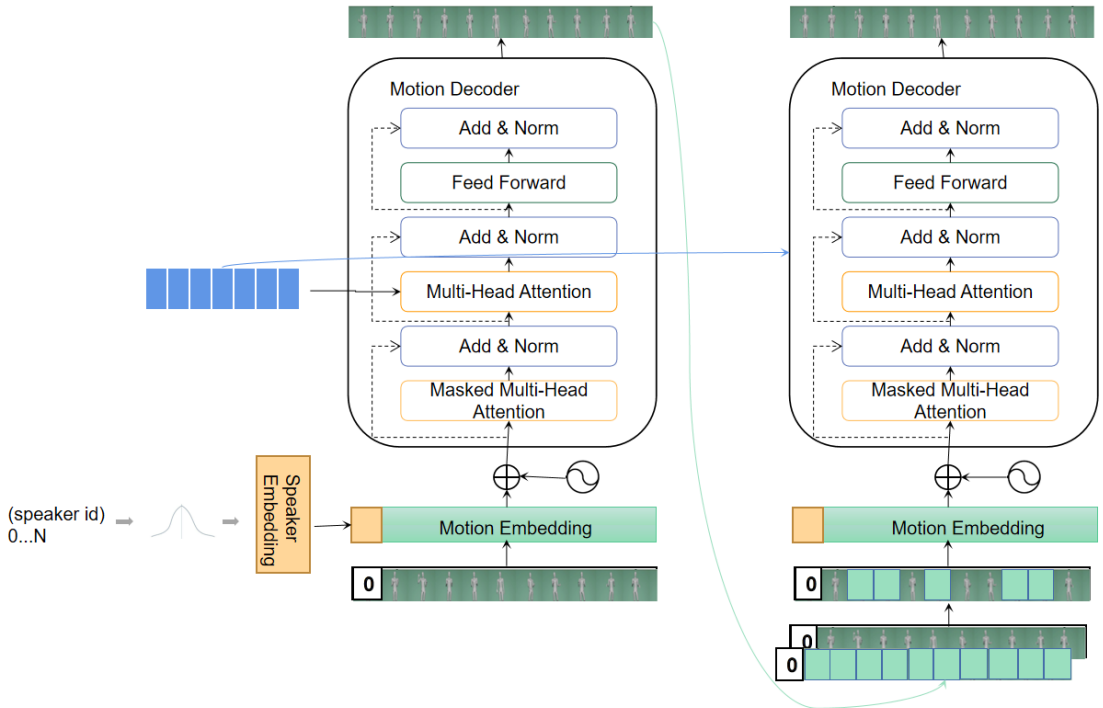


图 4.8　适用于 Transformer 的计划采样

Figure 4.8　Scheduled sampling for Transformer

在本研究中，为了加速模型训练过程，使用了强制教学策略，即将真实的动

作序列向右偏移一帧后，提供给了模型。然而，完全使用强制教学训练模型可能会导致模型出现误差累计问题，因为模型的每个预测都依赖于先前的真实标签数据，而不是先前的预测结果。为了缓解这个问题，本研究采用了 Mihaylova 等 (2019) 为 Transformer 设计的计划采样思想。如图 4.8 所示。具体而言，通过两次使用解码器，以一定概率向解码器传递了模型的预测输出，而不只是使用真实序列值作为前一帧，这样可以在一定程度上缓解了强制教学策略导致的误差累计问题。

详细步骤如下。首先使用完全的强制教学策略，将真实的动作序列作为解码器的输入，生成模型的预测序列。接着，以一定概率混合真实的动作序列和模型的预测序列，生成混合序列。混合序列再次传入解码器，生成最终的预测输出。在实现中，使用的概率阈值为 0.5，对于指定帧，其随机生成的概率若超过阈值，则使用真实动作帧作为解码器的输入，否则使用前一个编码器预测的动作帧作为输入。

为了提高生成动作序列的真实度，采用生成对抗网络（GAN）的训练方法。将动作解码器作为生成器，将动作质量判别器作为判别器，其中生成器的损失函数如 4.1.5 章节描述的相同，而判别器采用二元交叉熵损失函数。通过生成器和判别器以对抗学习的方式相互优化，以让生成器生成的动作序列尽可能地更接近真实的动作序列。同时，之前提到的语义判别器也参与到生成器的优化中，使用感知损失来提高生成动作序列和语音之间的相似度。

4.3　推理

在训练过程中，由于已知真实序列，对于每个位置的当前帧都知道其前面的动作序列，因此可以一次性得到音频对应的所有动作序列帧。但是在模型预测时，之前动作帧只能由模型预测得来，因此预测时只能逐帧生成动作序列。

训练时为了使得根节点的位置值范围跟旋转的表示范围一致，对其做了归一化。在模型预测生成得到动作序列后，需要将根节点位置值映射到其原来的值域范围内。采用式 4.12 还原根节点的位置，eps 选用 1e-8，其中 mean_pose 表示训练数据中动作的均值，max_pose 表示训练数据中动作的最大值。

$$pose = pose * (max\_pose - mean\_pose + eps) + mean\_pose \quad \ldots (4.12)$$

生成动作序列的旋转表示采用的是 pose6d，需要将其转换成欧拉角。欧拉角通常由旋转矩阵转换得来，故需要先将 pose6d 转换成旋转矩阵。对于一组 pose6d 的旋转表示，其可以拆解成两个 3 维的向量 $\boldsymbol{a_1}, \boldsymbol{a_2}$，依次对其采用施密特正交化（Schmidt Orthogonalization），叉乘得到 3 个标准的正交向量作为旋转矩阵 R 的方向向量 $\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}$。计算过程如式 4.13，式 4.14，式 4.15，式 4.16，式 4.17。获取到旋转矩阵后，由于给定的 BVH 文件中旋转顺序是 ZXY, 故需要将其按照原来的旋转顺序转换成欧拉角（γ, α, β）。由于旋转矩阵的每个值已经计算得到，可以根据反正切（arctan），反正弦（arcsin）函数分别得到 γ, α, β。在 BVH 文件中，欧拉角的单位是角度，故需要将弧度转化角度。计算过程如式 4.18、式 4.19，其中 $\boldsymbol{R_x}(\alpha)$、$\boldsymbol{R_y}(\beta)$、$\boldsymbol{R_z}(\gamma)$ 的计算分别见式 5.2、式 5.3、式 5.4。

$$\boldsymbol{b_1} = \frac{\boldsymbol{a_1}}{\|\boldsymbol{a_1}\|} \quad \ldots (4.13)$$

$$\boldsymbol{b_2} = \boldsymbol{a_2} - \frac{< \boldsymbol{a_2}, \boldsymbol{b_1} >}{< \boldsymbol{b_1}, \boldsymbol{b_1} >} \boldsymbol{b1} \quad \ldots (4.14)$$

$$\boldsymbol{b_2} = \frac{\boldsymbol{b_2}}{\|\boldsymbol{b_2}\|} \quad \ldots (4.15)$$

$$\boldsymbol{b_3} = \boldsymbol{b_1} \times \boldsymbol{b_2} \quad \ldots (4.16)$$

$$R = [\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}] \quad \ldots (4.17)$$

$$R = \boldsymbol{R_z}(\gamma) * \boldsymbol{R_x}(\alpha) * \boldsymbol{R_y}(\beta) = [\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}] \quad \ldots (4.18)$$

$$R = \begin{bmatrix} cos\gamma cos\alpha - sin\gamma sin\beta sin\alpha & -cos\beta sin\gamma & cos\gamma sin\alpha + cos\alpha sin\gamma sin\beta \\ cos\alpha sin\gamma + cos\gamma sin\beta sin\alpha & cos\gamma cos\beta & sin\gamma sin\alpha - cos\gamma cos\alpha sin\beta \\ -cos\beta sin\alpha & sin\beta & cos\beta cos\alpha \end{bmatrix} \quad \ldots (4.19)$$

直接由模型生成的动作数据较为抖动，本文使用了插值解决动作序列的抖动问题。动作的全局位移选择了线性插值。插值公式如式 4.20，其中 p, q 分别表

示相邻两帧中根节点的位置，t 表示插值系数，范围为 0 到 1 之间。动作的旋转采用了球面线性插值（Slerp）。Slerp 是一种在四元数空间中执行插值的方法，它通常用于在计算机图形学中执行动画的旋转和方向插值。Slerp 插值通过在四元数表示的旋转空间中沿着球面路径进行线性插值来计算两个四元数之间的中间值。这个球面路径是一个单位球面上的弧，它连接了两个四元数所代表的旋转。Slerp 插值保证了在这个球面路径上的平滑过渡，这样可以避免出现旋转翻转或抖动的问题。Slerp 插值公式如式 4.21，其中 p, q 是两个四元数，t 是插值系数，它通常在 0 到 1 之间，θ 是 p，q 间的夹角。在代码实现中，插值的步骤主要分为以下几个步骤。首先，使用四元数来表示旋转信息，然后对四元数进行归一化处理，并使用 Slerp 插值方法得到新的四元数。接着，将插值得到的四元数归一化，并将其转换为旋转矩阵，最后将旋转矩阵转换为欧拉角。为了保证插值后的动作帧数与原始动作序列帧数相同，即插值后的动作序列仍能对应到指定时间的音频片段，模型在推理时每隔 5 帧生成一个动作关键帧，并在相邻的动作关键帧中插值 4 帧。由于动作旋转在插值前为欧拉角，其需要先转化成四元数进行插值。对于 ZXY 顺序的欧拉角（γ, α, β），其对应的四元数计算方式如式 4.22，式 4.23，式 4.24，式 4.25。对于四元数（w, x, y, z），其对应的旋转矩阵计算公式如式 2.4。

$$Linear(p, q, t) = (1 - t) \cdot p + t \cdot q \quad \ldots (4.20)$$

$$Slerp(p, q, t) = \frac{sin[(1 - t)\theta] \cdot p + sin(t\theta) \cdot q}{sin\theta} \quad \ldots (4.21)$$

$$q = q_\gamma * q_\alpha * q_\beta \quad \ldots (4.22)$$

$$q_\gamma = [cos\frac{\gamma}{2}, 0, 0, sin\frac{\gamma}{2}] \quad \ldots (4.23)$$

$$q_\alpha = [cos\frac{\alpha}{2}, sin\frac{\alpha}{2}, 0, 0] \quad \ldots (4.24)$$

$$q_\beta = [cos\frac{\beta}{2}, 0, sin\frac{\beta}{2}, 0] \quad \ldots (4.25)$$

最终生成的动作数据用 BVH 文件表示。其在 Blender 软件中可视化结果如图 4.9 所示。为了更好的评估动作和语音的相关性，使用了 GENEA Challenge 2022 提供的渲染代码将音频和动作数据合成到了一个视频文件，如图 4.10 所示。

