arXiv:2601.08179v1 [cs.CV] 13 Jan 2026

指令驱动的三维面部表情生成与转换

Anh H. Vo、Tae-Seok Kim、Hulin Jin、Soo-MiChoi 和 Yong-Guk Kim*

摘要——**三维虚拟化身通常具有六种基本面部表情之一。为了模拟逼真的情绪变化，我们应能够呈现任意两种表情之间的面部转换。本研究提出了一种新的指令驱动面部表情生成框架，该框架能够生成三维人脸，并以人脸图像为起点，将面部表情从一种指定表情转换为另一种。我们引入了指令驱动的面部表情分解器（IFED）模块，以促进多模态数据学习，并捕捉文本描述与面部表情特征之间的相关性。随后，我们提出指令到面部表情转换（I2FET）方法，该方法利用 IFED 和顶点重建损失函数来改善潜向量的语义理解，从而根据给定指令生成面部表情序列。最后，我们提出面部表情转换模型，以生成平滑的面部表情转换。大量评估表明，所提出的模型在 CK+ 和 CelebV-HQ 数据集上的表现优于最先进的方法。结果显示，我们的框架能够根据文本指令生成面部表情轨迹。鉴于文本提示使我们能够对人类情绪状态进行多样化描述，面部表情的种类及其转换方式可以大幅扩展。我们期待该框架能在各种实际应用中发挥作用。有关项目的更多信息，请访问 https://vohoanganh.github.io/tg3dfet/**

索引词—**指令驱动的面部表情与过渡、可控虚拟化身、CK+ 和 CelebV-HQ 数据集。**

I. 引言

面Ace 和面部表情生成近年来受到广泛关注，因为它们广泛应用于增强现实/虚拟现实、游戏产业、电影制作或人机界面等众多领域。面部生成可根据面部是二维还是三维（2D 或三维人脸）分为两类。对于二维情况，已有若干方法 [1], [2], [3], [4], [5]被提出，用于生成面部图像，方法是多模态输入，例如文本、草图和分割。以往研究 [6], [2], [7], [8] 尝试根据情绪标签或文本提示输入来解决面部表情生成问题。然而，这些方法对输出图像几何信息的控制能力有限，且仅适用于特定应用。对于三维人脸，已有多项研究 [9], [10], [11], [12], [13]展示了从二维人脸生成三维人脸的方法。这些模型结合源图像中的几何信息（例如姿态、形状和纹理）来生成输出图像。

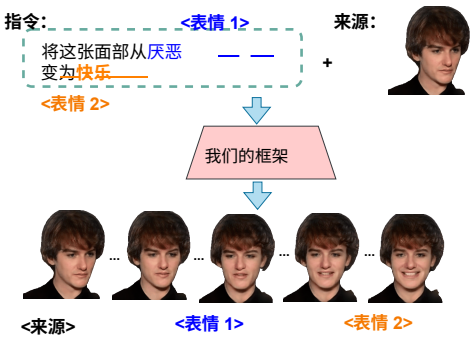


图 1：我们框架的示意图。该框架以文本指令和人脸图像（来源）作为输入，并由此生成三维人脸。随后，它通过中性表情将面部表情从厌恶（表情 1）转换为快乐（表情 2）。

其他研究人员 [14], [15] 根据文本提示生成特定的三维面部表情。他们利用 CLIP（对比语言-图像预训练）或 NeRF（神经辐射场）学习用于操控面部表情的潜在代码。然而，NeRF 需要大量的计算资源和时间。例如，[15] 学习面部参数，并根据文本提示生成面部纹理和表情。尽管这些方法能够生成特定的面部表情，但它们并未解决根据所提供的指令实现面部表情过渡的问题。与此同时，其他研究 [16], [17], [18], [19], [20] 提出从 RGB视频中控制面部表情，或生成具有可控面部表情、头部姿态和肩部动作的肖像图像 [21]。此外， [22], [23] 利用面部关键点轨迹，在网格人脸上生成多个四维表情过渡。

这些研究促使我们提出了一种新框架，可根据源图像生成三维人脸。随后，面部表情会根据用户提供的提示文本进行转换，以实现两种任意表情之间的面部表情转换。近期研究采用文本驱动方法，通过神经辐射场生成变形人脸，或通过 StyleGAN [8], [14] 创建动画人脸。这些方法通过指定某种情绪（例如“这个人很快乐”）来生成一系列面部表情。与之相反，如图 1 所示，我们的研究利用文本指令控制面部表情的变化，例如“将这张脸从厌恶转换为快乐”。然而，在依赖文本指令时，理解其中指定的目标表情仍是一项挑战。

A H Vo、T-S Kim、S-M Choi 和 Y-G Kim 就职于韩国首尔世宗大学计算机工程系。H Jin 就职于中国合肥安徽大学计算机科学与技术学院。* 通讯作者：ykim@sejong.ac.kr。本文为作者接受的手稿。最终版本发表于《IEEE多媒体汇刊》，2025 年

仅依赖全局文本嵌入。事实上，尽管 CLIP 是文本到图像任务 [8], [14], [24], [15], 中语言条件模块的一种卓越方法，但即便经过大规模训练或提供了全局指令，它在理解文本时仍面临偏差问题，也可能忽略细粒度的子指令。此外，纯文本方法在生成模型中可能较为敏感，因为指令结构的变化会显著影响其性能。另一个挑战在于，确保指令中描述的面部运动与语言语义相符。生成面部运动的一种直接方法，是在 StyleClip [25] 的源代码与编辑后代码之间进行插值。然而，在生成面部动画的同时，还要控制潜在代码来调整源图像的头部姿态，这一额外挑战使得实现预期目标变得困难。

为了建立文本与面部表情之间的联系，我们将用于面部表情过渡的文本指令补充到两个包含面部图像的标准数据集中，即 CelebV-HQ（高质量名人视频）和 CK+ （扩展版 Cohn-Kanade）。为了详细控制面部模型，我们采用 FLAME（基于关节模型和表情学习的人脸），并使用基本面部表情及对应姿态对其进行微调，以捕捉头部、眼睛和嘴部的运动。 [26]

此外，整合不同模态的特征表示能够减少冗余资源带来的歧义，并通过改善模态间关系，有效处理模态缺失问题，尤其是在生成模型中 [27],[28]。所提出的方法使我们的模型——面部表情转换指令（I2FET）——能够增强面部表情特征与文本描述之间的关联，从而改善潜在表征，以准确生成目标面部表情，并减少对指令文本的单独依赖。随后，利用 I2FET 根据源图像的面部参数生成目标面部表情，并据此创建面部表情轨迹。借助这一框架，我们可以根据一张面部照片生成三维面部表情虚拟形象，并根据包含初始和最终面部表情的文本提示指令创建任意面部表情转换。

本研究的主要贡献如下：

- 我们提出了一种指令驱动的三维面部表情生成与转换框架，可从面部照片出发，根据指令生成面部表情过渡。
- 我们提出了一种指令驱动的面部表情分解器（IFED），旨在学习多模态数据，并捕捉面部表情特征与文本描述之间的相关性。
- 我们提出了一种指令到面部表情过渡方法（I2FET），该方法采用条件变分自编码器，并结合 IFED 与顶点重建损失函数，以增强潜向量的语义理解能力。
- 在 CK+ 和 CelebV-HQ 数据集上的广泛评估表明，该模型优于最先进方法。

II. 相关工作

A. 面部表情过渡

生成对抗网络（GANs）一直受到青睐，被用于生成面部表情。例如，Motion3DGAN [22], 作为 MotionGAN [29],的扩展，处理了 3D关键点的动态变化；而 WGAN 则学习跨越超球面空间的 3D表情动态分布，并通过条件采样生成关键点序列。在这种情况下，网格解码器利用生成的关键点序列，逐帧对中性的 3D人脸网格进行变形。不过，该方法的缺点是，必须提供起始和结束表情，才能生成串联的表情过渡。另一方面，扩散模型通过建模输入分布来实现面部表情生成[23] 。3D网格的变形由特定关键点引导，同时将输入面部网格的原始形状用于对每个顶点施加位移。在这项工作中，MotionClip [30],最初是为人体运动合成设计的，并被用于与根据文本给定表情过渡条件的生成任务进行比较。EMOTE [31] 利用情绪标签和各种输入，生成能够保持语音唇形同步以及面部表情的说话头部虚拟形象。具体而言，被称为 FLINT（FLAME IN TIME）的面部运动先验使用 FLAME[26]的表情和姿态参数来表示面部运动序列。

尽管已有多项研究展示了在不同条件下生成面部表情过渡的方法，但生成的人脸通常以正面人脸为条件，忽略了头部姿态变化。受以往研究的启发 [22], [23], [31],，本研究提出了一种面部表情过渡模型，该模型能够学习表情和姿态参数。该模型的另一项优势是可以轻松集成到生成人脸外观的模型中。

B. 人脸渲染

在人脸渲染领域，3D可变形模型（3DMM）已广泛用于渲染逼真的人脸 [17], [32],[9], [11]。在这些方法中，VariTex [17]采用人脸纹理解码器获取神经纹理，再利用 UV 贴图进行采样。随后，将该神经人脸特征图像与加性特征图像合并，作为 U-Net 的输入，以生成所需图像。Exp-GAN [32] 采用神经体积生成器和 StyleGAN2 来生成不同分辨率的图像。此过程结合了不同的特征分量，例如人脸特征、深度图像和特征体积。人脸特征同样通过使用 UV贴图对神经纹理进行采样获得。然而，这项工作计算量大且耗时。ROME [9] 通过从输入图像估计特定的头部网格和神经纹理来合成逼真的人脸图像，随后采用神经渲染技术来渲染绑定网格。接下来，类似 U-Net 的神经渲染网络（

接受神经纹理和表面法线等输入，用于获取目标图像及其对应的掩码。类似地，CVTHead [11] 也会从输入图像估计神经纹理，但在这种情况下，神经纹理会先展平为特征向量，再与顶点令牌一同输入 Transformer，以输出顶点描述符。具体而言，每个顶点都被视为一个查询令牌，并使用 Transformer 从源图像中学习规范顶点特征。为了生成所需图像及其掩码，每个顶点的特征描述符和深度会被投影到图像空间，然后由 U-Net 进行处理。

在本研究中，我们利用 ROME 和 CVTHead 生成面部外观，并借助 DECA 提取面部表情系数。这简化了将其集成到面部表情转换模型中的过程，并增强了所提出框架在实际场景中的适用性。

