基于特征嵌入的 3D 点云人脸识别

杨俊鹏

摘要

由于大规模训练数据的可用性，2D 人脸识别 (FR) 的准确性有了显著的进步。然而，基于深度学习的 3D FR 的研究仍处于早期阶段。大多数可用的 3D FR 从 3D 数据生成 2D 图，并将现有的 2D cnn 应用到生成的 2D 图上进行特征提取。本文在 PointNet++ 框架的基础上复现并加入了 RS-CNN（Relation-Shape Convolutional Neural Network）用于特征提取，用于直接处理 3D 人脸的点集数据。在该框架中，设计了两个权重共享编码器，直接从一对 3D 人脸中提取特征，并分别使同一人和不同人的嵌入距离最小化和最大化。该框架还使用特征相似度损失来指导编码器获得鉴别人脸表示。为了进一步提高 FR 性能，我们从 transformer 中得到灵感，在其中加入了自注意力机制，最终效果得到一定的提升。

关键词：三维人脸识别；点云处理；深度学习；自注意力

1 引言

人脸识别是目前最常用的生物特征识别技术之一。在过去的几年里，由于二维卷积神经网络（cnn）的发展，基于深度学习的二维人脸识别（FR）取得了巨大的成功，已经超过了人类主动识别的性能。虽然有大量的训练数据和精心设计的 2D cnn，但 2D FR 仍然受到 2D 图像固有局限性的挑战，例如各种光照条件引起的像素值的变化，以及不同头部姿势引起的自聚焦。与 2D 图像相比，3D 人脸数据（如点云）对姿态和光照变化具有不变性，因此可以提供更丰富的几何信息。三维人脸数据的特性可以帮助人脸识别系统克服二维人脸识别的固有缺陷。因此，3D 人脸识别已成为近年来一个活跃的研究课题。

然而，在 3D FR 中，大多数传统的解决方案包括手工制作的基于特征的方法和深度学习的方法有几个缺陷。首先，它们都需要在特征提取之前进行数据转换，这需要大量的计算成本。许多基于特征的手工方法需要从点云或深度图像中重建 3D 网格（也称为 3D 多边形曲面），然后根据重构网格的关键点、曲率、形状指标和曲线计算特征描述子。此外，与基于深度学习的方法相比，许多基于特征的手工方法在带有噪声或带有姿态变化的人脸数据上泛化效果不太好。对于基于深度学习的方法，许多研究者考虑直接利用 2D cnn 进行 3D FR 任务。而 3D 人脸最常见的数据格式点云，与二维网格数据（如图像）相比，是无序的、无结构的。因此，需要将 3D 人脸转换为 2D 地图。具体来说，研究人员将深度图像转换为点云进行预处理（例如裁剪和密集对齐），然后从点云估计二维几何图，作为二维 cnn 的输入进行特征提取。虽然基于深度学习的方法在 3D FR 上取得了令人满意的性能，但由于数据重采样，数据转换过程中不可避免地会导致几何信息的丢失。

我们认为，可以直接从 3D 人脸点云中提取特征的深度学习模型更加简洁和有效。点云在重构算法上是独立的，也可以提供形状信息。尽管基于 PointNet 和 PointNet++ 的方法在常规三维物体识别上取得了令人印象深刻的表现，但在三维人脸识别任务上，它们获得的结果并不令人满意。原因是不同类别的常规三维形状之间的几何差异很大，而不同个体的三维面孔的几何结构看起来非常相似。

应设置一种能够探索区分三维人脸之间微小几何差异的深度网络。为了解决上述问题，我们复现了RS-CNN 作为特征提取模块，并且在对比学习的启发下，我们提出了一个轻量级但有效的框架（如图1所示），以直接从三维点云中学习三维 FR 的细粒度表示。