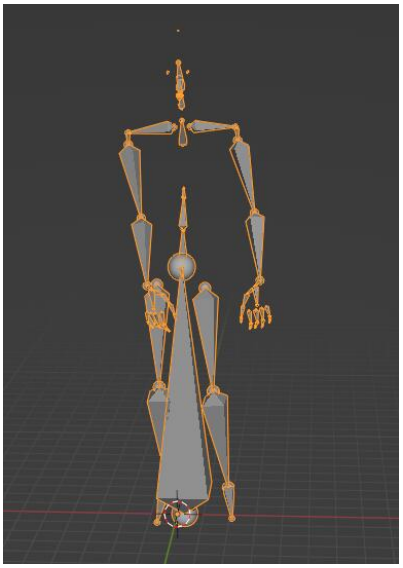


图 4.9　Blender 中可视化 BVH 文件

Figure 4.9　Visualization of BVH file in Blender

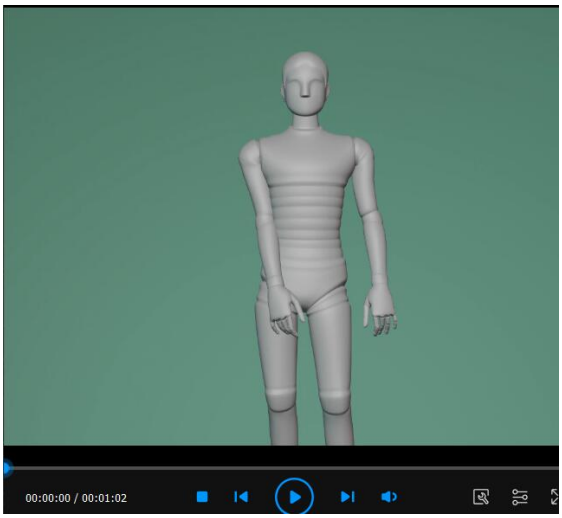


图 4.10　渲染后的骨骼动画

Figure 4.10　Rendered skeletal animation

4.4　本章小结

本文针对生成动作序列存在的静态、抖动、不连续、与语音相关性不高等问题，提出了相应的解决方法，并设计了语音生成手势的网络模型。该模型采用了编码器-解码器结构，其中编码器由时间融合编码器和特征融合编码器两种结构组成，分别用于提取音频特征和文本特征，并通过两种不同的融合方式将两种特征进行融合。解码器则用于基于前一步产生的动作序列、联合特征和说话人信息生成新的动作序列。此外，为了解决动作静态、抖动和不连续等问题，本文还使用了动作质量判别器和构建动作风格空间、6D 的旋转表示和 Slerp 插值等方法。而为了提高语音和动作序列的相关性，本文还采用了孪生网络结构设计了语义判别器。最后，为了建模不同说话人的动作风格，本文采用了 TriModal(Yoon 等, 2020) 提出的网络结构。

第 5 章 实验

5.1 数据处理

本文所使用的数据集为 GENEA Challenge 2022 所使用的 Talking With Hands 16.2M 数据集 (Lee 等, 2019)。该数据集被分为三个部分：训练数据集 (293 条数据)、验证数据集 (40 条数据) 和测试数据集 (40 条数据)。训练数据集和验证数据集均提供了语音、文本、说话人和动作数据，而测试数据集只提供了语音，文本，说话人。然而，GENEA Challenge 2022 提供了所有参赛方法和动捕系统针对测试数据集生成的动作序列，这可以用于比较本文所提出的方法生成的动作序列。每个数据集文件夹下均提供了一个 CSV 文件，用来存储该数据集的元数据。CSV 文件的每一行表示该数据集下的一个数据信息，每一行有三列。第一列表示文件名，其中音频、文本和动作的文件名相同，但音频数据以 WAV 文件格式保存，文本数据以 TSV 文件格式保存，动作数据以 BVH 文件格式保存。第二列显示该条数据是否包含手指运动。第三列显示该数据对应的说话人，共有 17 个说话人。语音数据涉及的语言是英文，音频采样率是 16,000。文本数据是通过使用谷歌云的语音识别工具转录其对应音频而获得的，TSV 文件中包含单词及其起始时间信息。动作数据是通过录制两个动捕演员自由交谈各种话题的动作获得的, 其帧率为 30fps。GENEA Challenge 2022 还重新定义了 T-pose 的骨骼结构、重定位了各个动作，并标准化了动作的位置和朝向，这是因为录制时两个说话者站在两个不同的位置上，且面对面。GENEA Challenge 2022 提供了一些处理数据集的代码，包括读写 BVH 文件、提取音频的基本特征（如 MFCCs、频谱图、韵律特征等），本文采用了部分代码。由于读 BVH 文件、提取音频特征等操作比较耗时，本文提前读取了这些数据，并将其保存在 pkl 文件中以便后续使用。

在加载音频时，需要进行单声道和双声道的判断。如果音频是双声道，需要对左右声道求平均值，得到单声道的 numpy 数组，保存了音频的采样点。通过 PASE 进行音频特征提取，得到 $(f_a, 256)$ 的向量表示。其中，f_a 表示音频的帧数, 256 表示每帧音频的特征维度。在读取文本文件时，获取每个单词的开始时间和结束时间。读取动作文件时，得到维度为 $(f_m, 56, 6)$ 的动作序列值。其中，f_m 表示动作序列的帧数，56 表示关节点个数，6 表示关节点的位置和旋转角度，角度

的表示单位为度。由于 BVH 文件的特性，除根节点外，其他关节点在不同帧的的位置数据值是相同的，根节点的位置决定了模型的位置。因此，为了去除这些冗余信息，选择使用根节点的位置和其他关节点的旋转值作为动作表示，得到了 (f_m, 56, 3) 的向量表示。再将动作数据中的欧拉角转换成旋转的 6D 表示（下文简称为 pose6d），最终得到 (f_m, 333) 的动作表示。根据 CSV 元数据文件读取说话人标识，将读取的这些数据通过字典形式保存成 pkl 文件。

将欧拉角转换成 pose6d 的具体步骤如下。先计算欧拉角对应的旋转矩阵。对于给定的欧拉角（α, β, γ），α 表示绕 X 轴旋转的角度，对应旋转矩阵计算方式为式 5.2，β 表示绕 Y 轴旋转的角度，对应旋转矩阵计算方式为式 5.3，γ 表示绕 Z 轴旋转的角度，对应旋转矩阵计算方式为式 5.4。欧拉角（α, β, γ）对应的旋转矩阵为 R，则 R 的计算方法如式 5.1，$\boldsymbol{a_1}$，$\boldsymbol{a_2}$，$\boldsymbol{a_3}$ 表示维度为 3×1 的列向量。由于 $\boldsymbol{a_1}$，$\boldsymbol{a_2}$，$\boldsymbol{a_3}$ 相互正交，任何一个向量可由其余两个向量计算得到，导致旋转矩阵存在冗余信息。Pose6d 的计算方式如式 5.5 所示，只保留旋转矩阵的前两列值。

$$R = R_z(\gamma) * R_y(\beta) * R_x(\alpha) = [\boldsymbol{a_1}, \boldsymbol{a_2}, \boldsymbol{a_3}] \quad \ldots (5.1)$$

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\alpha & -sin\alpha \\ 0 & sin\alpha & cos\alpha \end{bmatrix} \quad \ldots (5.2)$$

$$R_y(\beta) = \begin{bmatrix} cos\beta & 0 & sin\beta \\ 0 & 1 & 0 \\ -sin\beta & 0 & cos\beta \end{bmatrix} \quad \ldots (5.3)$$

$$R_z(\gamma) = \begin{bmatrix} cos\gamma & -sin\gamma & 0 \\ sin\gamma & cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \ldots (5.4)$$

$$pose6d = [\boldsymbol{a_1}, \boldsymbol{a_2}] \quad \ldots (5.5)$$

在网络加载数据集时，首先要加载先前保存的数据，并进行切片处理。本文尝试了两种切片方式。第一种是基于语句的切片方式，根据每句话的文本信息来

切割相应的音频和动作数据。第二种是采用滑动窗口的方式，对数据进行切片。两种切片方式都能够将数据集切割成小的数据块，便于模型进行处理。

首先尝试了根据文本对数据进行切片的方法。在原始数据中，每个单词的开始时间和结束时间都会被提供。借鉴 DSI(Saleh, 2022) 中的句子构建方法，我们启发式地将单词不断加入一个句子中，直到当前单词的开始时间距离前一个单词的结束时间大于 0.5s。这样可以认为上个句子在前一个单词结束后结束，然后开始构建一个新的句子。这样，我们就可以得到每个句子的开始时间和结束时间，并根据这两个时间来切割音频和动作数据。但是，这种方法存在一个问题：文本并不是每个时间段都有单词，导致部分音频和动作数据没有对应的文本。由于切片方法的限制，这部分数据会被忽略掉。在推理模型时，根据文本时间对音频进行切片，将音频和文本数据作为模型的输入，然后生成动作序列。然而，没有文本对应的音频则不能考虑到动作生成。

由于通过文本切片具有上述限制，本文选择了使用滑动窗口的方式对数据进行切片。在该方法中，先根据动作帧数和动作帧率计算出动作的时间范围。假设动作帧数为 f_m，动作帧率为 fps，则动作时间 t_m 的计算公式如式 5.6。根据音频采样点个数和音频采样率获取音频时间。假设音频采样点个数为 n_a，音频采样率为 sr，则音频时间 t_a 的计算公式如式 5.7。动作时间和音频时间不一致，根据时间较短的数据决定时间范围，最终的时间计算公式为式 5.8。设置的窗口大小为 10s，滑动的步数为 5s，从时间 0 开始，按照 10s 的窗口切割数据，分别获取该切片对应的原始音频、音频的 PASE 特征、文本，说话人表示和动作数据。对于每个切片，可以根据单词的开始时间和结束时间确定其对应的文本。此外，还需要获取所有训练数据中动作的均值 (mean_pose) 和最大值 (max_pose) 并保存，用来对动作数据前 3 维的位置做归一化。未归一化时，动作数据有 3 维的位置和 330 维的 pose6d 组成。pose6d 值的范围为 (-1,1), 而位置值的范围远大于这个范围。若使用未归一化的数据进行训练，最终会导致生成的动作序列中，根节点不动，一直在一个固定位置。归一化的公式如式 5.9，在实际实现中，eps 选用 1e-8。当切片切到数据末端时，可能无法凑满 10s 的窗口长度。所以同时处理多条数据时，需要对较短的数据进行填充 0 的操作，以保证每条数据的长度相同。在填充后，还需要生成一个填充掩码（Padding Mask），以指示模型在哪些位置

有填充数据，并忽略这些填充的 0 值。

$$t_m = \frac{f_m}{fps} \quad \ldots (5.6)$$

$$t_a = \frac{n_a}{sr} \quad \ldots (5.7)$$

$$t = \min(t_m, t_a) \quad \ldots (5.8)$$

$$pose = \frac{pose - mean\_pose}{max\_pose - mean\_pose + eps} \quad \ldots (5.9)$$

5.2　语义判别器，质量判别器验证

本实验旨在验证本文提出的语义判别器和质量判别器的有效性。

对于语义判别器，本实验使用训练数据和验证数据分别构造匹配的语音动作对和不匹配的动作序列对。训练时采用的切片时长是 10s，即动作序列有 300 帧，音频的 PASE 特征有 1000 帧。在训练过程中，采用对比损失（Contrastive Loss）作为损失函数，选择欧式距离作为特征间的距离函数，并对模型进行了 100 次迭代训练（epoch），学习率设置为 1e-4，对比损失中的 margin 设置为 0.4。

GENEA Challenge 2022 公开了各个参赛方法根据测试语音生成的动作序列及其与语音相关性（Appropriateness）的评分结果。本实验拟通过训练得到的语义判别器计算这些动作序列和其对应音频的距离，距离越小则说明音频和动作在特征空间中越接近，即音频和动作越相关，否则则说明音频和动作不相关。每个方法分别有 40 条动作序列，对这些动作序列的评估结果取均值。由于训练数据采用的是切片长度为 300 帧的动作序列和 1000 帧的音频特征，为了确保测试数据与训练数据一致，本实验将每个测试动作和音频按照 10s 的长度切片。这些切片作为语义判别器的输入，用于提取音频特征和动作特征。通过计算两种模态特征间的距离判断它们的相似性，所有切片的平均距离作为该动作序列和音频序列之间的距离。

表 5.1 展示了各个方法产生的动作序列和音频的特征距离。其中，m 表示方法（FBT(Yoon 等, 2019)，FNA(Lee 等, 2019)，FSA(Zhou 等, 2022)，FSB，FSC(Ghorbani 等, 2022)，FSD(Korzun 等, 2022)，FSF(Saleh, 2022)，FSG(Windle 等, 2022)，

FSH(Chang 等, 2022)，FSI(Lu 等, 2022)），d 表示动作和音频的特征距离。FBT(Yoon 等, 2019) 表示只根据文本生成的动作序列，FNA(Lee 等, 2019) 表示由动作捕捉系统录制的动作序列。