III. 方法

A. 问题陈述

我们的目标是开发一个模型，利用两个输入生成 3D 面部表情：一张人脸图像 I_s 和一条文本指令 t。面部表情转换模型 $\mathcal{T}_w(t, I_s)$具有权重 w，使用训练集 $\mathcal{X}$ ={$(t, (e_0, \theta_0), (e_1, \theta_1))^{(j)}\}_{j=1}^{n}$进行训练，其中 $e_0, e_1 \in R^{1\times 50}$ 表示 50 维表情向量，$\theta_0, \theta_1 \in R^{1\times 6}$表示 6 维姿态向量，分别对应指令 t所指定的面部表情序列中的第一个和下一个表情。 n 是训练数据集中的样本数。我们的网络学习 $\mathcal{T}_w$ (.)，以近似数据分布 $p(\mathcal{X})$，随后根据给定的指令 t生成面部表情系数序列 $\{(e_0, \theta_0), (e_1, \theta_1)\}$。接下来，使用人脸渲染模块$\mathcal{G}(I_s, \mathcal{S})$渲染一系列面部外观 $\mathcal{Y} = \{y_s^{(1)}, y^{(i)}, y_{e_0}^{(k)}, y^{(j)}, y_{e_1}^{(T)}\}$；其输入包括源图像 I_s的纹理和面部表情轨迹 $S = \{s_s^{(1)}, s^{(i)}, s_{e_0}^{(k)}, s^{(j)}, s^{(T)}{}_{e_1}\}$，其中 $\{s_s^{(1)}, s_{e_0}^{(k)}, s_{e_1}^{(T)}\}$是与源图像对应的锚定面部表情序列，以及由 $\mathcal{T}_w$(.)生成的特定面部表情。然后，可以通过线性插值函数推断锚定面部表情之间转换的面部表情序列 $s^{(i)}$ 和 $s^{(j)}$ ，以确保面部表情序列的平滑性。

B. 所提方法

图2概述了所提出的框架，该框架可根据输入指令 t 控制单目图像 I_s 中的三维面部表情。该框架由两个主要模块组成：(1) 面部表情转换（FET）， $\mathcal{T}_w(t, I_s)$，用于生成多样的表情轨迹；(2) 人脸渲染（FR）， $\mathcal{G}(I_s, S)$，用于基于源图像I_s 的纹理以及 FET 模块生成的表情轨迹 S 来渲染面部外观。

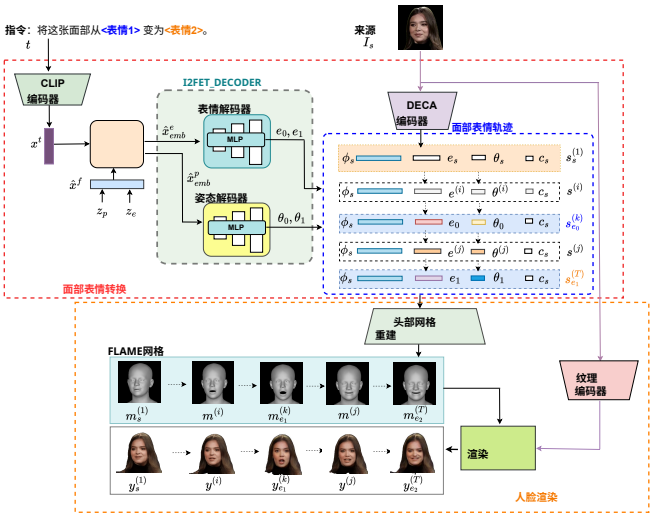


图 2：所提出的框架概览，该框架由两个主要模块组成：面部表情过渡（FET）和人脸渲染（FR）。首先，预训练CLIP编码器为FET模块生成文本向量 x_t。与此同时，从 $\mathcal{N}(0, I)$中提取潜在表示 z_p和 z_e。随后，对潜向量进行处理并拼接，得到$\hat{x}^f$。之后，IFED模块将 x^t和$\hat{x}^f$作为输入，为姿态解码器和表情解码器生成条件特征向量$\hat{x}^e_{emb}$和$\hat{x}^p_{emb}$。接着，为了使面部表情序列平滑，对源图像中的面部表情（e_s， θ_s）进行线性插值，并结合由I2FET解码器生成的特定面部表情（e_0， θ_0），（e_1，θ_1）。随后，将这些序列与通过DECA从源图像中获取的形状 ϕ_s 和相机 c_s 参数结合，形成面部表情轨迹 $\{s_s^{(1)}, s^{(i)}, s_{e_0}^{(k)}, s^{(j)}, s_{e_1}^{(T)}\}$。对于 FR，头部网格重建会生成与面部表情轨迹对齐的FLAME网格序列 $\{m_s^{(1)}, m^{(i)}, m_{e_0}^{(k)}, m^{(j)}, m_{e_1}^{(T)}\}$，而纹理编码器则从源图像中提取面部外观。最后，渲染模块利用FLAME网格序列和提取的面部外观，生成面部表情轨迹 $\{y_s^{(1)}, y^{(i)}, y_{e_0}^{(k)}, y^{(j)}, y_{e_1}^{(T)}\}$ 中的面部外观。

1) 面部表情转换： 该模块旨在生成多样的表情轨迹，这些轨迹对应于源人脸图像 I_s 的面部参数以及指令 t中描述的特定面部表情参数。该模块包含两个主要分支：第一个分支接收指令，并根据指令 t生成相应的面部表情序列；第二个分支则接收二维人脸图像，对其面部系数进行编码，并创建面部表情轨迹。

在前者中，预训练CLIP编码器 [33] 根据给定的指令 t 提取文本描述 $x^t \in R^{77\times 768}$ 。然后，I2FET模型生成一系列用于面部表情轨迹的表情和姿态系数。

指令驱动的面部表情分解器：采用双分支视觉 Transformer [34] 提取多尺度特征，并在不同尺度上融合交叉注意力信息，因为这种方法在图像分类任务中效率较高。在此，我们使用面部表情参数与文本交叉注意力（CAFT），通过增强对多模态数据的理解并细化潜在表征，学习指令与面部表情参数之间的关系，从而服务于指令到面部表情转换模型。所提出的 IFED 模块如图3所示。具体而言，我们直接将面部表情参数和文本描述输入 CAFT，而不是像 中那样添加类别标记。

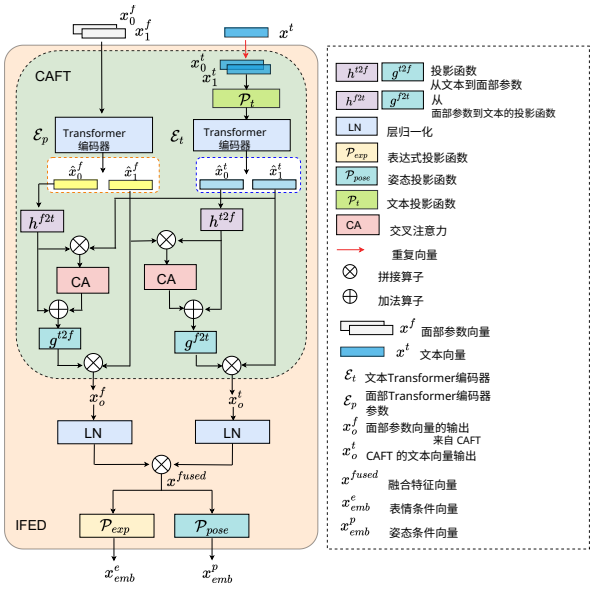


图3：IFED（指令驱动的面部表情分解器）的流程图。它由 CAFT 和参数分解组成。CAFT 的主要功能是在面部参数与文本指令之间引入交叉注意力。CAFT 的两个输出经过层归一化后进行组合，以生成融合特征向量 x^{fused}。随后，姿态投影函数 $\mathcal{P}_p$ 和表达式投影函数 $\mathcal{P}_e$ 用于将 x^{fused} 分解为两个用于面部表情（x^e_{emb}）和姿态（x^p_{emb}）的条件向量。

对于 Transformer 编码器，第一分支学习面部表情参数之间的相关性，而第二分支则侧重于学习文本描述。首先，每个表示面部表情参数的向量由两个元素构成：表情和姿态参数，记为 $x^f = \{e, \theta\} \in R^{m\times 56}$，其中 m 表示给定指令中的表情数量。由于 [35], [31], 我们仅使用下颌姿态 θ^{jaw} 和表情 e 来传达面部表情信息。因此，我们将 $x^f = \{e, \theta^{jaw}\} \in R^{m\times 53}$改写为第一分支的输入。随后，使用 Transformer 编码器作为

$$\begin{aligned} y^f &= x^f + MSA(LN(x^f)) \\ \hat{x}^f &= y^f + FFN(LN(y^f)) \end{aligned} \tag{1}$$

其中，$\hat{}x^f$ 表示面部表情参数分支 $\mathcal{E}_P$中 Transformer 编码器的输出。$LN(.)$、$MSA(.)$和 $FFN(.)$分别表示层归一化、多头自注意力和前馈网络。同样，在文本分支中，我们复制 x^t 以创建向量 $\{x\}^t_{i=\{0,1\}}$，用于计算。随后，对 $\{x\}^t_{i=\{0,1\}}$应用文本投影函数$P_t(.)$，生成 Transformer 编码器的输入。由此，通过 Transformer 编码器 $\mathcal{E}_t$得到文本描述$\hat{}x^t$ 。然后，通过交叉注意力融合两个分支，以捕捉面部表情参数与文本描述之间的关系之间的关系。融合过程可表示为

$$\begin{aligned} y^f_c = h^{f2t}(\hat{x}^f_0), \quad y^t_c = h^{t2f}(\hat{x}^t_0), \\ x^f_o = g^{t2f}([CA(y^f_c \otimes x^t_1) + y^f_c]) \otimes \hat{x}^f_1 \\ x^t_o = g^{f2t}([CA(y^t_c \otimes x^f_1) + y^t_c]) \otimes \hat{x}^t_1 \end{aligned} \tag{2}$$

其中，$h^{t2f}, h^{f2t}, g^{t2f}$ 和 g^{f2t} 是表示文本和面部表情参数的向量之间的投影函数和反投影函数，$x^f_o \in R^{m\times 53}$ 和 $x^t_o \in R^{m\times 768}$ 是 CAFT 的输出，CA 表示交叉注意力，$\otimes$ 表示拼接算子。接下来，融合特征向量通过将 x^f_o 和 x^t_o进行拼接获得，并经过 $LN(.)$，如下所示：

$$x^{fused} = LN(x^t_o) \otimes LN(x^f_o) \tag{3}$$

最后，对姿态 $\mathcal{P}_p(。)$ 和表情$\mathcal{P}_e(。)$的投影函数应用于融合特征向量，具体如下：

$$\begin{aligned} x^e_{emb} = \mathcal{P}_e(x^{fused}) \\ x^p_{emb} = \mathcal{P}_p(x^{fused}) \end{aligned} \tag{4}$$

其中，$x^e_{emb} \in R^{m\times 50}$ 和 $x^p_{emb} \in R^{m\times 6}$ 分别表示姿态和表情的条件特征向量。在本文中，$\mathcal{P}_p(。)$ 和 $\mathcal{P}_e(。)$被设计为线性变换函数，用于将特征 x^{fused} 映射到相应的姿态和表情特征向量。

指令到面部表情转换：所提出的 I2FET 方法基于条件变分自动编码器架构。该方法包括编码器 $\mathcal{E}^t_e(。)$ 和 $\mathcal{E}^t_p(。)$，以及解码器 $\mathcal{D}^t_e(。)$ 和 $\mathcal{D}^t_p(。)$，这些组件均使用三个多层感知机（MLP）进行定义。这些组件以分别表示表情和姿态信息的特征向量为条件。图4展示了所提出的 I2FET 架构概览。