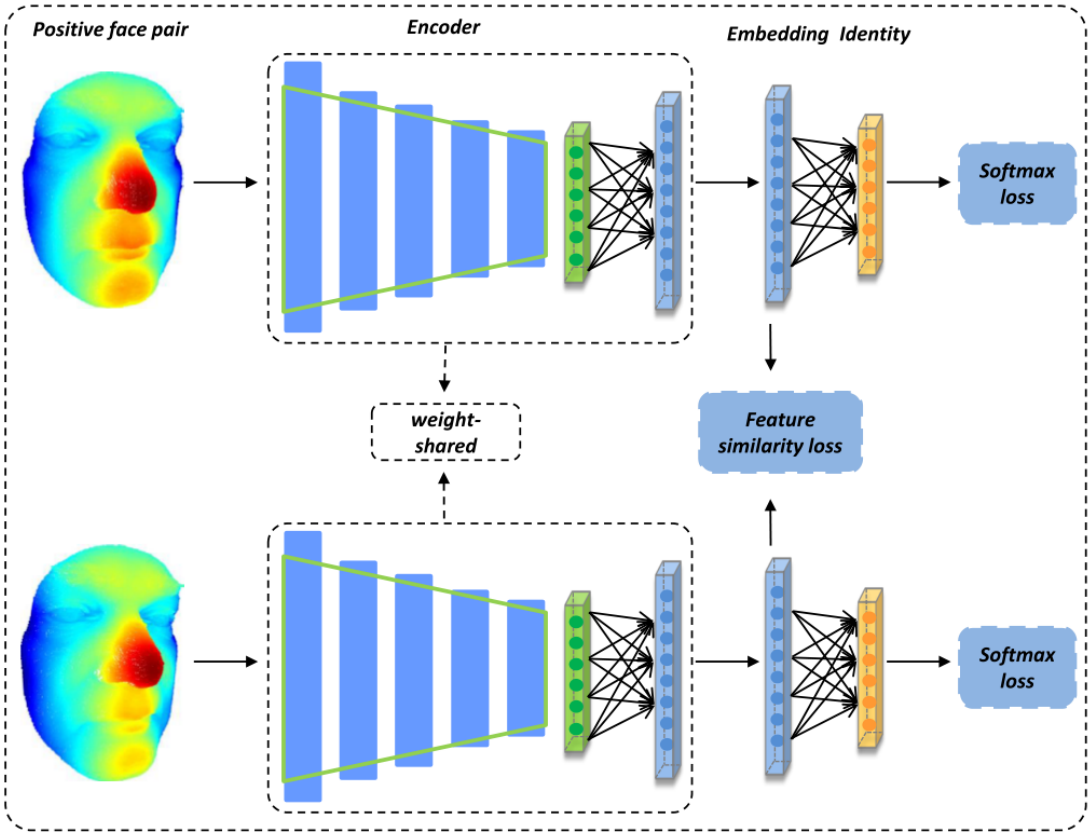

图 1: 框架图

模型由两个共享权重的编码器和两个共享权重的全连接层组成，前者从一对点云面孔中提取嵌入（特征向量），后者在训练期间将嵌入映射到身份（ground-truth）。我们用集合抽象模块和全连接层构建编码器，并且提出了一个配对选择策略来产生正反两方面的配对以促进训练，并设计了特征相似性损失来监督我们的编码器学习更多的辨别特征。特征相似度损失的目的是通过一个距离差将正样本与负样本分开。同时设计 softmax 损失评估身份分类准确性，用来监督网络训练。联合学习样本之间的差异和每个样本本身的身份信息提高了编码器的性能。在测试阶段，我们只需要编码器从三维人脸中提取特征向量用于识别和验证任务。我们的方法主要在 Lock3DFace 上进行实验。同时还与用 2D FR 任务中使用的常见损失函数训练的单一编码器进行比较。实验细节将在第 4 节描述。

2 相关工作

2.1 三维人脸重建

3D FR 的方法可以大致分为两类：基于手工制作的特征方法和基于深度学习的方法。这两种方法都在一些公开的数据库上进行了评估，如 FRGC v2[1]和 Bosphorus[2]，并在这些高质量的数据库上取得了令人印象深刻的性能。最近，Zhang 等人[3]提出了一个低质量的人脸数据库，并提出了一个新的挑战：低质量数据上的三维人脸识别。