表 5.1　动作和音频的特征距离

m	FBT	FNA	FSA	FSB	FSC	FSD	FSF	FSG	FSH	FSI
d	0.0261	0.0270	0.0406	0.0271	0.0318	0.0263	0.0263	0.0317	**0.0257**	0.0258

用折线图可视化上表数据如下图 5.1 所示，其中横轴表示动作序列的生成方法，方法的排列顺序与 GENEA Challenge 2022 中动作的语义相关性评分一致，从左至右表示从语义相关性高到语义相关性低。纵轴表示动作序列与对应音频的距离。图示结果表明该语义判别器能够在一定程度上评估音频和动作序列的相关性，但是易受到动作质量的干扰。例如，FSA(Zhou 等, 2022) 和 FSC(Ghorbani 等, 2022) 是 GENEA Challenge 2022 中动作质量评分较高的方法，但在图中这两者得到距离都比较大，这表明判别器容易将丰富多变的动作序列与音频之间的距离计算为一个较大的值，从而干扰了判别器对其语义相关性的评估。

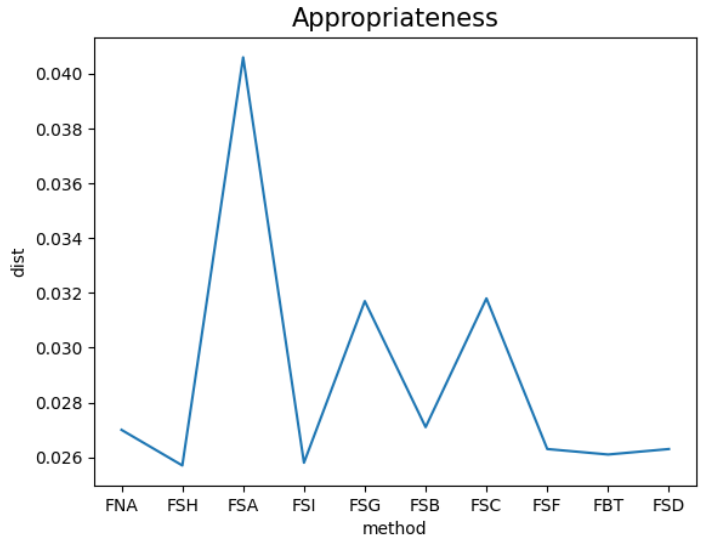


图 5.1　通过语义判别器计算得到的各个方法的语义相关性评分

Figure 5.1　Semantic relevance scores of comparing methods. The scores are calculated by the semantic discriminator

表 5.2 展示了同一段音频和不同的动作序列的距离。其中 w00 表示音频名

称，m00 至 m09 表示动作名称。真实标签是音频 w00 和动作数据 m00 来源于同一个数据对，两者相关，音频和动作数据 m01-m09 均不相关。折线图 5.2 展示了各个数据对的距离趋势，其中只有 5 个不相关的数据对距离超过了 w00 和 m00 的距离，说明该判别器虽然能识别部分不相关的音频数据对，但是还存在一定的限制。

表 5.2　不同动作和音频的特征距离

m	m00	m01	m02	m03	m04	m05	m06	m07	m08	m09
w00	0.0265	0.0271	**0.0250**	0.0253	0.0287	0.0270	0.0267	0.0262	0.0264	0.0270

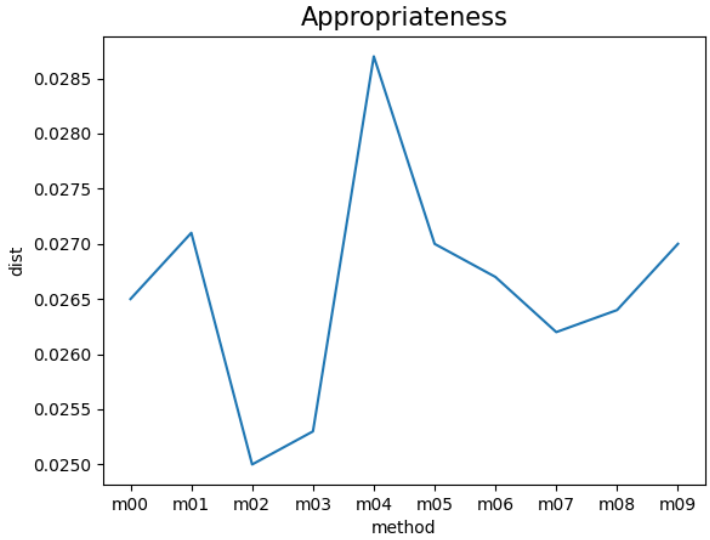


图 5.2　同一音频不同动作序列的语义相关性评分

Figure 5.2　Semantic relevance scores for different action sequences of the same audio

对于质量判别器，其通过训练对抗生成网络的方式得到，与动作生成器交替训练。GENEA 2022 公开了各个参赛方法根据测试语音生成的动作序列及其动作质量（Human-likeness）的评分结果。本实验拟通过训练得到的质量判别器计算这些动作序列的分类结果，其中动作质量高的序列分类为 1，动作质量低的序列分类为 0。每个方法分别有 40 条动作序列，统计每个方法中被分类为 1 的动作序列数量。由于训练时采用了切片数据，为了使得测试数据和训练数据一致，测试时也将动作数据按照 10s 的窗口长度进行切片，对切片后的动作序列进行分类。若有超过一半的切片序列被分类为高质量序列，则该动作序列视为高质量，

否则将其视为低质量。表 5.3 展示了分类结果。

表 5.3　质量判别器对各方法对应的动作序列的分类结果

m	FBT	FNA	FSA	FSB	FSC	FSD	FSF	FSG	FSH	FSI
n	**40**	33	37	**40**	**40**	36	14	30	**40**	**40**

用折线图可视化表 5.3 的结果如图 5.3 所示，横轴按照 GENEA Challenge 2022 中质量评分从高到底的排名排列，表示各个参赛方法。根据折线图趋势，可以反映出该判别器还未能很好地分类动作序列质量。

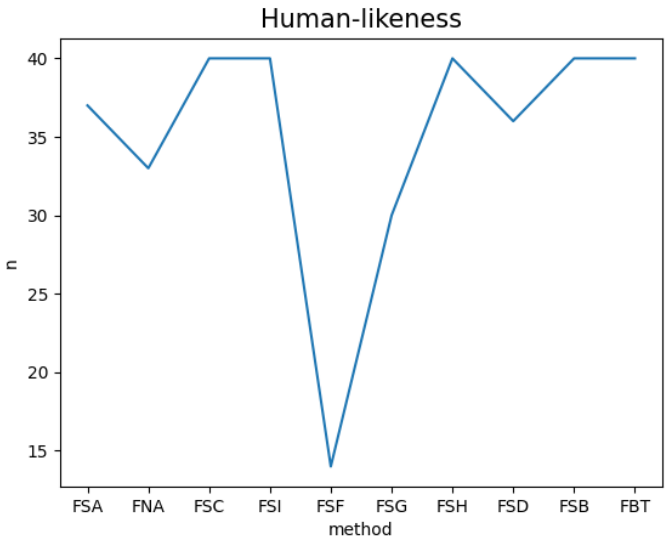


图 5.3　质量判别器对各个方法产生的动作质量评分

Figure 5.3　Action quality discriminator score for each method

5.3　时间融合编码器，特征融合编码器的效果比较

本文设计了两种编码器结构。时间融合编码器使用 Transformer 中的多头注意力模块和前馈网络模块分别处理音频和文本，在时间维度上组合音频特征和文本特征。特征融合编码器结合了 SpeechTemplates(Qian 等, 2021) 中的音频编码器和 Trimodal(Yoon 等, 2020) 中的文本编码器，尝试将音频文本和动作序列在时间维度上对齐，在特征维度上组合音频特征和文本特征。本实验用于探究两种编码器的效果。

每个方法均训练迭代了 100 次，选择损失最小时对应的网络参数进行测试。学习率为 1e-4，采用 Adam 作为生成器和判别器的优化器。实验设置如下，从测试数据集中选取一条语音，分别采用两个编码器对应的方法生成动作序列，比较两个动作序列的质量和语义相关性。

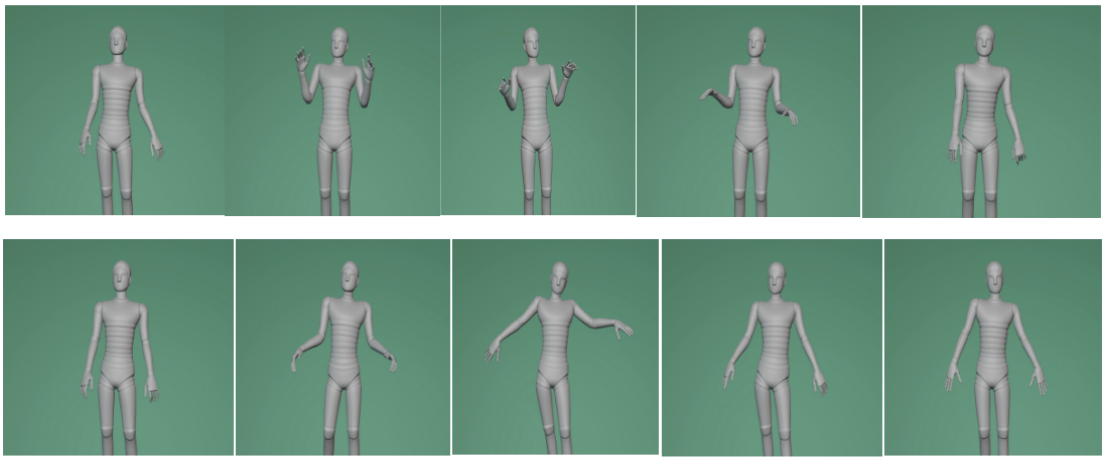


图 5.4　时间融合编码器产生的动作序列

Figure 5.4　Action sequence generated by Time Fusion Encoder

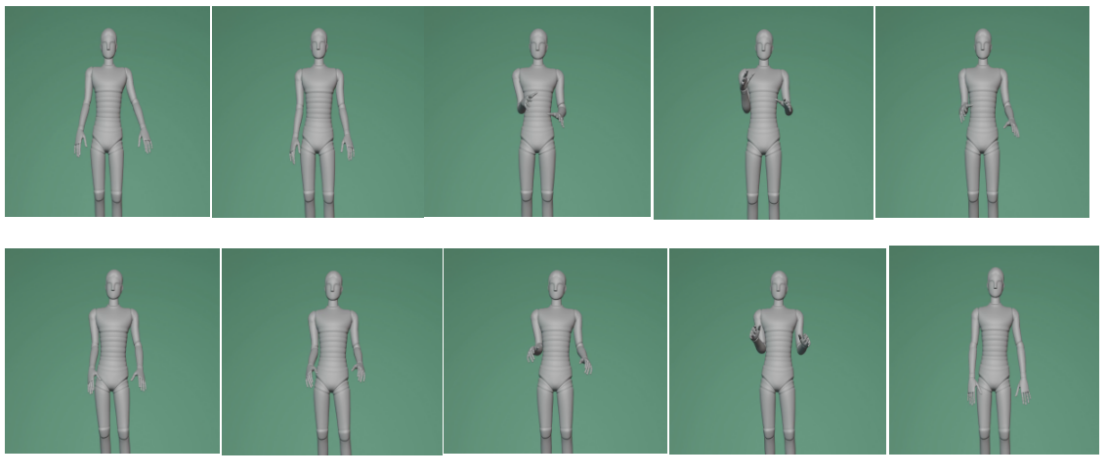


图 5.5　特征融合编码器产生的动作序列

Figure 5.5　Action sequence generated by Feature Fusion Encoder

动作质量通过计算生成动作序列与真实动作序列的欧式距离进行客观评估。其中由时间融合编码器生成的动作序列计算得到的欧式距离为 90.0952，由特征融合编码器生成的动作序列计算得到的欧式距离为 86.0590。通过本文提出的语义判别器对生成的动作语义相关性进行客观评估。其中由时间融合编码器生成的动作序列与音频的特征距离为 0.0275，由特征融合编码器生成的动作序列与

音频的特征距离为 0.0257，说明采用特征对齐的方式融合有助于提高生成动作序列的语义相关性。

图 5.4 是通过时间融合编码器生成的动作序列，图 5.5 是通过特征融合编码器生成的动作序列。如图所示，时间融合编码器产生的动作序列整体上比较动态，但是姿势比较夸张，看着不合理。相对于时间融合编码器的效果，特征融合编码器对应的动作序列较静态，但是没有明显出错的动作帧。