这里，编码器定义如下：

$$\mathcal{E}^t_e(e \otimes x^e_{emb}) \longrightarrow (\mu_e, \sigma_e) \tag{5}$$

and

$$\mathcal{E}^t_p(\theta \otimes x^p_{emb}) \longrightarrow (\mu_p, \sigma_p) \tag{6}$$

其中，$\otimes$ 表示拼接算子，x^e_{emb} 和x^p_{emb} 分别表示表情和姿态的条件特征向量。对于编码器阶段，x^e_{emb} 和 x^p_{emb} 由文本描述 x_t 和面部表情参数（e，θ）生成。μ_e 和 μ_p 表示表情分布的均值和标准差，而 σ_e和 σ_p 表示姿态分布的均值和标准差。在本研究中，解码器中的条件特征向量 $\hat{}x^e_{emb}$ 和 $\hat{}x^p_{emb}$ 由文本描述 x_t 计算得到，而表示姿态和表情的潜向量 $\hat{}z_p$和 $\hat{}z_e$ 分别表示姿态和表情。这些潜向量的计算如下：

$$\begin{aligned} \tilde{z}_p = \sigma_p * z_p + \mu_p \\ \hat{z}_p = \mathcal{T}_p(\tilde{z}_p) \\ \tilde{z}_e = \sigma_e * z_e + \mu_e \\ \hat{z}_e = \mathcal{T}_e(\tilde{z}_e) \end{aligned} \tag{7}$$

其中，z_p和 z_e是从 $\mathcal{N}(0, I)$中采样得到的潜向量。在这种情况下，我们利用线性变换函数 $\mathcal{T}_p(。)$

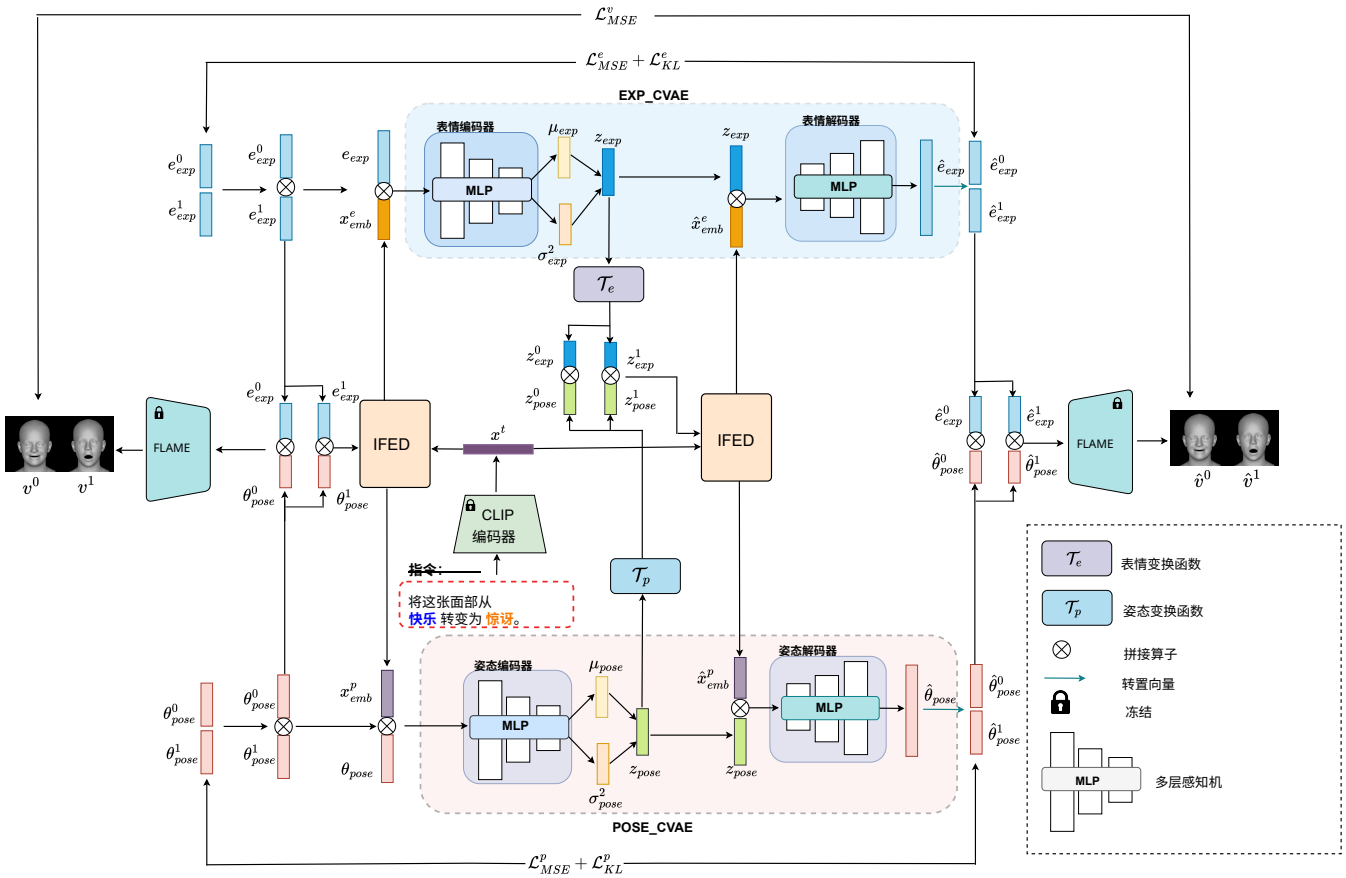


图4：I2FET（指令到面部表情转换）方法的详细架构。给定一条指令和一系列面部参数（包括姿态和表情），IFED模块利用这些面部参数以及由CLIP提取的文本描述，分别生成对应的姿态 x^p_{emb} 和表情 x^e_{emb}条件特征向量。随后，姿态和表情编码器将这一系列面部参数及条件特征向量映射为对应于姿态和表情的潜在表征。这些潜向量与文本描述一同输入IFED模块，以生成条件特征向量$\hat{x}^e_{emb}$ 和$\hat{x}^p_{emb}$。之后，表情和姿态解码器分别利用这些潜在表征以及条件特征向量$\hat{x}^e_{emb}$ 和$\hat{x}^p_{emb}$，重建表情$\hat{e}$ 和姿态$\hat{\theta}$ 参数。最后，预训练的FLAME解码器基于$\hat{e}$ 和$\hat{\theta}$重建顶点坐标$\hat{v}$ 。

以及 $\mathcal{T}_e$(。)分别将 $\tilde{z}_p$ 和 $\tilde{z}_e \in R^{m\times 16}$ 转换为$\hat{z}_p \in R^{m\times 6}$和$\hat{z}_e \in R^{m\times 50}$，这与表情和姿态维度类似。因此，潜向量 $\tilde{z}_e$ 和 $\tilde{z}_p$ 可用作IFED模块的输入。随后，将潜向量$\hat{z}_e$ 和$\hat{z}_p$ 拼接在一起，并与文本向量 x^t一同输入IFED模块，以获得条件特征向量$\hat{x}^e_{emb}$ 和$\hat{x}^p_{emb}$ ，其形式为

$$\hat{x}^e_{emb}, \hat{x}^p_{emb} = IFED(\tilde{z}_p \otimes \tilde{z}_e, x^t) \tag{8}$$

该方法通过根据训练过程中学习到的潜向量优化解码器的条件特征向量，增强潜向量的语义表示。解码器定义如下

$$\mathcal{D}^t_p(\tilde{z}_p \otimes \hat{x}^p_{emb}) \longrightarrow (\hat{\theta}_0, \hat{\theta}_1) \tag{9}$$

and

$$\mathcal{D}^t_e(\tilde{z}_e \otimes \hat{x}^e_{emb}) \longrightarrow (\hat{e}_0, \hat{e}_1) \tag{10}$$

通过最小化真实系数 e, θ 与预测系数 $\hat{e}, \theta$ 之间的重建损失函数 $\mathcal{L}_{MSE}(.,.)$，对 I2FET模型的参数进行优化。此外，Kullback-Leibler散度损失函数用于最小化真实值的概率分布与预测值概率分布之间的距离。

$$\mathcal{L}_e = \mathcal{L}_{MSE}(e, \hat{e}) + 0.5\left[-\sum_i(log\ \sigma_i^2 + 1) + \sum_i \sigma_i^2 + \sum_i \mu_i^2\right] \tag{11}$$

$$\mathcal{L}_p = \mathcal{L}_{MSE}(\theta, \hat{\theta}) + 0.5\left[-\sum_i(log\ \sigma_i^2 + 1) + \sum_i \sigma_i^2 + \sum_i \mu_i^2\right] \tag{12}$$

其中， $\mathcal{L}_e$ 和 $\mathcal{L}_p$ 分别表示表情损失函数和姿态损失函数。

此外，我们使用预训练的 FLAME [26]，三维头部参数化模型，该模型集成于 DECA[36]，中，包含 $N = 5023$ 个基础顶点 $V_b \in \mathcal{R}^{N\times 3}$，以及两组 M 和 K 基向量，分别用于编码形状混合形状$\mathcal{S} \in R^{N\times 3\times M}$和表情混合形状 $B \in R^{N\times 3\times K}$。首先对基向量进行混合，然后应用线性混合蒙皮（LBS）$\mathcal{W}$（.），按照姿态旋转顶点

θ。最终的世界坐标系下的重建结果可按如下方式计算

$$\mathcal{V}(\phi, e, \theta) = \mathcal{W}(V_b + \mathcal{S}\phi + \mathcal{B}e, \theta) \tag{13}$$

其中， ϕ、 e和 θ 分别表示身份形状、面部表情和头部姿态参数。在这种情况下，使用 FLAME 重建顶点，以提升 I2FET 的性能。

$$\mathcal{L}_v = \mathcal{L}_{MSE}(v, \hat{v}) \tag{14}$$

其中，顶点坐标 v 和 $\hat{v}$ 是将真实值 $\{\theta, e, \phi\}$ 和重建结果 $\{\theta, \hat{e}, \phi\}$ 输入 FLAME 后计算得到的。

最后，总损失函数表示为

$$\mathcal{L}_{total} = \mathcal{L}_e + \mathcal{L}_p + \mathcal{L}_v \tag{15}$$

对于第二分支，使用预训练的 DECA [36] 对源人脸 I_s 的面部系数进行编码，包括形状 $\phi_s \in R^{1\times100}$、表情 $e_s \in R^{1\times50}$、姿态 $\theta_s \in R^{1\times6}$ 和相机 $c_s \in R^{1\times3}$。完成这一步后，将 $\{\phi_s, e_s, \theta_s, c_s\}$ 与第一分支生成的面部表情系数序列 $\{(e_0, \theta_0), (e_1, \theta_1)\}$ 结合，以生成锚定面部表情 $\{s_s^{(1)}, s_{e_0}^{(k)}, s_{e_1}^{(T)}\}$。然后，应用线性插值函数 $\Psi(.)$，使两个锚定面部表情之间的过渡更加平滑。用于在面部表情序列 $s^{(l)}$ 和 $s^{(n)}$ 之间进行插值的 $\Psi(s^{(l)}, s^{(n)})$ 函数可定义为

$$\begin{aligned} e^{(k)} &= \delta * e^{(l)} + (1-\delta) * e^{(n)} \\ \theta^{(k)} &= \delta * \theta^{(l)} + (1-\delta) * \theta^{(n)} \end{aligned} \tag{16}$$