基于手工制作特征的方法[4][5]从关键点、曲率、形状指数中提取特征。和三维人脸网格的曲线，然后通过比较这些特征来识别人脸。例如，Mian 等人[6]提出了一种针对三维人脸的可重复的关键点检测

算法，并用二维尺度不变特征变换表示三维关键点，用于多模态人脸识别。Gilani 等人[7]提出了一个基于关键点的密集对应模型，并通过匹配三维可变形模型的参数进行三维 FR。Samir 等人[5]通过水平曲线的联合来表示表面，称为面部曲线，并比较了面部曲线的形状，用于人脸识别。

然而，基于手工制作的特征方法在廉价三维扫描仪捕获的嘈杂数据上表现的性能无法令人满意，而且它们在大型数据库上的概括能力也很有限.

对于基于深度学习的方法，许多研究人员努力将二维 CNN 直接用于三维 FR 任务，将三维表面转化为二维几何图（如法线图）并使用二维 CNN 提取特征。例如，Kim 等人[8]使用一个包含 123,325 张深度图像的增强数据集对预训练的 VGG-Face 网络进行了微调。Gilani[9]通过生成大量具有丰富形状变化的合成三维面孔来扩展训练数据，并将三通道几何图输入到 VGG-16 网络[10]。

最近，Mu 等人[11]提出了一个轻量级网络 Led3D，以在低质量数据上执行 FR，他们用二维深度图像与法线图串联来训练 Led3D。虽然[11]在低质量的 3D 数据上取得了最先进的性能，但他们的算法也将 3D 数据转换为 2D 地图，这可能会造成几何信息损失。

2.2 基于深度学习的点云处理

由于点云的结构是不规则的和稀疏的，在深度学习时代，网络不仅需要从点云中找到一个有效的表示，还需要满足排列不变性和尺度不变性。为了将 CNN 应用于三维物体识别，一些研究人员考虑将点云体素化，并应用三维 CNN 来处理体积表示[12],[13]。这类方法存在计算复杂性和内存成本立体增长的问题，从而限制了三维网格的分辨率。

作为基于深度学习的点云处理的一项开创性工作，Qi 等[14]是第一个提出高效网络 PointNet 的人，利用排列不变性直接处理点云，在常规三维形状分类中表现出满意的性能。为了克服 PointNet 的一些缺点，Qi 等[15]提出了一种层次网络 Pointnet++，进一步提高了常规三维形状识别的性能。基于 PointNet 和 Pointnet++，Wang 等人[16]提出 EdgeConv，使用从点云计算的动态图来纳入局部邻域特征。与[16]类似，Zhou 等人[17]提出了自适应图形卷积（AdaptConv）来设计自适应卷积核，基于动态学习的特征和每个点的全球位置。

3 本文方法

3.1 整体架构

如图 1所示，给定训练用的点云数据，我们首先通过我们提出的策略（详见第 3.3 节）将其配对作为输入。然后，由一组抽象模块组成的两个分权编码器（详见第 3.2 节）将点云对编码为两个 512 维的嵌入，用于身份预测和特征相似度测量。

在训练阶段，我们设计了特征相似性损失来评估两个样本的嵌入之间的相似性，以便编码器能够区分来自不同个体的三维人脸，并将来自同一个体的人脸的特征紧凑地聚类。我们采用 softmax 分类损失来监督编码器的训练，这样编码器就能从每个样本本身学到身份信息。在特征相似性损失和 softmax 损失的共同监督下，我们的编码器可以获得更细化的表征。在测试阶段，我们只需要由编码器产生的嵌入来进行身份识别。