5.4 消融实验

本实验为了探究不同损失项对动作效果的影响。由于本文涉及了两种编码器结构，分别针对两种编码器对应的方法做了消融实验。具体而言，本实验包括以下四个消融实验：只采用重建损失训练网络，采用重建损失和 KL 散度训练网络，采用重建损失、KL 散度和动作质量判别器训练网络以及采用重建损失、KL 散度、动作质量判别器和语义判别器训练网络。每个实验均进行了 100 次迭代训练，并选择在训练过程中损失最小时对应的权重进行测试。对于生成的动作序列，分别采用和真实动作序列的欧式距离和语义判别器作为其动作质量和语义相关性的客观评估方式。

5.4.1 时间融合编码器的消融实验

表 5.4 时间融合编码器的消融实验结果

Loss	动作质量	语义相关性
L_{rec}	76.9586	**0.0229**
$L_{rec} + L_{kl}$	60.2469	0.0254
$L_{rec} + L_{kl} + L_{quality}$	**49.9033**	0.0240
$L_{rec} + L_{kl} + L_{quality} + L_{sync}$	54.8572	0.0230

该编码器对应消融实验的评分如表 5.4 所示。随着损失项的增加，在客观评估指标上，动作质量和语义相关性整体上有一定程度的提高。各实验生成的动作帧序列如图 5.6 所示。图（1）显示了只使用重建损失训练网络生成的动作序列。动作整体上运动幅度不大。图（2）显示了使用重建损失和 KL 散度训练网络生

成的动作序列。为了验证 KL 散度是否能提高动作的多样性，在测试时对于同一条语音数据，分别指定了 2 个不同的说话人信息，但是生成的动作序列完全一样。相对于只使用重建损失对应的动作序列，该序列偶尔会出现异常的动作帧。图（3）显示了同时使用重建损失、KL 散度和质量判别器训练网络生成的动作序列。相比于前两个实验对应的动作序列，该动作序列更加动态，并且偶尔伴随有手势动作。图（4）显示了同时使用重建损失、KL 散度、质量判别器和语义判别器训练网络生成的动作序列。相比于前三个实验，该序列对应的手势更加准确，更接近说话时可能做出的手势。

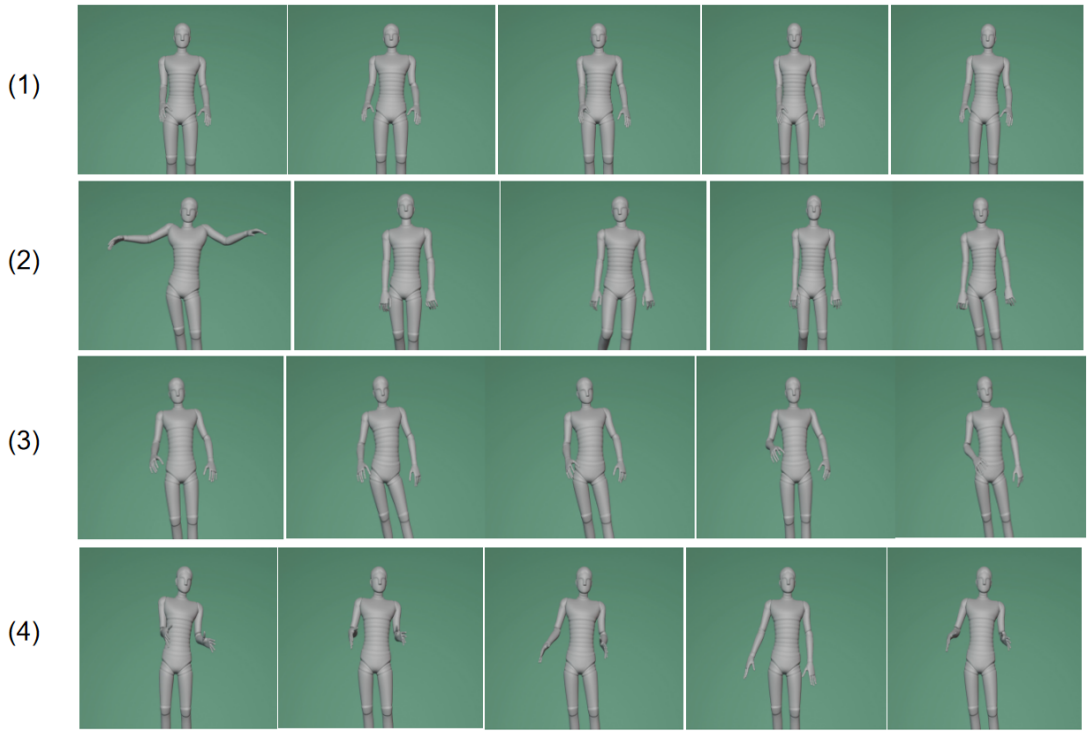


图 5.6　时间融合编码器：消融实验生成的动作帧序列

Figure 5.6　Time Fusion Encoder:Action frame sequences generated by ablation experiments

根据上述实验可得到如下结论。首先，仅使用重建损失训练网络生成的动作序列缺乏动态和多样性，说明仅仅优化动作序列的外观特征是不足够的。其次，添加 KL 散度的训练方法可以增加一定的多样性，但仍然存在同一条语音数据生成的动作序列完全一样的问题。这表明，我们需要更多的训练方法来提高动作序列的多样性。接着，使用质量判别器训练网络可以进一步提高动作序列的动态性，使其更加符合说话时可能做出的动作。最后，使用语义判别器训练网络可以进一步提高动作序列的准确性，使其更加接近真实情况中可能出现的手势。

5.4.2 特征融合编码器的消融实验

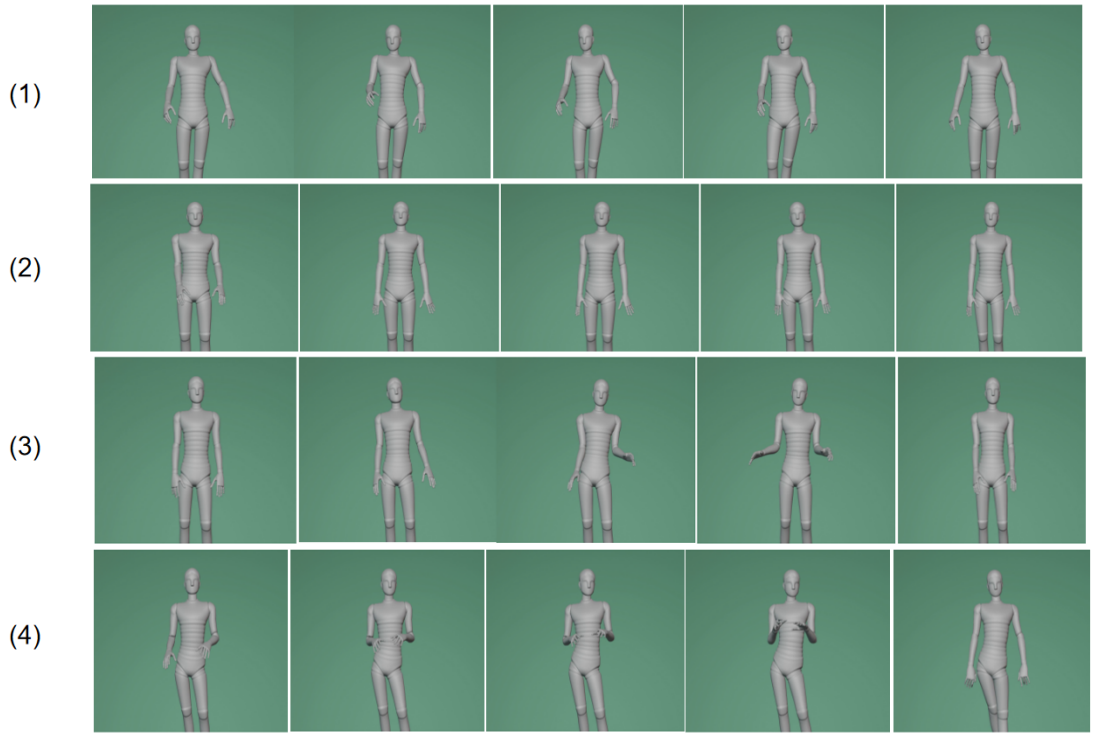


图 5.7 特征融合编码器：消融实验生成的动作序列

Figure 5.7 Feature Fusion Encoder:Action frame sequences generated by ablation experiments

表 5.5 特征融合编码器的消融实验结果

Loss	动作质量	语义相关性
L_{rec}	116.1670	0.0249
$L_{rec} + L_{kl}$	144.3788	0.0242
$L_{rec} + L_{kl} + L_{quality}$	**90.0692**	**0.0223**
$L_{rec} + L_{kl} + L_{quality} + L_{sync}$	92.4340	0.0229

该编码器对应的消融实验的评分如表 5.5 所示。随着损失项的增加，在客观评估指标上，动作质量和语义相关性整体上有一定程度的提高。各实验生成的动作帧序列如图 5.7 所示。图（1）显示了只使用重建损失训练网络生成的动作序列。图（2）显示了使用重建损失和 KL 散度训练网络生成的动作序列。对于同一条语音信息，分别尝试了输入不同的说话人信息，但是其输出的动作序列完全

一样。图（3）显示了使用重建损失、KL 散度和质量判别器训练网络生成的动作序列。相较于前 2 个实验，动作更加动态，但是整体上不断在重复一个相同的动作。图（4）是使用重建损失、KL 散度、质量判别器和语义判别器训练网络生成的动作序列，相比于之前的实验，该序列对应的手势更加真实、自然。

5.5　对比实验

表 5.6　对比实验结果

m	FNA	FSA	FSD	FSI	Our1	Our2
mse	0	392.0760	64.3246	95.3544	**54.8572**	92.4340
sync	0.0286	0.0406	0.0255	0.0248	0.0230	**0.0229**

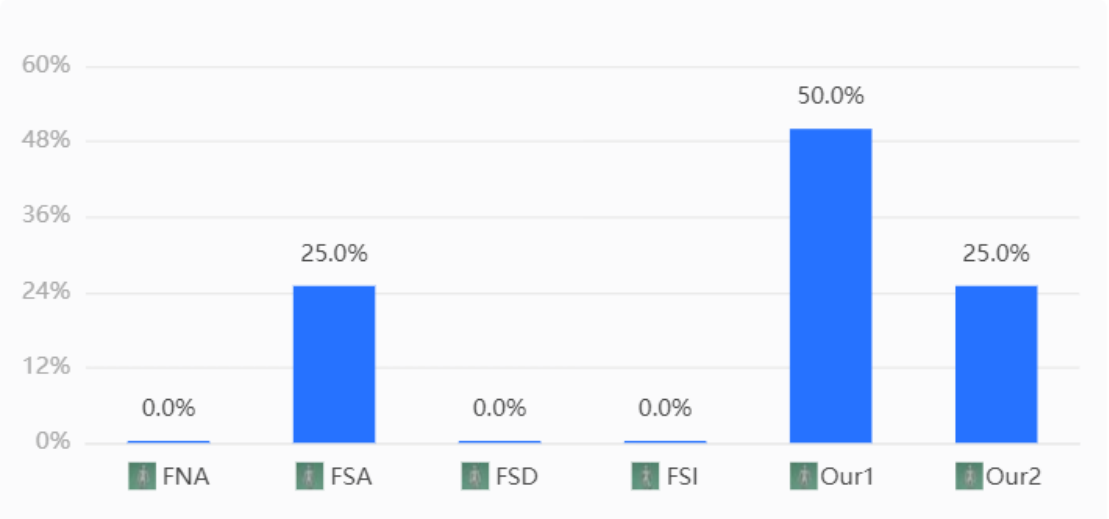


图 5.8　动作质量用户投票结果

Figure 5.8　The results of user study on motion quality

本实验拟跟其他实验结果进行对比。对比的方法是 GENEA Challenge 2022 的参赛方法。对于生成的动作序列，分别采用和真实动作序列的欧式距离和语义判别器作为其动作质量和语义相关性的客观评估方式。比较结果如表 5.6 所示，其中 FNA(Lee 等, 2019) 为动捕数据，FSA(Zhou 等, 2022) 是基于运动图的方法，FSD(Korzun 等, 2022) 计算了每个手势单元的概率，FSI(Lu 等, 2022) 是基于 Seq2Seq 的方法，Our1 表示时间融合编码器对应的方法，Our2 表示特征融合编码