其中， $0 \le \delta \le 1$ 是一个线性系数， $\{e^{(l)}, \theta^{(l)}\} \in s^{(l)}$。这使得我们的框架能够在运动变化均匀采样时确保时间一致性。

2) 人脸渲染：该模块为由 FET 生成的面部表情轨迹渲染面部外观。在本文中，人脸渲染方法基于 FLAME 的面部参数，以便与 FET 集成。

头部网格重建：

它以面部表情转换序列作为输入，使用公式 13 生成与面部表情相对应的顶点坐标。随后，将这些表情与头发和肩部区域的变形 $f_H(。)$ 结合，得到目标顶点坐标 I_s，如下所示

$$\mathcal{M}^t = \mathcal{V}(\phi, e, \theta) + f_H(I_s) \tag{17}$$

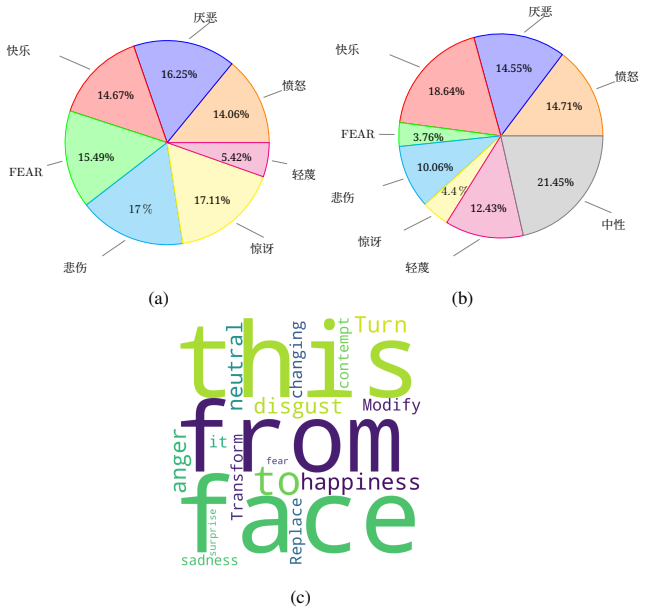
其中， $f_H(。)$ 是预训练线性变形模型 [9]，用于细化顶点位置。

渲染：

在头部网格重建步骤之后，纹理解码器 $\mathcal{E}_{tex}(I_s)$可以根据重建的头部顶点$\mathcal{M} = \{m_s^{(1)}, m^{(i)}, m_{e_0}^{(k)}, m^{(j)}, m_{e_1}^{(T)}\}$ 以及从源图像中提取的中性纹理 I_s 渲染面部外观。本研究探索了一种适用于

面部表情转换。因此，我们采用预训练的人脸渲染模型，也就是已知的最先进模型 [9]，

[11]，该模型能够根据使用 FLAME 计算得出的人脸参数渲染面部外观。随后，面部外观序列 $\mathcal{Y} = \{y_s^{(1)}, y^{(i)}, y_{e_0}^{(k)}, y^{(j)}, y_{e_1}^{(T)}\}$将作为所提出的框架的输出生成。



(a) (b) (c)

图5：本研究分别在 CK+ (a) 和 CelebV-HQ (b) 数据集以及从文本指令中提取的采样词汇集合 (c) 中所使用的面部表情统计。

IV. 实验

A. 数据集

表 I： N_1， N_2 $\in$ { 快乐、厌恶、愤怒、恐惧、惊讶、轻蔑、悲伤 } + N_1 N_2 $\in$ 在 CK 数据集和，{ 快乐、厌恶、愤怒、恐惧、惊讶、轻蔑、悲伤、中性} 在 CelebV-HQ 数据集中。

No	指令/提示词
1	将此面部从 [N_1] 变为 [N_2]。
2	将此面部从 [N_1]变为 [N_2]。
3	将此面部从 [N_1] 变为 [N_2]。
4	修改此面部，将其从 [N_1]变为至 [N_2]。
5	将此面部从 [N_1 替换为 [N_2]。

1) CK+ 数据集：CK+ 数据集补充了用于面部表情生成与转换的文本指令。原始 CK+ 数据集 [37]包含 117 名受试者和 7 种情绪，例如快乐、厌恶、愤怒、恐惧、惊讶、轻蔑和悲伤。这些指令用于描述这些面部表情之间的转换。为了提取图像中与面部表情对应的面部参数和形状图像，我们使用了预训练的 DECA 模型。对于每个样本，我们创建一个由指令、目标形状图像以及同一受试者对应的面部参数组成的元组。本研究中的 CK+ 数据集共包含 26,352 个样本，其七种面部表情的组成情况如图 5a 所示。

2) CelebV-HQ 数据集：与 CK+ 的情况类似，我们为 CelebV-HQ 数据集添加了文本指令，用于实验 [38]。原始 CelebV-HQ 数据集包含 15,653 名受试者和 8 种表情：中性、快乐、厌恶、愤怒、恐惧、惊讶、轻蔑和悲伤。CelebV-HQ 数据集中的图像是自然采集的，存在诸多挑战，例如头部姿态大幅偏离正面、模糊和亮度问题。我们使用头部姿态算法 [39] 估计头部姿态的偏航角、翻滚角和俯仰角，范围为 -15 至 15 度，从而解决了这一问题。鉴于受试者是随机选取的，我们筛选出拥有两种以上面部表情的受试者。随后，使用预训练的 DECA 提取对应的形状图像和面部参数。本研究采用的 CelebV-HQ 数据集包含 614 名受试者和 28,335 个样本，图5b 展示了八种面部表情的统计分布。

表I列出了用于生成面部表情转换引导指令的模板句。我们为每个样本生成了五条指令，并描述了从表情 N_1 到表情 N_2的过渡。随后，从这些指令中随机选择一条呈现句。图5c展示了文本指令中使用的一些采样词语。

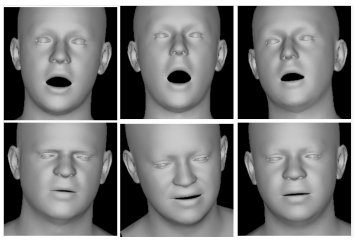


图6：使用CK+ 数据集中的图像生成人脸表情的形状。上排和下排分别展示了具有不同头部姿态的惊讶和愤怒表情。

B. 实现

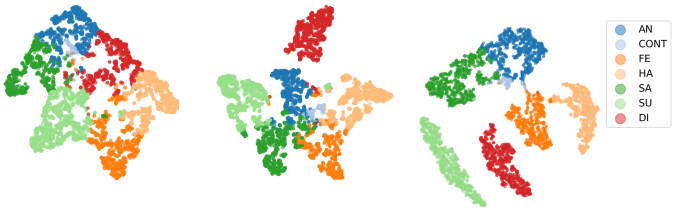
I2FET模型训练了200轮，批量大小为128，学习率为 $8e^{-4}$。采用Adam优化器 [40] 对模型进行优化。在实验中，数据集的10%被分配为测试集，其余部分用于训练。随后，从训练集中抽取10%作为验证集。本研究中，每项实验均重复十次，并使用平均结果评估所提出的I2FET的整体性能。由于我们的任务是新的，且目前没有可用于公平比较的特定方法，因此我们选择MotionClip [30]作为基线。对于人脸渲染实验，采用了 [9], [11] 的配置。

C. 面部表情分类

在本研究中，采用适当的分类器来评估用于生成面部表情的方法十分重要，正如先前研究 [22], [23]所示。鉴于CK+ 和CelebV-HQ数据集存在偏斜分布，在这些不平衡数据集上训练的面部表情分类器可能会影响少数类的性能，从而使结果偏向多数类。此外，分类模型的有效性也可能受到头部运动的影响，如图6所示，这种情况十分常见。

为解决这些问题，我们进行了一项实验，以确定一种合适的分类器，使其不仅能够处理不平衡数据集，还能应对旋转不变变换。我们采用集成了重加权焦点损失（RFL）的 ResNet-101 [41], MEK [42], 和 ResNet-101 来评估面部表情分类的性能。为此，我们通过将原始 CK+ 和 CelebV-HQ 数据集分别旋转 $\pm 10^o, \pm 15^o$ 和 $\pm 30^o$ 度，构建了增强数据集。随后，使用原始数据集训练这三个网络，并在所有数据集上进行评估。我们采用准确率评估所有类别中的样本是否被正确分类，并采用 G-mean 评估模型有效识别面部表情分类模型所预测的少数类的能力。第 IV-E 节消融研究中的补充数据表明，MEK 能够处理不平衡数据集，但对旋转变换较为敏感。ResNet101 在旋转变换下表现良好，但在高度不平衡的数据集上表现较差。集成了 RFL 的 ResNet101 在这两种情况下均优于 ResNet101 和 MEK。因此，我们选择采用集成 RFL 的 ResNet101 作为分类器，以评估我们模型生成面部表情的能力。

D. 结果



(a) (b) (c)

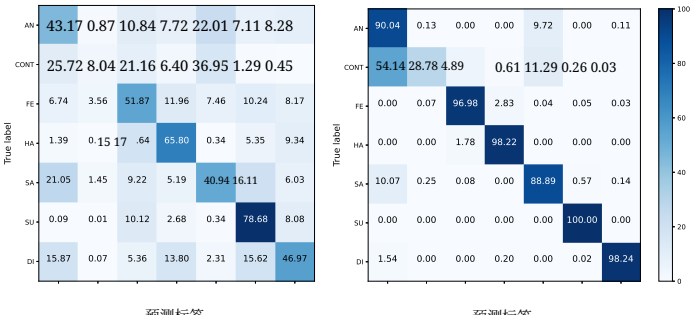
图7：分别使用 t-SNE 可视化 (a) MotionClip、(b) 不含 IFED 的 I2FET 和 (c) 含 IFED 的 I2FET 所学习到的七种面部表情的潜在空间。

表II：我们的方法与 MotionClip 在 THECK+ 数据集上的准确率比较

方法	准确率$_1$ (%) ↑	准确率$_2$ (%) ↑	几何均值 (%) ↑
真实值	99.69 ± 0.0001	99.39 ± 0.0002	99.69 ± 9.3e^{-5}
MotionClip	52.00 ± 0.0058	20.00 ± 0.0082	40.48 ± 0.0060
Ours	**91.44 ± 0.0024**	**84.03 ± 0.0034**	**80.30 ± 0.0064**

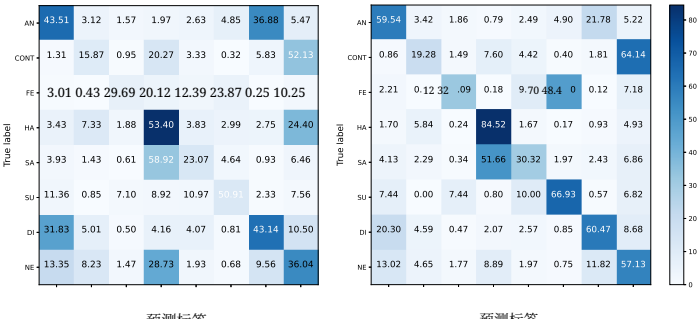
表 III：CelebV-HQ 数据集上我们的方法与 MotionClip 的准确率比较

方法	准确率$_1$ (%) ↑	准确率$_2$ (%) ↑	几何均值 (%) ↑
真实值	93.67 ± 0.0000	87.89 ± 0.0000	95.04 ± 1.1e^{-16}
MotionClip	40.00 ± 0.0039	13.73 ± 0.0056	34.42 ± 0.0069
Ours	**58.24 ± 0.0028**	**33.45 ± 0.0052**	**46.47 ± 0.0065**



(a) (b)

图8：MotionClip（a）和我们的方法（b）在 CK+ 上对七种面部表情的混淆矩阵。



(a) (b)

图9：MotionClip（a）和我们的方法（b）在 CelebV-HQ 上对八种面部表情的混淆矩阵。

本研究旨在根据面部的二维图像生成具有三维面部表情的虚拟化身，并根据给定的指令转换面部情绪，因此面临两大挑战：（1）根据指令提供的表情信息实现面部表情过渡；（2）渲染面部外观，使其符合期望的面部表情。我们设计了两项实验，从定量和定性两个方面评估所提出的框架所取得的结果。

1）定量比较：我们设计了两项任务进行定量比较。

面部表情转换
DECA编码器从源图像中提取形状、面部表情和姿态。同时，结合 IFED 和 I2FET 的 CLIP编码器用于解读文本指令：初始面部表情、最终面部表情以及姿态变化。为了将我们的方法与 MotionClip 进行比较，我们使用第一个准确率指标（Acc_1）统计指令中所有表情标签的正确分类样本数。另一个指标（Acc_2）用于衡量与每条输入指令中呈现的两个表情标签均对应的正确分类样本数。结果表明，如表II所示，在两个数据集上，我们模型的准确率均高于 MotionClip。例如，在 CK+数据集上，我们的模型在生成独立表情和生成表情序列方面的准确率分别达到 91.44% 和 84.03%。我们模型的几何均值为 80.30%。

表IV：我们的方法与 ROME 或 CVTHead 结合后，分别在 CK+ 和 CelebV-HQ 数据集上进行纹理生成所获得的结果。请注意，↓ 表示数值越低越好，而 ↑ 表示数值越高越好。

数据集	方法	L_1 ↓	峰值信噪比↑	LPIPS↓	多尺度结构相似性↑
CK+	我们的方法+ROME	0.199	9.939	0.490	0.347
	我们的方法+CVTHead	**0.005**	**33.74**	**0.021**	**0.978**
CelebV-HQ	我们的方法+ROME	0.168	11.316	0.505	0.418
	我们的方法+CVTHead	**0.003**	**37.08**	**0.010**	**0.990**

对于 CelebV-HQ 数据集（真实世界数据集），如表 III 所示，我们模型的 Acc_1、Acc_2，和几何均值分别为 58.24%、33.45% 和 46.47%。这一结果表明，在学习面部表情参数方面，我们的模型优于 MotionClip。

使用 *t-SNE* 进行可视化：图7展示了针对七种面部表情学习到的三种潜在表示，即使用 MotionClip、I2FET，以及使用带 IFED 的 I2FET 所获得的表示。这些可视化结果表明，与单独使用 MotionClip 或 I2FET 相比，将 IFED 与 I2FET 结合后，不同面部表情之间的分离更加明显。

混淆矩阵分析：图8和图9显示，在两个数据集上，与 MotionClip 相比，IFED 提高了大多数类别的准确率。然而，某些表情类别之间仍出现了少量错误分类，例如愤怒与悲伤之间，或轻蔑与愤怒、悲伤之间。在 CelebV-HQ 等真实世界数据集中，类别之间也会出现一些错误分类，尤其是在恐惧与惊讶，或轻蔑与中性等具有挑战性的表情之间。

人脸渲染：在人脸渲染任务中，FET 与最先进的模型 [9], [11] 结合，用于渲染与指令所请求的特定表情相对应的面部外观。具体而言，通过将 ROME [9]或 C VTHead [11] 与面部表情转换相结合来实现。在此过程中，DECA 提取面部参数，以渲染特定面部的几何信息。评估性能时不使用形状，而是采用 RGB图像和指令。使用 L_1 损失、峰值信噪比（PSNR）、学习感知图像块相似度（LPIPS）[44], 和多尺度结构相似性（MS-SSIM）来衡量生成表情中目标图像与合成图像之间的重建质量。如表IV所示，我们的 CVTHead 模型取得了最佳结果。

2) 定性比较：同样地，我们设置了两项任务进行定性比较。

面部表情转换：图10比较了 MotionClip 和我们的方法分别在 CK+ 和 CelebV-HQ 上生成的面部表情序列。这些结果清楚地表明，与 MotionClip 相比，我们的方法能够生成表情表达更准确的面部表情。

人脸渲染：可以通过观察并判断哪种表情看起来更好、更自然，对生成的表情进行定性比较。图11展示了基于在两个数据集上学习面部参数的潜在空间所生成的面部表情。观察者可能会注意到，CK+ 数据集中 (b) 和 (c) 之间存在差异，而

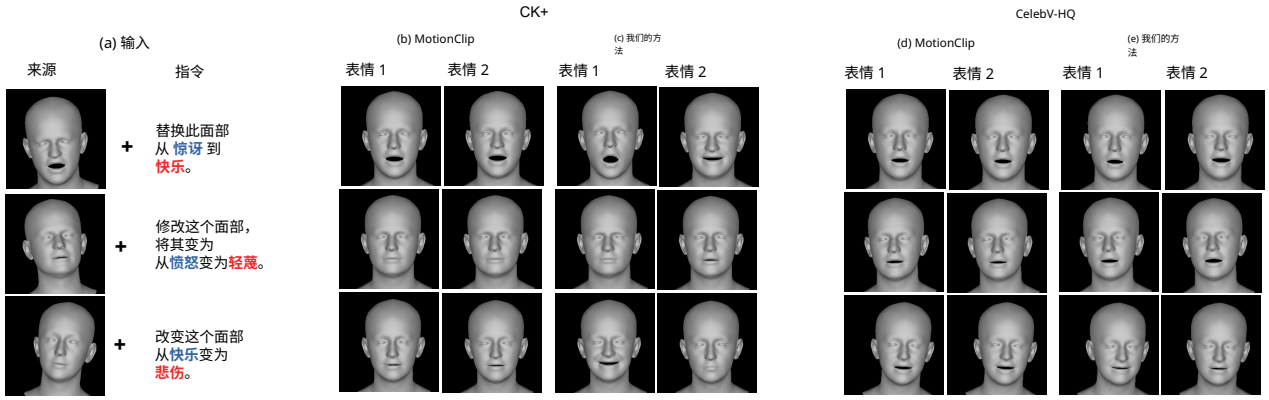


图10：MotionClip 和我们的方法分别生成的 3D 表情头部的定性比较。输入由形状图像和指令组成，用于通过 MotionClip（b）或我们的方法（c）在 CK+ 和 CelebV-HQ（d、e）上生成表情 1 和 2。

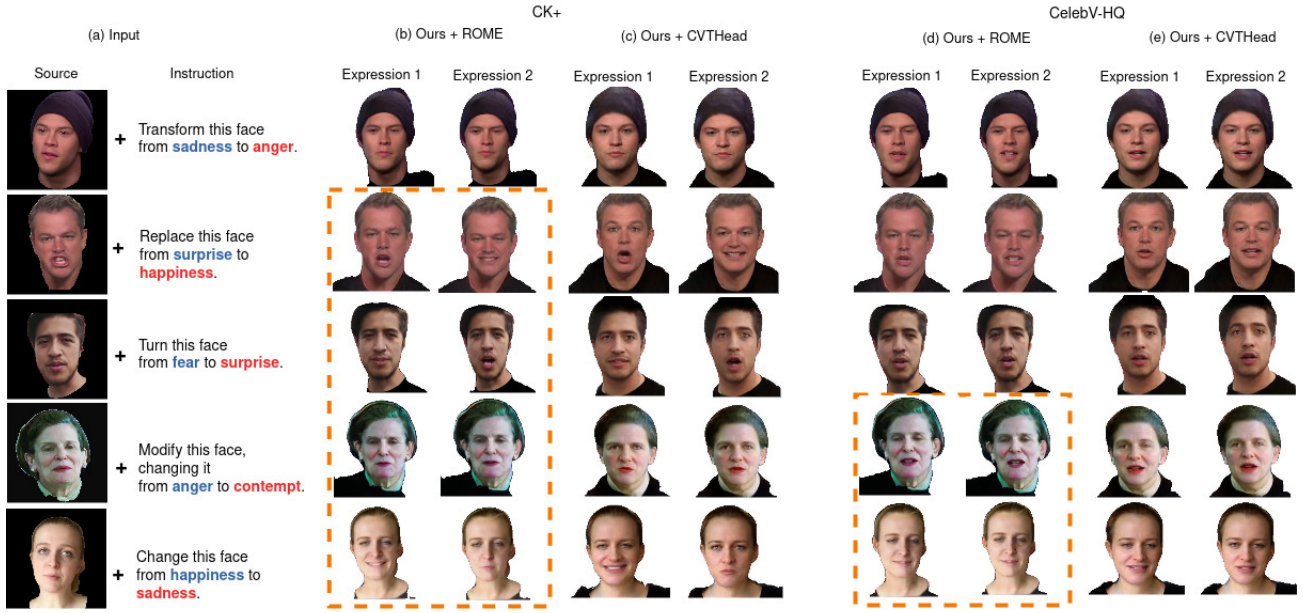


图11：将我们的方法与纹理生成 [9], [11]结合以生成所需表情时，生成的 3D 表情面部的定性比较。输入（a）由源图像和指令组成，用于通过我们的方法+ROME（b）或我们的方法+CVTHead（c）在 CK+ 和 CelebV-HQ（d、e）上生成表情 1 和 2。

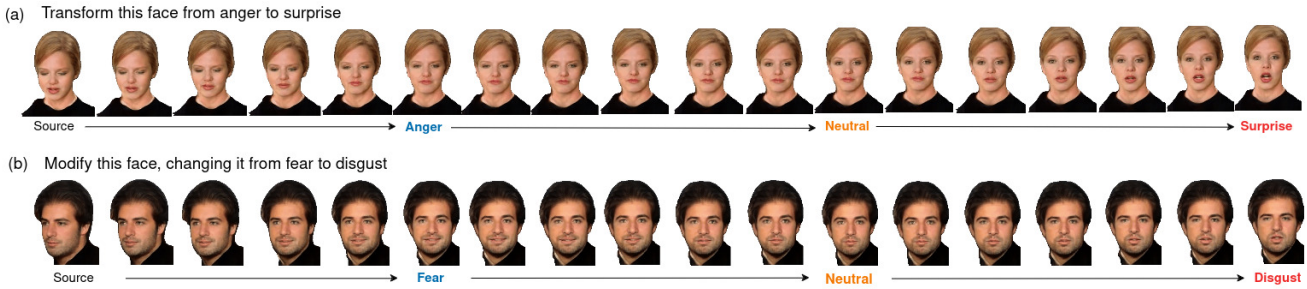


图12：生成过程的连续快照　　生成的 3D 表情面部，其中两个由文本提示指定的面部表情之间包含一个中性表情 .