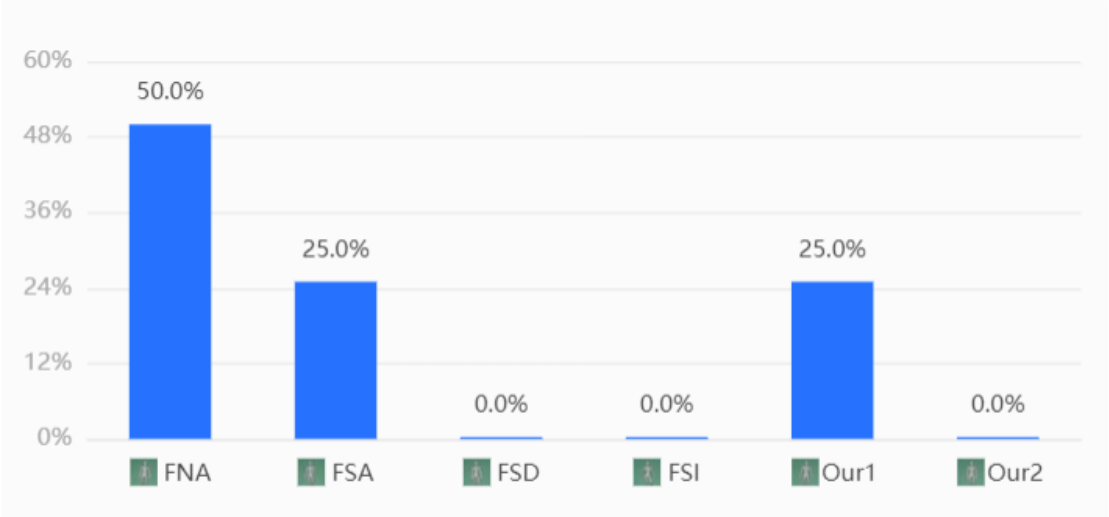


图 5.9　语义相关性用户投票结果

Figure 5.9　The results of user study on appropriateness

器对应的方法，mse 行表示动作质量评分，sync 行表示语义相关性评分。对于动作质量的客观评分，我们的方法（Our1）相比于最佳方法 FSD(Korzun 等, 2022) 获得了 14.71%的性能提升。对于语音相关性的客观评分，我们的方法（Our2）相比于最佳方法 FSI(Lu 等, 2022) 获得了 7%的性能提升。除了采用客观公式评分，我们还针对动作质量及语义相关性设计了调查问卷，用户投票结果分别见图 5.8 和图 5.9。各方法生成的动作帧序列如图 5.10 所示。FSD(Korzun 等, 2022) 和 FSI(Lu 等, 2022) 虽然分别在动作质量、语义相关性上从客观评分上超越了 FSA(Zhou 等, 2022)，但是据图所示，这两个方法产生的动作序列相对于 FSA(Zhou 等, 2022) 比较静态，而 FSA(Zhou 等, 2022) 的手势动作更真实、自然。本文提出的两个方法在视觉上也取得了不错的效果。

5.6　客观评估公式效果验证

在语音生成手势领域，存在很多的客观评价指标。相比于人为投票的主观评估方式，客观评估方式具有成本低、实验结果易复现等优点。然而，不同的论文中使用的客观评估指标和其评估公式、数据集以及动作展示方式等存在差异，因此即使该方法在某个指定评估公式上获得最优结果，也难以说明其实际效果最佳。目前已有的客观评估指标主要涵盖动作质量、动作多样性和语音相关性三个方面，但其评估公式的有效性尚未得到证明。本研究计划在 GENEA Challenge

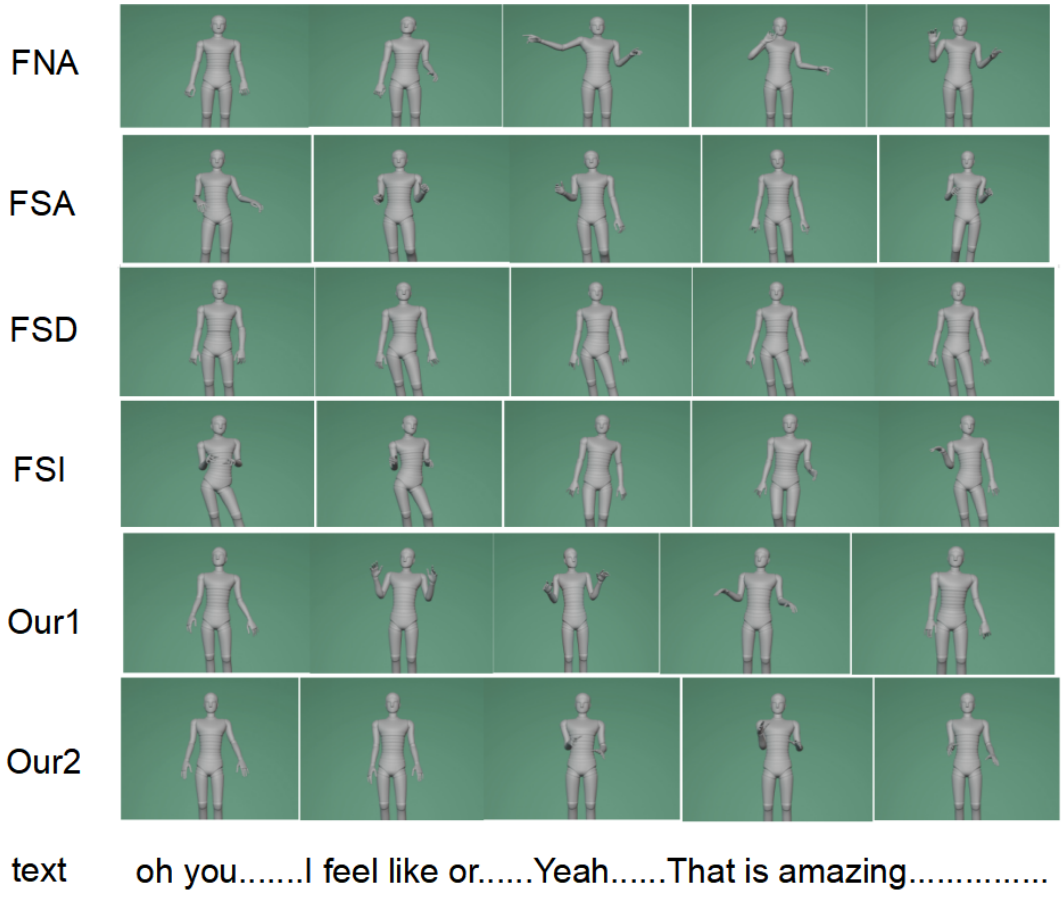


图 5.10　对比实验结果

Figure 5.10　The results of comparative experiments

2022 比赛结果基础上进行实验，统一方法采用的数据集、展示方法等其他变量，以验证这些评估公式的有效性。

5.6.1　实验数据

GENEA Challenge 2022 提供了 40 条测试音频，每个参赛方法提交了 40 条动作序列，共有 10 个方法参与提交了全身动作序列。本实验拟采用这些动作序列作为实验数据，并以各个方法已经获得的主观评分作为实验的目标分值。通过比较由客观评估公式计算出的各个方法的得分，可以对该客观评估公式的有效性进行判断。

GENEA Challenge 2022 的主观评分如图 2.5，图 2.6 所示。如果客观评估公式计算得到的分值趋势与主观评分结果中的趋势相符，则认为该评估方法有效。为了直观比较两者趋势是否一样，对于动作质量方面的评估，展示结果按

照 FSA(Zhou 等, 2022), FNA(Lee 等, 2019), FSC(Ghorbani 等, 2022), FSI(Lu 等, 2022), FSF(Saleh, 2022), FSG(Windle 等, 2022), FSH(Chang 等, 2022), FSD(Korzun 等, 2022), FSB, FBT(Yoon 等, 2019) 的顺序依次展示。对于动作语音相关性方面的评估，展示结果按照 FNA, FSH, FSA, FSI, FSG, FSB, FSC, FSF, FBT, FSD 的顺序依次展示。其中，FNA 表示动作捕捉系统产生的序列，被视为目标序列。

5.6.2 动作质量

动作质量常用于评估生成的动作序列的真实性。常见的动作质量评估方法包括：计算生成序列与目标序列的 L1 距离或 L2 距离，计算生成序列中正确关节点的比例，计算两个序列之间的 FGD 距离以及使用 2 分类器进行评估。

$$L_1 = \|\hat{M} - M\|_1 \quad \ldots (5.10)$$

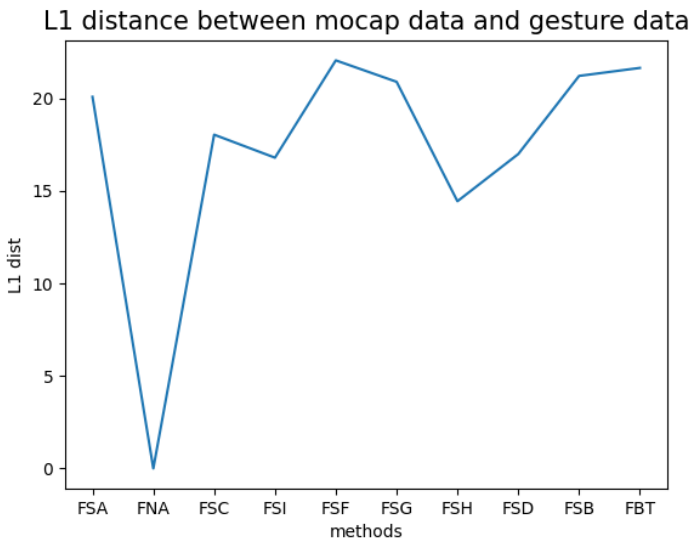


图 5.11 关节点位置的平均 L1 距离

Figure 5.11 Average L1 distance of joint positions

通过计算生成序列与目标序列之间的 L1 距离评估动作质量（式 5.10）。关节点位置间的平均 L1 距离如图 5.11 所示，关节点旋转角度间的平均 L1 距离如图 5.12 所示。根据实验结果显示，对于关节点位置的平均 L1 距离的评估，图形未呈现出与图 2.5 一致的趋势，难以说明该公式的有效性。然而，对于关节点旋转的平均 L1 距离的评估，移除动捕数据 FNA(Lee 等, 2019) 后，其结果趋势与

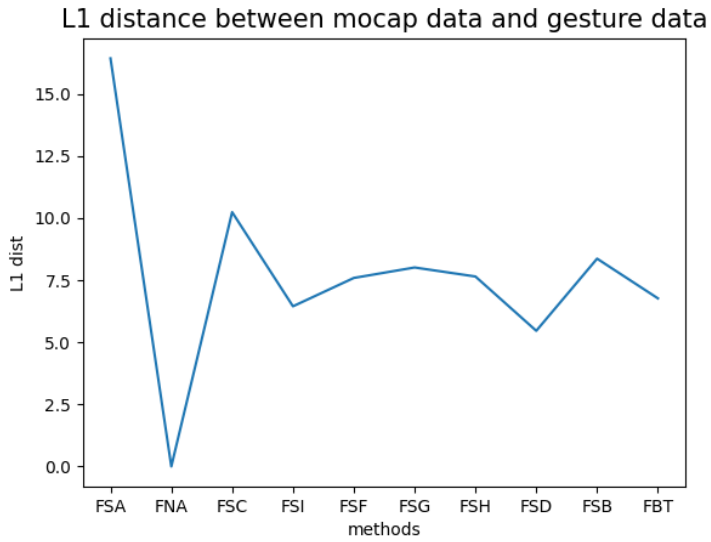


图 5.12　关节点旋转的平均 L1 距离

Figure 5.12　Average L1 distance of joint rotation

图 2.5 大致相似，说明该公式在某种程度上能有效地评估动作的质量。从图中趋势可以看出，该评估方法认为生成的动作序列与真实序列的旋转角度差异越大越好。但是需要注意的是，这种评估方法可能只适用于生成的动作序列是合理的情况下，因为当生成的动作序列不合理时，关节点的旋转可能会随意变化而不受控制，这可能会导致一个很大的得分。