对于 CelebV-HQ 数据集，则是 (d) 和 (e) 之间的差异，体现在面部表情和面部纹理上。例如，第二行中从惊讶到快乐的面部表情转换在 (c) 中比在 (b) 中更加清晰；第四行中的面部纹理在 (c) 中也比在 (b) 中平滑得多。(d) 和 (e) 之间也能观察到类似现象，这表明

我们的方法+CVTHead 优于我们的方法+ROME。例如，在图11中虚线框圈出的某些案例中，当源图像的面部区域被遮挡，或指令描述了伴随显著姿态变化的表情（例如从惊讶转变为快乐，或从恐惧转变为惊讶）时，会出现轻微失真。这导致

与我们的方法+CVTHead 相比，生成的面部区域存在不一致和细节缺失。对两个数据集进行比较（例如比较 (b) 与 (d)，或 (c) 与 (e)）可以发现，CK+ 的视觉质量优于 CelebV-HQ。这些结果证实，我们的模型能够出色地将基于 CK+ 或 CelebV-HQ 学习到的面部表情转换到特定面部图像中。

此外，我们邀请 24 名受试者参与了一项用户研究，请他们观看 60 组随机配对的图像。图像中的身份各不相同，选自 CVTHead +MotionClip 或 CVTHead+ 我们的方法。受试者需要选择最接近指令中指定表情的图像，或指出其中没有任何图像符合所提供的表情。当然，他们并不知道生成所展示图像时使用的是哪种算法。每段视频由四名受试者观看，他们投票选出更接近指令中表情的视频。如果票数持平，则会请另一名受试者作出最终决定。根据这些结果，在 CK+ 上，CVTHead + 我们的方法的准确率为 73%，而 CVTHead +MotionClip 的准确率为 58%。在 CelebV-HQ 上，CVTHead +我们的方法和 CVTHead +MotionClip 的准确率分别为 77% 和 70.5%。

3）推理时间：推理过程中，CVTHead + MotionClip 生成一个视频需要 3.97 秒，而 CVTHead + 我们的方法在单块 NVIDIA RTX A6000 上仅需 3.92 秒，这表明我们的方法比 MotionClip 略快。

E. 消融研究

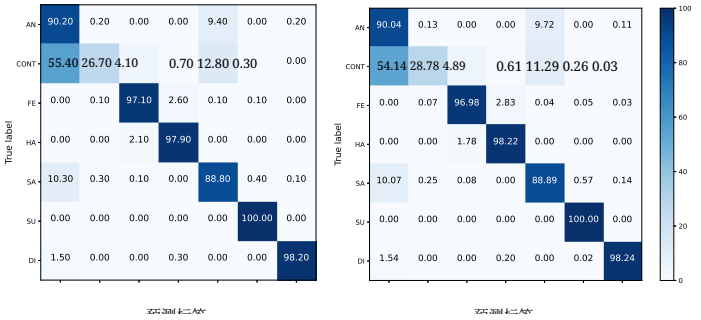
本节介绍了一系列消融研究，用于评估我们模型的性能。

1）所提出组件的有效性：为证明各个所提出组件的有效性，我们进行了对比实验，如表V所示。可以观察到，与不含IFED的I2FET相比，IFED模块在CK+ 和CelebV-HQ数据集上均显著提升了I2FET的性能。此外，将顶点损失函数与IFED结合后，在两个数据集上都提高了Acc_1 和几何均值。尽管在CK+上的Acc_2略有下降，但在CelebV-HQ上仍表现良好。

表V：我们的方法（I2FET）在两个数据集上不同所提出组件之间准确率的比较

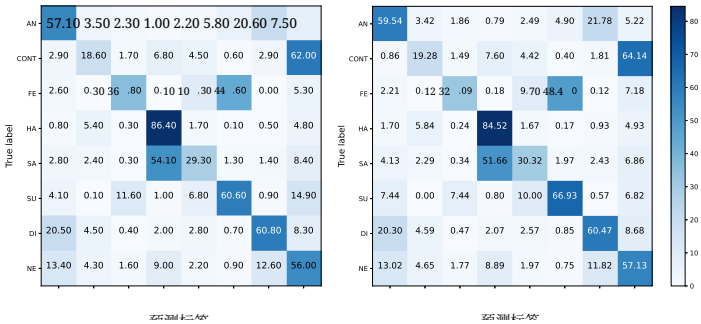
数据集	方法	准确率$_1$ (%) ↑	准确率$_2$ (%) ↑	几何均值 (%) ↑
CK+	未使用 IFED	76.89 ± 0.0029	63.23 ± 0.0047	63.59 ± 0.0059
	使用 IFED	91.30 ± 0.0017	**84.06 ± 0.0026**	79.41 ± 0.0061
	使用 IFED	**91.44 ± 0.0024**	84.03 ± 0.0034	**80.30 ± 0.0064**
	+w/ $\mathcal{L}_v$			
CelebV-HQ	不使用 IFED	48.35 ± 0.0037	22.1 ± 0.0040	32.88 ± 0.0058
	w/ IFED	57.80 ± 0.0020	32.60 ± 0.0037	46.04 ± 0.0062
	w/ IFED	**58.24 ± 0.0028**	**33.45 ± 0.0052**	**46.47 ± 0.0065**
	+结合 $\mathcal{L}_v$			

2) 三种损失函数的影响：为分析 I2FET 中各个损失函数的影响，我们开展了一项实验，比较模型在不同情形下的性能。具体而言，我们将 IFED 集成到 I2FET模型中，并逐步添加各个损失函数，以确定其对模型性能的影响。结果表VI表明，当姿态损失函数与表情损失函数结合时，性能有所提升；而当顶点损失函数与姿态和表情损失函数同时使用时，性能也有小幅改善。在本研究中，将顶点损失与其他损失函数结合，以捕捉面部几何结构，从而提高面部表情识别的准确率。图13和图14分别展示了我们的模型在总损失函数与 $\mathcal{L}_v$ 结合或不结合 $\mathcal{L}_v$ 的情况下，在CK+和CelebV-HQ数据集上的混淆矩阵。需要注意的是，在应用顶点损失后，两个数据集上的恐惧准确率反而略有下降。众所周知，恐惧表情经常与其他表情（例如惊讶和快乐）重叠，因此难以区分。此外，由于图5中CK+的恐惧部分远大于CelebV-HQ中的对应部分，表情转换过程中的混淆似乎对Acc_2 产生了相反的影响，如表VI所示



(a) (b)

图 13：I2FET 在 CK+ 上对七种面部表情的混淆矩阵：(a) 不含 $\mathcal{L}_v$，(b) 含 $\mathcal{L}_v$。



(a) (b)

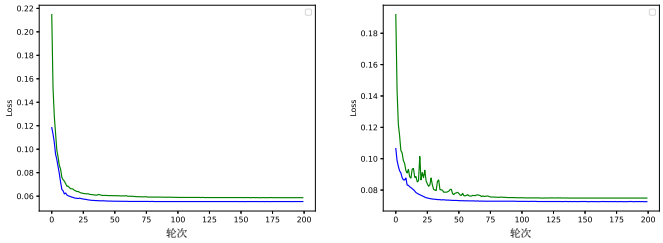
图 14：I2FET 在 CelebV-HQ 上对八种面部表情的混淆矩阵：(a) 不含 $\mathcal{L}_v$，(b) 含 $\mathcal{L}_v$。

表VI：损失函数对我们的方法（I2FET）在两个数据集上性能的影响比较

数据集	方法	准确率$_1$ (%) ↑	准确率$_2$ (%) ↑	几何均值 (%) ↑
CK+	$\mathcal{L}_e$	66.80 ± 0.0056	45.96 ± 0.0112	52.42 ± 0.0730
	$\mathcal{L}_e$+$\mathcal{L}_p$	91.30 ± 0.0017	**84.06 ± 0.0026**	79.41 ± 0.0061
	$\mathcal{L}_e$+$\mathcal{L}_p$+$\mathcal{L}_v$	**91.44 ± 0.0024**	84.03 ± 0.0034	**80.30 ± 0.0064**
CelebV-HQ	$\mathcal{L}_e$	37.57 ± 0.0052	11.99 ± 0.0326	25.82 ± 0.0114
	$\mathcal{L}_e$+$\mathcal{L}_p$	57.80 ± 0.0020	32.60 ± 0.0037	46.04 ± 0.0062
	$\mathcal{L}_e$+$\mathcal{L}_p$+$\mathcal{L}_v$	**58.24 ± 0.0028**	**33.45 ± 0.0052**	**46.47 ± 0.0065**

*3) IFED*对训练过程的影响：本实验旨在比较I2FET中加入和不加入IFED模块时的两种损失函数。图15a和图15b分别展示了CK+ 和CelebV-HQ上的结果，证实了I2FED发挥着重要作用。

4) IFED 对面部表情强度的影响：图 16 展示了IFED 如何帮助 I2FED 生成高强度面部表情。



) 图 15：I2FET在 (a) CK+ 和 (b) CelebV-HQ 上训练时，有 IFED 和无 IFED 情况下验证损失的对比。绿色曲线表示不含 IFED 的 I2FET，蓝色曲线则表示含 IFED 的 I2FET 的结果。

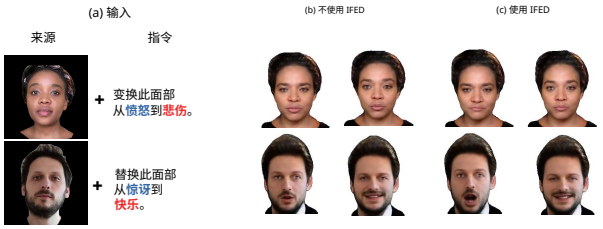


图 16：有 (a) 和无 (b) IFED 时生成的 3D 表情面部的定性比较。

*5) CAFT*层数量的影响：我们通过设计包含不同数量CAFT层的 IFED模块，开展实验以评估所提出的 I2FET模型的结果。我们将两个分支中的 Transformer层数量固定为 1，并调整 CAFT层的数量。表 VII 中的结果表明，增加 CAFT层的数量并不能提升所提出的 I2FET 的性能。

6）面部参数分支和文本分支中 *Transformer* 编码器数量的影响：在本实验中，我们通过调整 IFED 的 Transformer 编码器层数，采用了不同的配置。结果表明，与其他配置相比，当 **N** = 2 和**M** = 1时，I2FET 模型的表现更佳，尤其是在 Acc_1和 G-mean 方面，如表 VIII 所示。

表 VII：通过改变 I2FED 中 CAFT 层的数量，比较 I2FET 方法在两个数据集上的准确率

数据集	#层数	准确率$_1$ (%) ↑	准确率$_2$ (%) ↑	几何均值 (%) ↑
CK+	1	**91.36 ± 0.0023**	**84.13 ± 0.0046**	**79.65 ± 0.0080**
	2	88.95 ± 0.0019	80.95 ± 0.0038	78.72 ± 0.0071
CelebV-HQ	1	**58.25 ± 0.0034**	**33.03 ± 0.0056**	**46.46 ± 0.0066**
	2	55.68 ± 0.0034	29.82 ± 0.0066	42.59 ± 0.0085