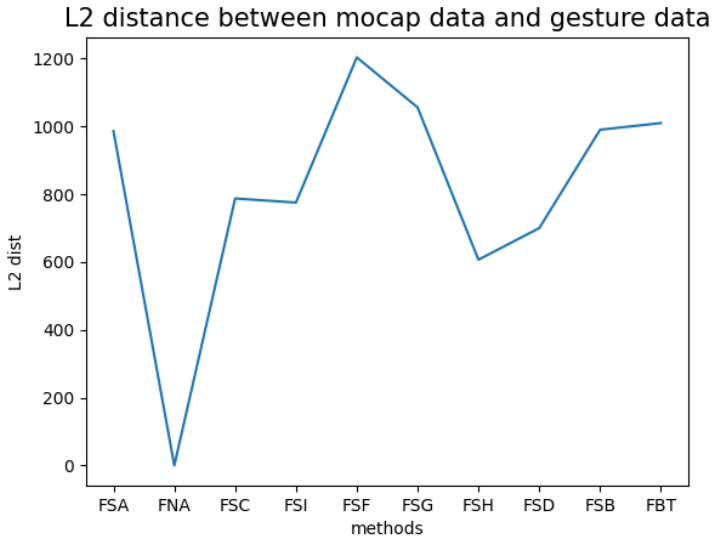


图 5.13　关节点位置的平均 L2 距离

Figure 5.13　Average L2 distance of joint positions

$$L_2 = \|\hat{M} - M\|_2 \quad \ldots (5.11)$$

通过计算生成序列与目标序列之间的 L2 距离评估动作质量（式 5.11）。关节点位置间的平均 L2 距离如图 5.13 所示，关节点旋转角度间的平均 L2 距离如图 5.14 所示。其实验结果跟使用 L1 距离的实验结果相近，说明了使用两个序列间距离进行动作质量评估时，采用动作的旋转表示进行计算是比较有效的。图 5.13 和图 5.11 展示的结果表明，使用 L1 距离计算在 FSG(Windle 等, 2022) 附近能够得到更加准确的趋势。这可能是因为相比于 L2 距离，L1 距离对于离群值的处理更好。

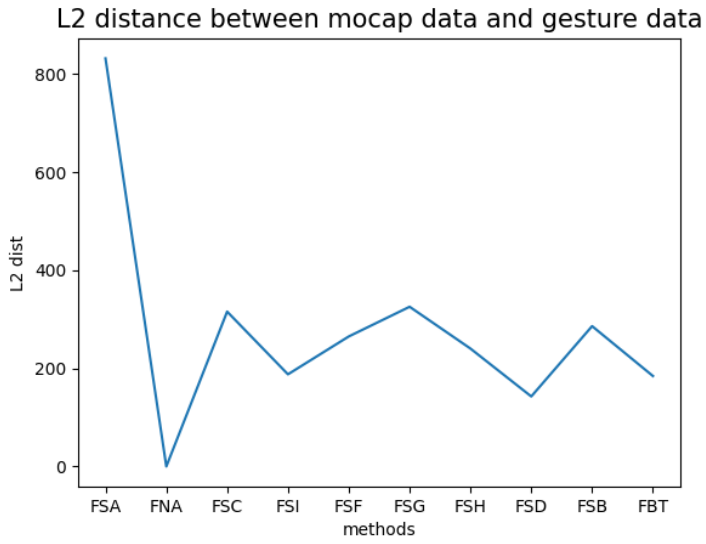


图 5.14　关节点旋转的平均 L2 距离

Figure 5.14　Average L2 distance of joint rotation

通过 2 分类器评估动作质量。采用的 2 分类器结构仍是 4.2.4 章节提到的质量判别器，但是不同于之前通过训练 GAN 的方式训练质量判别器，这里通过将各个参赛方法生成的部分动作序列作为训练数据训练分类器。在 GENEA Challenge 2022 的质量评分结果中，FSA (Zhou 等, 2022) 产生的动作数据好于动捕数据，所以在训练时将 FSA(Zhou 等, 2022) 和动捕系统产生的数据都设置为标签 1，表示高质量数据，其余方法产生的数据设置为标签 0，表示质量低的动作序列。用训练的分类器对各个方法剩余的动作序列进行分类，分类结果如下如所示，其中图 5.15 是使用了速度分支的结果，图 5.16 是没使用速度分支的结果。

实验结果显示，该动作趋势与图 2.5 中的趋势大致相同，但在中间某些方法的表现的存在偏差，这表明该方法具有一定的有效性，但仍存在一定的限制。此外，结果图表明使用速度特征可以在一定程度上提高分类的准确率。

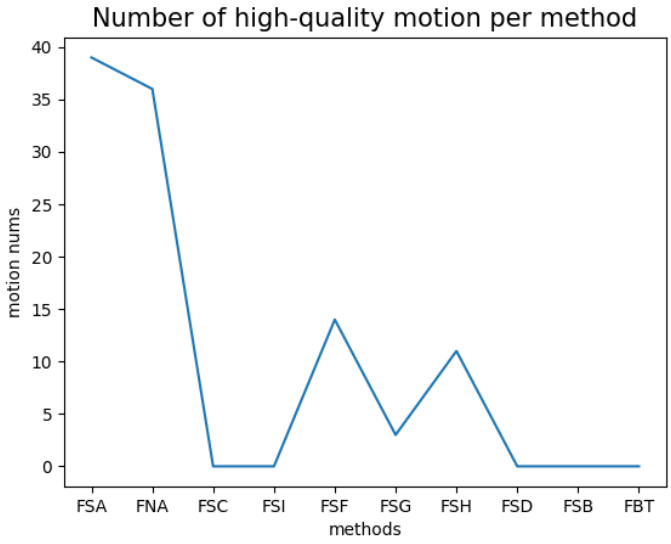


图 5.15　使用了速度分支：2 分类器对动作序列的分类结果

Figure 5.15　Classification results of binary classifier for action sequences when the velocity branch is used

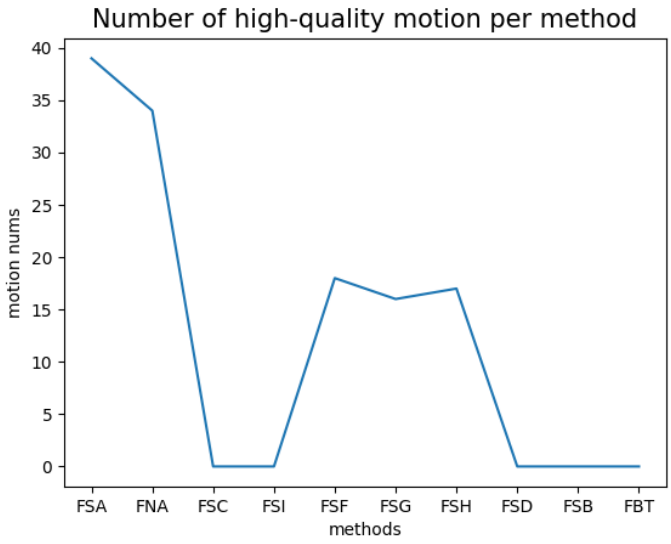


图 5.16　未使用速度分支：2 分类器对动作序列的分类结果

Figure 5.16　Classification results of action sequences by binary classifier when velocity branch is not used

$$FGD(X,\hat{X}) = \left|\mu_r - \mu_g\right|^2 + Tr(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{\frac{1}{2}}) \quad \ldots (5.12)$$

表 5.7　FGD 评估结果

m	FSA	FNA	FSC	FSI	FSF	FSG	FSH	FSD	FSB	FBT
FGD	2e+21	-3e-5	1e+28	758	5e+20	3e+21	874	**715**	738	929

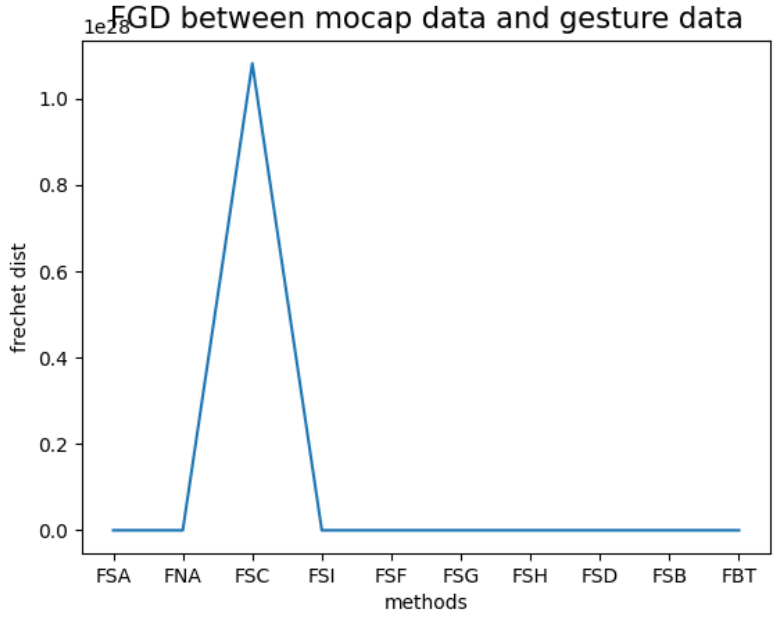


图 5.17　用 FGD 评估各方法动作质量的结果

Figure 5.17　Evaluation of the action quality of comparing method based on FGD

在图像生成任务中，FID(Frechet Inception Distance)(Heusel 等, 2017) 被用作计算真实样本和生成样本在特征空间中的距离的指标。FID 基于 Inception 网络提取的特征进行计算，使用高斯模型对特征空间进行建模，并计算两个特征之间的距离。较低的 FID 意味着生成的图像有较高的质量和多样性。Frechet Distance 通过计算两曲线距离，来判断两个曲线的相似度，计算结果越小，说明相似度越高。对于两个给定的多元高斯 $X \sim N(\mu_r, \Sigma_r)$，$\hat{X} \sim N(\mu_g, \Sigma_g)$，其 Frechet 距离计算公式如式 5.12，式中 FGD 是 Frechet Gesture Distance 的缩写，表示手势的特征距离。参考于 AI Choreographer(Li, Yang 等, 2021)，本实验主要计算生成序列与动捕序列间动态特征的距离。用关节点旋转角度作为输入，每个关节点的动态特

征由其水平速度、垂直速度、加速度共同构成。指定速度的垂直方向为 Y 轴，设置滑动窗口的大小为 2。水平速度是通过累加滑动窗口的速度，再除以序列时间得到的 X 和 Z 方向的平均速度。垂直速度是通过累加滑动窗口的速度，再除以序列时间得到的 Y 方向的平均速度。加速度是通过累加滑动窗口的加速度，再除以序列长度得到的平均加速度。生成序列与动捕序列之间动态特征的距离数值如表 5.7 所示，趋势如图 5.17 所示。该数值和图直观上感觉比较异常，可能是因为 AI Choreographer(Li, Yang 等, 2021) 是音乐生成舞蹈的任务，动作节奏是主要考量的一个方面，而本文是音频生成手势序列，相对而言，本任务对应的动作节奏不够强烈。

5.6.3　动作多样性

有两个方法可以评估动作的多样性。

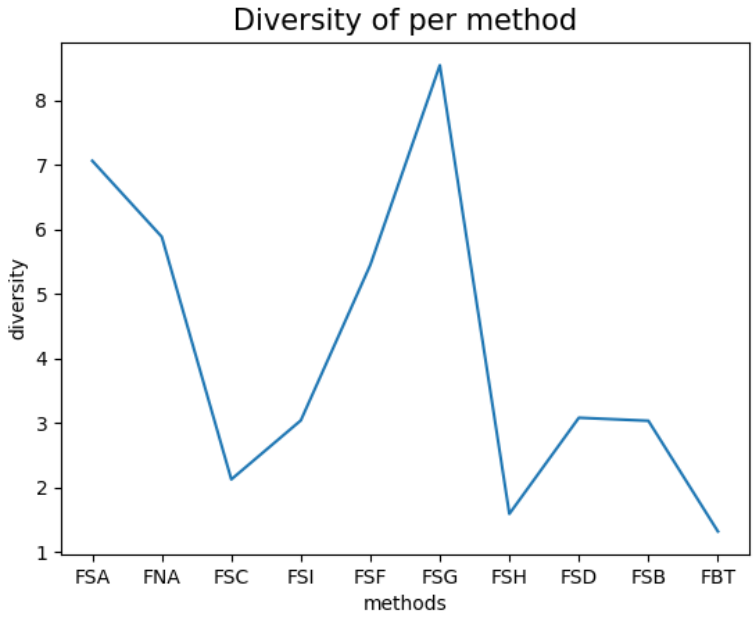


图 5.18　多样性评估结果

Figure 5.18　Diversity assessment results

通过计算一个长动作序列中包含有多少个不同的动作来评估动作的多样性。先将生成的动作划分成等长不重叠的动作切片，再计算这些切片的平均 L1 距离作为多样性指数。在本实验中，选择的切片长度为 300，使用关节点的旋转角度参与计算。具体的计算公式如式 5.13，式中 Diversity 表示动作的多样性得分。结

果如图 5.18 所示。忽略图中 FSF(Saleh, 2022) 和 FSG(Windle 等, 2022) 的评分，其他方法的评分趋势跟图 2.5 中的趋势大致相同，说明该公式也存在一定的有效性。

$$Diversity = \frac{1}{N \times \lceil \frac{N}{2} \rceil} \sum_{a_1=1}^{N} \sum_{a_2=a_1+1}^{N} \|\hat{M}_{a_1} - \hat{M}_{a_2}\|_1 \quad \ldots (5.13)$$

第二种评估动作多样性的方法为，对于同一段给出的音频，测试其可以产生多少个不同的动作序列。计算公式如式 5.14，式中 Multimodality 表示动作的多峰性。但是由于 GENEA Challenge 2022 并没有公开每个参赛方法根据同一个音频生成多个动作序列的结果，本文没有测试该指标。

$$Multimodality = \frac{1}{N \times \lceil \frac{N}{2} \rceil} \sum_{a=1}^{N} \sum_{b=a+1}^{N} \|\hat{M}_{a} - \hat{M}_{b}\|_1 \quad \ldots (5.14)$$

5.6.4 语音相关性

语音相关性用于评估生成动作和音频的相关性。其有两方面含义，一个是指语音动作在节奏上的相关性，这个可参考于音乐生成舞蹈领域的客观评估公式，另一个是指语音动作在语义上的相关性，这个目前尚未看到有论文用客观评估方法衡量。

对于节奏相关性，其需要先分别计算音频节奏和动作节奏。音频节奏可以根据 Librosa 内置的函数检测到拍子事件对应的帧号，最终得到维度为 $(f_a,)$ 的音频节拍，其中有节拍发生的位置值为 1，其余位置值为 0。动作节奏的计算方法为，先通过相邻两帧的帧差计算得到动作速度，速度的局部最小值作为动作的节奏特征，最终得到维度为 $(f_m,)$ 的动作节拍，其中有节拍发生的位置值为 1，其余位置值为 0。将每个动作节拍与和其最近的音频节拍的平均距离作为节奏相关性的衡量值。具体计算公式如式 5.15(Li, Yang 等, 2021)。

$$BeatAlign = \frac{1}{m} \sum_{i=1}^{m} exp(-\frac{min_{\forall t_j^y \in B^y} \|t_i^x - t_i^y\|^2}{2\sigma^2}) \quad \ldots (5.15)$$

使用关节点位置的计算结果如图 5.19 所示。实验结果表明，图中的得分趋势并没有跟图 2.6 中趋势重合的部分，难以说明该方法的有效性。使用关节点旋转角度的计算结果如图 5.20 所示。实现结果显示，部分方法的得分趋势与图 2.6

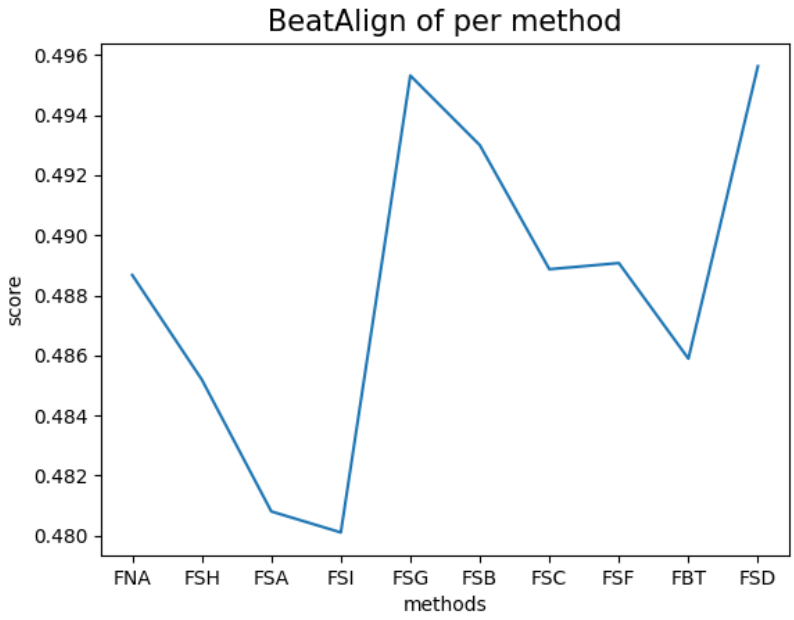


图 5.19　使用关节点位置计算的节奏相关性

Figure 5.19　Rhythm dependencies computed by joint positions

中趋势相似，表明该方法一定程度上具有评估效能，但同时也存在大量偏差的评分，说明该方法在评估动作与节奏相关性时具有很大的局限性。

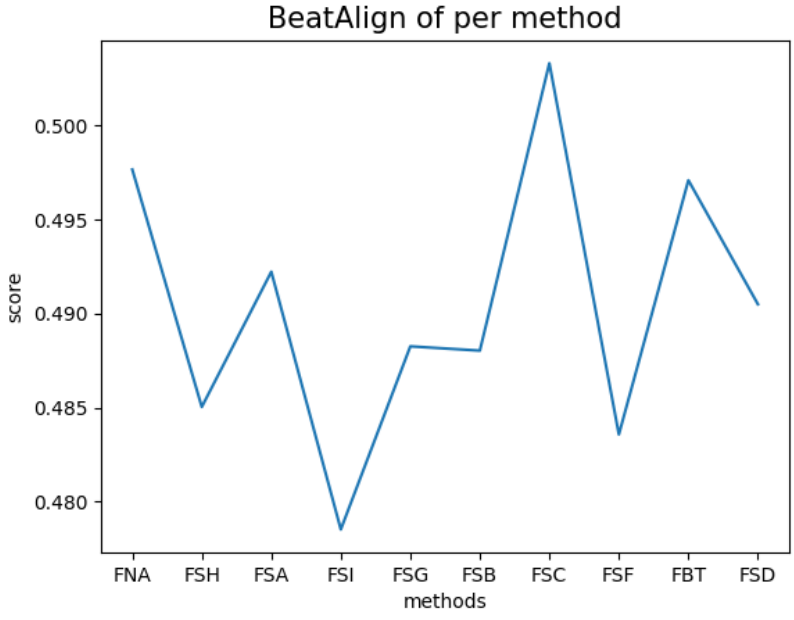


图 5.20　使用关节点旋转计算的节奏相关性

Figure 5.20　Rhythm dependencies computed by joint rotations

5.7 本章小结

本章围绕本文提出的语音生成手势模型进行了一系列实验，旨在验证模型的有效性和探究模型的不同方面对动作效果的影响。具体实验包括验证语义判别器和质量判别器的有效性，比较时间融合编码器和特征融合编码器的效果，进行消融实验探究不同损失项对动作效果的影响，以及与其他语音生成手势的方法进行对比实验。此外，还进行了客观评估公式效果验证，其中客观评估公式包括动作质量、动作多样性和语音相关性三个方面的评估。通过这些实验，本文验证了所提出模型的有效性和优越性，并为语音生成手势领域的进一步研究提供了有价值的参考。

第 6 章　结论和展望

6.1　本文工作总结

本文首先对音频生成动作序列领域进行了全面的调研和总结。从数据集、研究方法、评估指标、音频表示和动作表示等方面，分析了该领域的相关知识和现有方法存在的限制和不足。为了解决这些限制，本文提出了一些改进方法，并进行了一系列的尝试。

此外，本文还探索了虚拟人的驱动方式，即动捕驱动和视频驱动，根据实践经验，设计了适用于工业场景的手势生成算法。针对动作质量和语义相关性不足的问题，本文提出了质量判别器和语义判别器。通过这两种判别器，可以分别提高生成的动作序列在质量和语义相关性方面的表现。针对多模态数据的融合问题，本文探索了两种特征融合方式，即分别在时间维度和特征维度上融合不同模态的数据。实验证明，将不同模态数据在时间维度对齐后，在特征维度融合有助于提高生成序列（动作）与输入序列（语音）的相关性。此外，针对该领域存在的客观评估公式不统一的问题，本文整理了这些评估方法，并通过使用它们对 GENEA Challenge 2022 的参赛方法进行评估来验证了这些方法的有效性。通过实验结果，本文发现使用生成序列与真实序列关节点旋转的 L1 距离或者 L2 距离来评估生成序列的质量比较有效，通过计算序列中不同动作个数来评估动作多样性，也在一定程度上适用于动作质量评估。对于语义相关性缺失客观评估公式的问题，本文提出的语义判别器可以作为评估语义相关性的工具。

然而，本文的工作还存在一些限制。在动作质量、动作多样性和语音相关性上，本文的工作都还有待于提高。要使得该工作最终应用于工业场景，需要在这些方面继续探索和优化。

6.2　未来展望

语音生成手势领域现在仍存在很多的研究问题值得深入探索，后续的研究可以从下面几个方面展开。

首先，客观评估公式是语音生成手势领域一个重要的研究问题。目前的客观

评估公式还不能准确地评估动作序列，只能通过人工评判。因此，研究人员需要设计高效的客观评估公式，使其能够更好、更准确地在动作质量、动作多样性、动作与语音相关性方面进行评估。这将有助于提高实验的效率和可重复性，推动语音生成手势领域的发展。

其次，语音生成手势方法的高效准确是该领域的另一个重要研究方向。现有的方法在动作质量、动作多样性和语音相关性方面仍有一定的限制。因此，未来的研究可以探索更高效、更准确的语音生成手势方法。

目前语音生成手势领域主要面向的是单人演讲场景，对于多人对话等场景的研究还比较缺乏。然而，在现实生活中，多人对话是很常见的情况，因此在未来的研究中，可以考虑引入其他信息，例如多人对话和语音背景，以提高语音生成手势系统的适用性和鲁棒性。

参考文献

Autodesk, n.d. FBX 用户文档 [EB/OL]. https://help.autodesk.com/view/FBX/2020/ENU/?guid=FBX_Developer_Help_animation_animation_data_structures_animation_classes_interrelation_html.

AKSAN E, CAO P, KAUFMANN M, et al., 2020. Attention, please: A spatio-temporal transformer for 3d human motion prediction[J]. arXiv preprint arXiv:2004.08692, 2(3): 5.

ATHANASIOU N, PETROVICH M, BLACK M J, et al., 2022. TEACH: Temporal Action Composition for 3D Humans[J]. arXiv preprint arXiv:2209.04066,

BAI S, KOLTER J Z, KOLTUN V, 2018. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling[J]. arXiv:1803.01271,

BENGIO S, VINYALS O, JAITLY N, et al., 2015. Scheduled sampling for sequence prediction with recurrent neural networks[J]. Advances in neural information processing systems, 28.