表VIII：在 CK+ 上，TFED 中面部参数分支（**N**）和文本分支（**M**）的 Transformer 层数与所提出方法准确率的比较

模式	**N**	**M**	准确率$_1$ (%) ↑	准确率$_2$ (%) ↑	几何均值 (%) ↑
A	1	1	91.36 ± 0.0023	**84.13 ± 0.0046**	79.67 ± 0.008
B	2	1	**91.44 ± 0.0024**	84.03 ± 0.0034	**80.30 ± 0.0064**
C	1	2	91.27 ± 0.0019	83.84 ± 0.0037	79.69 ± 0.0052
D	2	2	89.07 ± 0.0021	81.13 ± 0.0034	78.71 ± 0.0070

F. 使用中性表情生成较长的面部表情

根据评价理论 [45]，（主要情绪理论之一），情绪的动态过程被描述为情绪分化序列检查理论。换言之，人类情绪状态的分化是特定刺激评估评价检查序列的结果，而非以离散方式从一种情绪状态跳跃到另一种情绪状态。在我们的框架中，面部表情从 A 到 B 的转变类似于情绪状态的序列分化：从具有强烈情绪的 A 面部表情开始，经过中性表情，再转变为强烈的面部表情 B。除了评价理论之外，本方法也适用于研究愉悦度-唤醒度（P-A）空间 [46]，中的面部行为；在该空间中，中性表情充当两种特定表情之间的过渡表情。为此，我们采用了中性编码器-解码器（NED）来生成中性面孔。此举旨在当指令中未提供中性表情时，提升框架的性能，并增强其扩展面部表情序列的灵活性。

1）实现细节：在此情境下，构建了两个包含中性表情和姿态参数的数据集，并使用预训练的 DECA 从 CelebV-HQ 和 CK+ 数据集中提取数据。由于 CK+ 数据集不包含中性表情，因此提取每段视频的第一帧，并将其视为代表性中性案例。随后，使用这两个中性数据集训练 NED。

2）网络架构：中性编码器-解码器模型由全连接层构成。编码器包含四个全连接层，而解码器则由四个全连接层组成，分别用于姿态和表情。

3) 分析： 同样，我们也进行了实验，从定量和定性两个方面评估中性表情生成效果。

定量比较：具体而言，NED 在 CK+ 和 Celeb V-HQ 数据集上生成中性表情的准确率分别为 49.5% 和 51.33%。结果见表IX，证实 NED +CVTHead 的表现优于 NED + ROME。

定性比较：图 17 展示了 NED + CVTHead 在 CK+ 和 CelebV-HQ 上生成的中性表情。此外，我们还展示了 FLAME 在表情参数设为零时生成的中性表情，如图 17c 所示。

结果表明，面部单元会因环境是否可控而有所不同，

也会因是否使用 FLAME 而有所不同。这表明，现实场景中的中性表情更为复杂，因此我们倾向于使用神经编码器-解码器，而非 FLAME 提供的表情参数设置

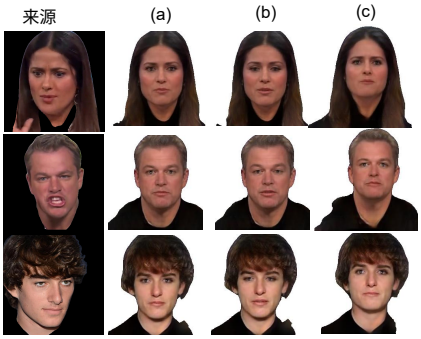


图 17：NED+CVTHead生成的中性表情面部图像样本。最左列展示输入图像（来源）。使用CK+数据集（可控环境）生成的图像见(a)，而使用CelebV-HQ数据集（非可控环境）生成的图像见(b)。另一方面，(c)展示了将表情参数设为零时，使用FLAME生成的三维人脸。请注意，这三种生成结果的图像质量存在细微差异。

表IX：NED分别与ROME或CVTHEAD结合，用于在CK+ 和CELEBV-HQ上生成纹理时所获得的结果。

数据集	方法	L_1↓ 峰值信噪比↑ LPIPS↓ 多尺度结构相似性↑	
CK+	NED + ROME	0.193 10.207 0.516	0.348
	NED + CVTHead 0.004 35.158 0.014		**0.987**
CelebV-HQ	NED +ROME	0.183 10.818 0.498	0.407
	NED + CVTHead 0.004 36.347 0.0117 0.989		

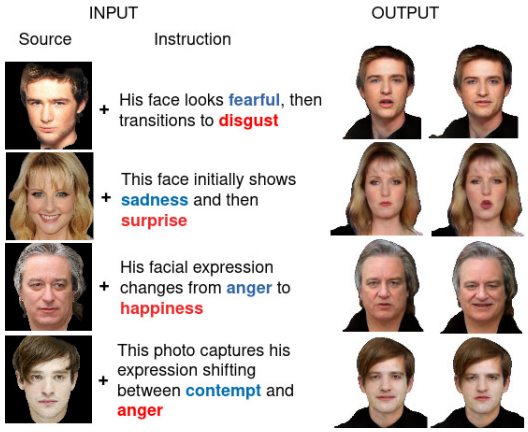
在一项与第 IV-D2 节所述类似的用户研究中，我们请受试者判断生成的图像是否呈现中性表情。大多数受试者都成功识别出我们的模型在 CK+上生成的中性表情，而对于学习模型在 CelebV-HQ 上生成的样本，他们仅对其中 4% 难以判断。

G. 面部表情分类

1) 实现： 对于带有 RFL [43], 的 ResNet-101 [41] 和 ResNet-101，学习率为 $1e^{-4}$ ，批量大小为 64。我们使用 Adam 优化模型，每个周期训练 50 轮。对于 MEK 模型，我们采用 [42] 中的配置来优化 CK+ 和 CelebV-HQ 数据集的训练过程。

2) 分析： 根据表 X 中的结果，与 ResNet-101 或 MEK 相比，带有 RFL 的 ResNet-101 在分类性能方面更加稳定，这表明带有 RFL 的 ResNet-101 能够较好地处理不平衡数据集和旋转变换。具体而言，MEK 在原始图像上取得了最佳结果，但其性能受到旋转图像的影响。在 CK+上，带有 RFL 的 ResNet-101 比 ResNet-101 和 MEK 表现得更加稳定。对于 CelebV-HQ 数据集中的现实场景，我们也获得了类似的结果。这些发现表明，面部表情的性能
对于旋转 ±15度或以上的面部，分类器的性能会下降

此外，MEK 在原始数据集上的表现更好，但在增强数据集上的表现较差。ResNet-101 在不平衡率较低的数据集上性能较高，而在不平衡率较高的数据集上性能较低。同时，采用 RFL 的 ResNet-101 在不平衡和旋转数据集上表现稳定。基于这一结果，我们在评估过程中默认采用 ResNet-101。



(a)



(b)

图18：通过更改模板句生成的三维人脸（a），以及一些未成功的案例（b）。

H. 讨论

1）面部表情序列的多样性：我们的框架能够在指令和具有姿态变化的面部图像的引导下，生成各种面部表情动画，如图12所示。此外，它能够生成与受控环境和真实环境相对应的面部表情，从而生成丰富多样的表情，用于制作长视频。

2）更改模板句时的指令结果：图18a展示了更改模板句后得到的结果。结果表明，我们的模型能够成功生成与指令中描述的特定情绪高度相似的面部表情。

3) 失败案例：尽管我们的模型在许多情况下表现良好，但在某些情况下仍会失败，如图18b所示。第一行中，输入图像中的姿态变化较大，导致生成图像缺失部分肩部，这是由于预训练人脸渲染模型的限制所致。

表 X：比较 CK+和 CELEBV-HQ 上不同角度旋转样本的面部表情分类器$\{0^o, \pm10^o, \pm15^o, \pm30^o\}$

数据集	方法	0°		$\pm10^o$		$\pm15^o$		$\pm30^o$	
		准确率$_1$ ↑	G-mean ↑	准确率$_1$ ↑	G-mean ↑	准确率$_1$ ↑	G-mean ↑	准确率$_1$ ↑	G-mean ↑
CK+	MEK [42]	**100**	**100**	86.41	**76.20**	76.01	52.16	32.56	0.0
	ResNet-101 [41]	99.56	99.4	84.40	73.28	79.50	**67.6**	75.29	**33.39**
	采用 RFL 的 ResNet-101	99.69	99.69	**87.18**	75.85	**83.66**	66.13	**77.48**	31.5
CelebV-HQ	MEK [42]	**99.75**	**99.83**	**60.49**	**55.07**	49.08	33.84	32.05	9.64
	ResNet-101 [41]	37	12.77	35.77	13.39	34.97	0.0	28.92	0.0
	采用 RFL 的 ResNet-101	93.67	95.04	58.54	49.92	**51.53**	**41.22**	**40.71**	**26.80**

同样，源图像中缺失部分头部区域或头发的图像，也会因这一限制而导致人脸渲染过程中信息不足，如第二行所示。第二行中，生成结果受到表情词汇规模的限制，导致害怕的面部被转换成愤怒的面部。

五、结论

本研究提出了一种新颖的框架，可根据输入的RGB图像生成三维面部表情，并按照文本指令指定的内容，为两种表情之间的面部表情转换制作动画。首先，我们引入FET来生成面部表情序列。我们设计了指令驱动的面部表情分解器，使其能够从多模态数据中学习，并捕捉文本描述与面部表情特征之间的相关性。我们提出了指令到面部表情转换模型，通过单条指令引导生成目标面部表情。我们的框架结合FET与预训练的人脸渲染模型，生成与预期表情序列相一致的面部外观。此外，我们的框架还可扩展为生成中性表情，从而能够在视频中创建多种表情，逼真地模拟现实场景中的面部表情行为。我们在CK+ 和CelebV-HQ数据集上进行了大量实验，以验证框架的有效性。尽管我们提出的方法能够根据给定指令控制RGB图像中的面部表情，并取得了积极效果，但仍存在一些局限性。与以往使用面部参数的研究 [9], [11] 类似，我们模型的性能取决于三维人脸系数的精度；尤其是在我们的配置中使用DECA [36]时，模型可能难以解耦面部因素，这在具有挑战性的场景中尤为常见。此外，尽管线性插值能够保持时间一致性，却往往会损害合成视频的真实感。虽然目前的工作仅使用基本面部表情或两种指定表情，但通过将大语言模型（LLM）与我们的模型结合，词汇数量可以轻松扩展。鉴于文本提示是一种强大的工具，可借助我们的框架在三维虚拟化身上表达各种情绪状态，我们期待它在未来得到广泛应用。

致谢

本研究得到信息技术研究中心（ITRC）支持计划（IITP-2022-RS-2022-00156354）和韩国政府（MSIT）通过信息通信技术规划评估院（IITP）提供的资助（编号：RS-2019-II190231）的支持；同时也得到韩国国家研究基金会（NRF）通过教育部资助的基础科学研究计划（2020R1A6A1A03038540）支持。