BOJANOWSKI P, GRAVE E, JOULIN A, et al., 2017. Enriching word vectors with subword information[J]. Transactions of the association for computational linguistics, 5: 135-146.

BROMLEY J, GUYON I, LECUN Y, et al., 1993. Signature verification using a" siamese" time delay neural network[J]. Advances in neural information processing systems, 6.

CAO Z, SIMON T, WEI S E, et al., 2017. Realtime multi-person 2d pose estimation using part affinity fields[C]//Proceedings of the IEEE conference on computer vision and pattern recognition: 7291-7299.

CASIEZ G, ROUSSEL N, VOGEL D, 2012. 1€ filter: a simple speed-based low-pass filter for noisy input in interactive systems[C]//Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: 2527-2530.

CHANG C J, ZHANG S, KAPADIA M, 2022. The IVI Lab entry to the GENEA Challenge 2022–A Tacotron2 based method for co-speech gesture generation with locality-constraint attention mechanism[C]//Proceedings of the 2022 International Conference on Multimodal Interaction: 784-789.

CHO K, VAN MERRIËNBOER B, GULCEHRE C, et al., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078,

CHUNG J S, ZISSERMAN A, 2017. Out of time: automated lip sync in the wild[C]//Computer

Vision–ACCV 2016 Workshops: ACCV 2016 International Workshops, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part II 13: 251-263.

DOU Q, LU Y, MANAKUL P, et al., 2021. Attention forcing for machine translation[J]. arXiv preprint arXiv:2104.01264,

FERSTL Y, MCDONNELL R, 2018. Investigating the use of recurrent motion modelling for speech gesture generation[C]//Proceedings of the 18th International Conference on Intelligent Virtual Agents: 93-98.

GHORBANI S, FERSTL Y, CARBONNEAU M A, 2022. Exemplar-based stylized gesture generation from speech: An entry to the GENEA Challenge 2022[C]//Proceedings of the 2022 International Conference on Multimodal Interaction: 778-783.

GINOSAR S, BAR A, KOHAVI G, et al., 2019. Learning individual styles of conversational gesture [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition: 3497-3506.

GOODFELLOW I J, POUGET-ABADIE J, MIRZA M, et al., 2020. Generative adversarial networks. [J]. Commun. Acm, 63(11): 139-144.

HEUSEL M, RAMSAUER H, UNTERTHINER T, et al., 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium[J]. Advances in neural information processing systems, 30.

HOCHREITER S, SCHMIDHUBER J, 1997. Long short-term memory[J]. Neural computation, 9(8): 1735-1780.

HODRICK R J, PRESCOTT E C, 1997. Postwar US business cycles: an empirical investigation[J]. Journal of Money, credit, and Banking, 1-16.

KARRAS T, AILA T, LAINE S, et al., 2017. Audio-driven facial animation by joint end-to-end learning of pose and emotion[J]. ACM Transactions on Graphics (TOG), 36(4): 1-12.

KORZUN V, BELOBORODOVA A, ILIN A, 2022. ReCell: replicating recurrent cell for auto-regressive pose generation[C]//Companion Publication of the 2022 International Conference on Multimodal Interaction: 94-97.

KOVAR L, GLEICHER M, PIGHIN F, 2008. Motion graphs[G]//ACM SIGGRAPH 2008 classes: 1-10.

KUCHERENKO T, HASEGAWA D, HENTER G E, et al., 2019. Analyzing input and output representations for speech-driven gesture generation[C]//Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents: 97-104.

KUCHERENKO T, JONELL P, YOON Y, et al., 2021. A large, crowdsourced evaluation of gesture generation systems on common data: The GENEA Challenge 2020[C]//26th international conference on intelligent user interfaces: 11-21.

KUNDU J N, BUCKCHASH H, MANDIKAL P, et al., 2020. Cross-conditioned recurrent networks for long-term synthesis of inter-person human motion interactions[C]//Proceedings of the IEEE/CVF winter conference on applications of computer vision: 2724-2733.

LEE G, DENG Z, MA S, et al., 2019. Talking with hands 16.2 m: A large-scale dataset of synchronized body-finger motion and audio for conversational motion analysis and synthesis[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision: 763-772.

LI J, YIN Y, CHU H, et al., 2020. Learning to generate diverse dance motions with transformer[J]. arXiv preprint arXiv:2008.08171,

LI J, KANG D, PEI W, et al., 2021. Audio2gestures: Generating diverse gestures from speech audio with conditional variational autoencoders[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision: 11293-11302.

LI R, YANG S, ROSS D A, et al., 2021. Ai choreographer: Music conditioned 3d dance generation with aist++[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision: 13401-13412.

LIN C Y, 2004. Rouge: A package for automatic evaluation of summaries[C]//Text summarization branches out: 74-81.

LOPER M, MAHMOOD N, ROMERO J, et al., 2015. SMPL: A skinned multi-person linear model [J]. ACM transactions on graphics (TOG), 34(6): 1-16.

LU S, FENG A, 2022. The DeepMotion entry to the GENEA Challenge 2022[C]//Proceedings of the 2022 International Conference on Multimodal Interaction: 790-796.

LUGARESI C, TANG J, NASH H, et al., 2019. Mediapipe: A framework for building perception pipelines[J]. arXiv preprint arXiv:1906.08172,

MCFEE B, RAFFEL C, LIANG D, et al., 2015. librosa: Audio and music signal analysis in python [C]//Proceedings of the 14th python in science conference: vol. 8: 18-25.

MCNEILL D, 1992. Hand and mind1[J]. Advances in Visual Semiotics, 351.

MIHAYLOVA T, MARTINS A F, 2019. Scheduled sampling for transformers[J]. arXiv preprint arXiv:1906.07651,

PAPINENI K, ROUKOS S, WARD T, et al., 2002. Bleu: a method for automatic evaluation of machine translation[C]//Proceedings of the 40th annual meeting of the Association for Computa-

tional Linguistics: 311-318.

PAVLLO D, FEICHTENHOFER C, GRANGIER D, et al., 2019. 3d human pose estimation in video with temporal convolutions and semi-supervised training[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition: 7753-7762.

PETROVICH M, BLACK M J, VAROL G, 2022. TEMOS: Generating diverse human motions from textual descriptions[C]//Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII: 480-497.

PRAJWAL K, MUKHOPADHYAY R, NAMBOODIRI V P, et al., 2020. A lip sync expert is all you need for speech to lip generation in the wild[C]//Proceedings of the 28th ACM International Conference on Multimedia: 484-492.

QIAN S, TU Z, ZHI Y, et al., 2021. Speech drives templates: Co-speech gesture synthesis with learned templates[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision: 11077-11086.

RADFORD A, NARASIMHAN K, SALIMANS T, et al., 2018. Improving language understanding by generative pre-training[J].

RAVANELLI M, ZHONG J, PASCUAL S, et al., 2020. Multi-task self-supervised learning for robust speech recognition[C]//ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 6989-6993.

REN X, LI H, HUANG Z, et al., 2020. Self-supervised dance video synthesis conditioned on music [C]//Proceedings of the 28th ACM International Conference on Multimedia: 46-54.

SALEH K, 2022. Hybrid seq2seq architecture for 3D co-speech gesture generation[C]//Proceedings of the 2022 International Conference on Multimodal Interaction: 748-752.

SANH V, DEBUT L, CHAUMOND J, et al., 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter[J]. arXiv preprint arXiv:1910.01108,

SHOEMAKE K, 1985. Animating rotation with quaternion curves[C]//Proceedings of the 12th annual conference on Computer graphics and interactive techniques: 245-254.

SUN Y, BAO Q, LIU W, et al., 2021. Monocular, one-stage, regression of multiple 3d people[C]//Proceedings of the IEEE/CVF international conference on computer vision: 11179-11188.

SUTSKEVER I, VINYALS O, LE Q V, 2014. Sequence to sequence learning with neural networks [J]. Advances in neural information processing systems, 27.

TAKEUCHI K, KUBOTA S, SUZUKI K, et al., 2017. Creating a gesture-speech dataset for speech-based automatic gesture generation[C]//HCI International 2017–Posters' Extended Abstracts:

19th International Conference, HCI International 2017, Vancouver, BC, Canada, July 9–14, 2017, Proceedings, Part I 19: 198-202.

TIAN G, YUAN Y, LIU Y, 2019. Audio2face: Generating speech/face animation from single audio with attention-based bidirectional lstm networks[C]//2019 IEEE international conference on Multimedia & Expo Workshops (ICMEW): 366-371.

TSUCHIDA S, FUKAYAMA S, HAMASAKI M, et al., 2019. AIST Dance Video Database: Multi-Genre, Multi-Dancer, and Multi-Camera Database for Dance Information Processing.[C]//ISMIR: vol. 1: 5: 6.

VASWANI A, SHAZEER N, PARMAR N, et al., 2017. Attention is all you need[J]. Advances in neural information processing systems, 30.

WINDLE J, GREENWOOD D, TAYLOR S, 2022. UEA Digital Humans entry to the GENEA Challenge 2022[C]//Proceedings of the 2022 International Conference on Multimodal Interaction: 771-777.

XU J, ZHANG W, BAI Y, et al., 2022. Freeform body motion generation from speech[J]. arXiv preprint arXiv:2203.02291,

YANG S, WU Z, LI M, et al., 2022. The ReprGesture entry to the GENEA Challenge 2022[J]. arXiv preprint arXiv:2208.12133,

YOON Y, CHA B, LEE J H, et al., 2020. Speech gesture generation from the trimodal context of text, audio, and speaker identity[J]. ACM Transactions on Graphics (TOG), 39(6): 1-16.

YOON Y, KO W R, JANG M, et al., 2019. Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots[C]//2019 International Conference on Robotics and Automation (ICRA): 4303-4309.

YOON Y, WOLFERT P, KUCHERENKO T, et al., 2022. The GENEA Challenge 2022: A large evaluation of data-driven co-speech gesture generation[C]//Proceedings of the 2022 International Conference on Multimodal Interaction: 736-747.

YU C, TAPUS A, 2020. Srg 3: Speech-driven robot gesture generation with gan[C]//2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV): 759-766.

ZHOU C, BIAN T, CHEN K, 2022. GestureMaster: Graph-based speech-driven gesture generation [C]//Proceedings of the 2022 International Conference on Multimodal Interaction: 764-770.

ZHOU Y, BARNES C, LU J, et al., 2019. On the continuity of rotation representations in neural networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition: 5745-5753.

ZHUANG W, WANG C, XIA S, et al., 2020. Music2Dance: DanceNet for Music-driven Dance Generation[J]. arXiv e-prints, arXiv-2002.

致　谢

在我的学习和论文完成过程中，有许多人给予了我帮助和支持，我在此向以下人士表示深深的感谢：

首先，我要感谢我的导师孙老师。很幸运有他作为自己的导师，他在我的研究方向上给予了我很多的指导和支持，在我论文的写作和修改过程中给予了我宝贵的建议和意见。同时，我也要祝愿孙老师的未来发展更加顺利，取得更加辉煌的成就，为学术事业做出更大的贡献。

感谢在论文完成过程中给予我帮助的同学们，感恩遇见。他们不仅帮我解决了许多问题，还给我提供了许多新的思路和观点，让我在论文的撰写中受益匪浅。

感谢家人给予的关爱和支持，是他们的无私奉献和悉心呵护，让我在学习和生活中能够充满动力和信心，才能够顺利地完成论文。

最后，感谢坚持到现在的自己。一开始对毕设抱有很高的期待，但实现过程中遇到了很多意料之外的错误，感谢自己尝试了所有可能的方法并坚持到了现在，没有留下遗憾。希望今后还能继续在虚拟人领域发展。

作者简历及攻读学位期间发表的学术论文与研究成果

作者简历：

2016 年 9 月——2020 年 6 月，在成都信息工程大学软件工程专业获得学士学位。

2020 年 9 月——2023 年 6 月，在上海科技大学计算机科学与技术专业攻读硕士学位。

申请或已获得的专利：

杨惠，向钊豫，吴红 一种虚拟人驱动方法、装置、设备及可读存储介质 CN 202310098261.0