参考文献

[1] Z. Huang、K. C. Chan、Y. Jiang 和 Z. Liu，“用于多模态人脸生成与编辑的协作扩散”，载于 *IEEE/CVF*计算机视觉与模式识别会议论文集，2023 年。[2] W. Xia、Y. Yang、J.-H. Xue 和 B. Wu，“Tedigan：文本引导的多样化人脸图像生成与操控”，载于 *IEEE* 计算机视觉与模式识别会议（*CVPR*），2021 年。[3] Y. Jiang、Z. Huang、X. Pan、C. C. Loy 和 Z. Liu，“Talk-to-edit：通过对话进行细粒度面部编辑”，载于 *IEEE/CVF*国际计算机视觉会议论文集，2021 年。[4] M. Kim、F. Liu、A. Jain 和 X. Liu，“Dcface：利用双条件扩散模型进行合成人脸生成”，载于 *IEEE/CVF*计算机视觉与模式识别会议论文集（*CVPR*），2023 年 6 月，第 12 715–12 725 页。[5] S. Su、J. Zhu、L. Gao 和 J. Song，“利用贪婪特性在 Transformer 中进行多模态条件图像合成”，*IEEE*多媒体汇刊，第 26 卷，第 2354–2366 页，2024 年。[6] H. Ding、K. Sricharan 和 R. Chellappa，“Exprgan：具有可控表情强度的面部表情编辑”，*AAAI*，2018 年。[7] J. Guo、H. Manukyan、C. Yang、C. Wang、L. Khachatryan、S. Navasardyan、S. Song、H. Shi 和 G. Huang，“Faceclip：通过简短文本描述实现面部图像到视频的转换”，*IEEE*视频技术电路与系统汇刊，第 1–1 页，2023 年。[8] T. Hang、H. Yang、B. Liu、J. Fu、X. Geng 和 B. Guo，“基于循环StyleGAN的生成器进行语言引导的人脸动画”，*IEEE*多媒体汇刊，第 25 卷，第 1–12 页，2023 年。[9] T. Khakhulin、V. Sklyarova、V. Lempitsky 和 E. Zakharov，“逼真的单次网格头部头像”，载于 欧洲计算机视觉会议（*ECCV*），2022 年。[10] W. Zielonka、T. Bolkart 和 J. Thies，“即时体积头部头像”，载于 *IEEE/CVF*计算机视觉与模式识别会议（*CVPR*），2023 年 6 月。[11] H. Ma、T. Zhang、S. Sun、X. Yan、K. Han 和 X. Xie，“CVTHead：利用顶点特征Transformer实现单次可控头部头像”，*IEEE/CVF*冬季计算机视觉应用会议（*WACV*），2024 年。[12] B. Peng、H. Fan、W. Wang、J. Dong 和 S. Lyu，“高保真人脸替换与表情重演的统一框架”，*IEEE*视频技术电路与系统汇刊，第 32 卷，第 6 期，第 3673–3684 页，2022 年。[13] Y. Chen 和 S. Xiong，“重新思考单次人脸重演：时空重建视角”，基于知识的系统，第 277 卷，C 期，2023 年。[14] S. Hwang、J. Hyung、D. Kim、M.-J. Kim 和 J. Choo，“Faceclipnerf：利用可变形神经辐射场进行文本驱动的三维人脸操控”，载于 *IEEE/CVF*国际计算机视觉会议（*ICCV*）论文集，2023 年 10 月，第 3469–3479 页。

[15] S. Aneja、J. Thies、A. Dai 和 M. Nießner，“Clipface：文本引导的纹理化三维可变形模型编辑”，载于 SIGGRAPH ’23 会议论文集，2023 年。[16] M. C. Doukas、S. Zafeiriou 和 V. Sharmanska，“HeadGAN：单次神经头部合成与编辑”，载于 IEEE/CVF国际计算机视觉会议（ICCV）论文集，2021 年 10 月，第 14 398–14 407 页。[17] M. C. Buehler、A. Meka、G. Li、T. Beeler 和 O. Hilliges，“VariTex：变分神经人脸纹理”，载于 IEEE/CVF国际计算机视觉会议论文集，2021 年。[18] P.-W. Grassal、M. Prinzler、T. Leistner、C. Rother、M. Nießner 和 J. Thies，“从单目 RGB视频生成神经头部化身”，载于 IEEE/CVF计算机视觉与模式识别会议论文集，2022 年，第 18 653–18 664 页。[19] F. Paraperas Papantoniou、P. P. Filntisis、P. Maragos 和 A. Roussos，“Neural Emotion Director：野外视频中保留语音的面部表情语义控制”，载于 IEEE/CVF计算机视觉与模式识别会议（CVPR）论文集，2022 年。[20] S. Qian、T. Kirschstein、L. Schoneveld、D. Davoli、S. Giebenhain 和 M. Nießner，“GaussianAvatars：利用绑定骨骼的三维高斯实现逼真头部化身”，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2024 年。[21] Y. Wu、S. Xu、J. Xiang、F. Wei、Q. Chen、J. Yang 和 X. Tong，“AniportraitGAN：从二维图像集合生成可动画化三维肖像”，载于 SIGGRAPH Asia 2023 会议论文集，2023 年。[22] N. Otberdout、C. Ferrari、M. Daoudi、S. Berretti 和 A. Bimbo，“通过学习面部标志点轨迹生成复杂的四维表情过渡”，IEEE 情感计算汇刊，2023 年。[23] K. Zou、S. Faisan、B. Yu、S. Valette 和 H. Seo，“四维面部表情扩散模型”，2023 年。[24] X. Gu、C. Wen、W. Ye、J. Song 和 Y. Gao，“SEER：利用潜在扩散模型进行语言指令视频预测”，ICLR 2024，2023 年。[25] O. Patashnik、Z. Wu、E. Shechtman、D. Cohen-Or 和 D. Lischinski，“StyleCLIP：文本驱动的StyleGAN图像操控”，载于 IEEE/CVF国际计算机视觉会议（ICCV）论文集，2021 年 10 月，第 2085–2094 页。[26] T. Li、T. Bolkart、M. J. Black、H. Li 和 J. Romero，“从四维扫描中学习面部形状与表情模型”，ACM 图形学汇刊（TOG），第 36 卷，第 1–17 页，2017 年。[27] T. Sutter、I. Daunhawer 和 J. Vogt，“利用 Jensen-Shannon 散度进行多模态生成学习”，第 33 卷，2020 年，第 6100–6110 页。[28] M. Suzuki 和 Y. Matsuo，“多模态深度生成模型综述”，Advanced Robotics，第 36 卷，第 261–278 页，2022 年。[29] N. Otberdout、C. Ferrari、M. Daoudi、S. Berritti 和 A. Del Bimbo，“从稀疏到稠密的动态三维面部表情生成”，载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2022 年 6 月。[30] G. Tevet、B. Gordon、A. Hertz、A. H. Bermano 和 D. Cohen-Or，“MotionClip：将人体动作生成引入CLIP空间”，载于《计算机视觉–ECCV 2022：第 17 届欧洲会议》，2022 年，第 358–374 页。[31] R. Danˇeˇcek、K. Chhatre、S. Tripathi、Y. Wen、M. Black 和 T. Bolkart，“具有内容—情绪解耦的情感语音驱动动画”，载于 SIGGRAPH Asia 2023 会议论文，2023 年。[32] Y. Lee、T. Choi、H. Go、H. Lee、S. Cho 和 J. Kim，“Exp-GAN：具有表情控制的三维感知面部图像生成”，载于《计算机视觉–ACCV 2022：第 16 届亚洲计算机视觉会议》。Springer-Verlag，2023 年，第 151–167 页。[33] A. Radford、J. W. Kim、C. Hallacy、A. Ramesh、G. Goh、S. Agarwal、G. Sastry、A. Askell、P. Mishkin、J. Clark 等，“从自然语言监督中学习可迁移视觉模型”，载于国际机器学习会议。PMLR，2021 年，第 8748–8763 页。[34] C.-F. Chen、Q. Fan 和 R. Panda，“CrossViT：用于图像分类的交叉注意力多尺度视觉Transformer”，2021 IEEE/CVF国际计算机视觉会议（ICCV），第 347–356 页，2021 年。[35] K.-E. Lin、A. Trevithick、K. Cheng、M. Sarkis、M. Ghafoorian、N. Bi、G. Reitmayr 和 R. Ramamoorthi，“PVP：利用StyleGAN实现可编辑动态肖像的个性化视频先验”，Computer Graphics Forum，第 42 卷，第 4 期，第 13 页，2023 年。[36] Y. Feng、H. Feng、M. J. Black 和 T. Bolkart，“从野外图像中学习可动画化的详细三维人脸模型”，载于 ACM 图形学汇刊（SIGGRAPH 论文集），第 40 卷，第 8 期，2021 年。[37] P. Lucey、J. F. Cohn、T. Kanade、J. Saragih、Z. Ambadar 和 I. Matthews，“扩展 Cohn-Kanade数据集（CK+）：用于动作单元和情感指定表情的完整数据集”，载于 2010 IEEE 计算机学会计算机视觉与模式识别会议——研讨会，2010 年，第 94–101 页。[38] H. Zhu、W. Wu、W. Zhu、L. Jiang、S. Tang、L. Zhang、Z. Liu 和 C. C. Loy，“CelebV-HQ：大规模视频面部属性数据集”，发表于 *ECCV*，2022 年。[39] X. Li、D. Zhang、M. Li 和 D.-J. Lee，“利用图像校正和轻量级卷积神经网络进行精确头部姿态估计”，*IEEE*多媒体汇刊，第 25 卷，2251，2023 2239页。[40] D. P. Kingma 和 J. Ba，“Adam：一种随机优化方法”，*ICLR*，第 abs/1412.6980 卷，2014 年。[41] K. He、X. Zhang、S. Ren 和 J. Sun，“用于图像识别的深度残差学习”，发表于*2016*年*IEEE*计算机视觉与模式识别会议论文集，系列：CVPR ’16。IEEE，2016 年，第 770–778 页。[42] Y. Zhang、Y. Li、lixiong Qin、X. Liu 和 W. Deng，“不放过任何细节：挖掘额外知识以实现类别不平衡的面部表情识别”，发表于第三十七届神经信息处理系统会议，2023 年。[43] Y. Cui、M. Jia、T.-Y. Lin、Y. Song 和 S. J. Belongie，“基于有效样本数的类别平衡损失”，*2019*年*IEEE/CVF*计算机视觉与模式识别会议（*CVPR*），第 9260–9269 页，2019 年。[44] R. Zhang、P. Isola、A. A. Efros、E. Shechtman 和 O. Wang，“深度特征作为感知度量的惊人效能”，发表于*CVPR*，2018 年。[45] K. R. Scherer，“将评价视为多层次序列检查过程。”牛津大学出版社，第 92–120 页，2001 年。[46] J. Russell 和 J. M. F. dez Dols，“从面部读取情绪并将情绪读入面部：重拾维度—情境视角”，面部表情心理学，第 295–320 页，1997 年。