3.2 编码器

如图 2所示，编码器使用五个集合抽象模块来产生 64、128、256、512、1024 维度的特征，以及一个全连接层来产生 512 维度的特征。

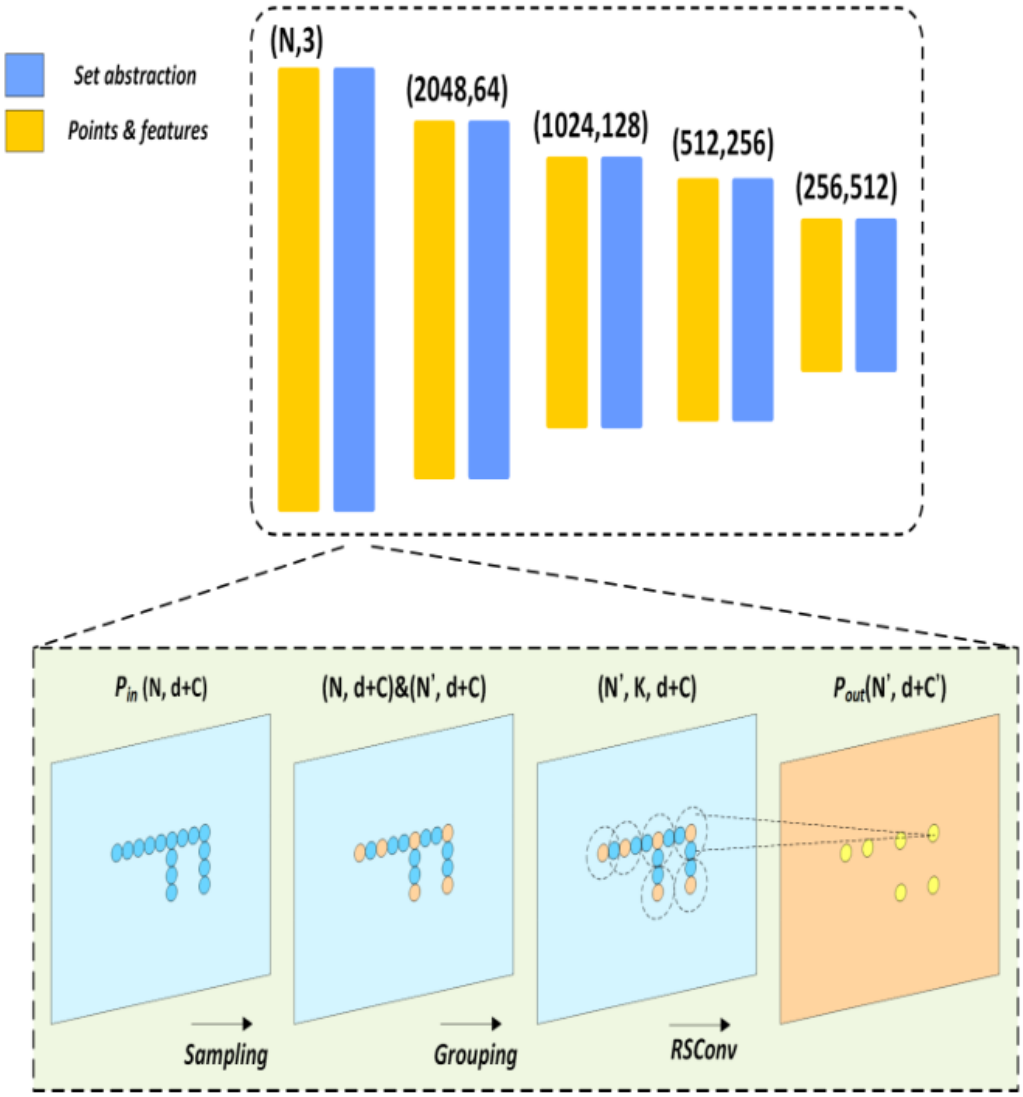

图 2: 方法示意图

集合抽象模块是一个类似于二维图像卷积的模块，对一个点集进行抽象，产生一个具有较少元素和高层次特征的新集。集合抽象模块由三个关键层组成。采样层、分组层和特征学习层。

给定一个有 N 个点的输入三维点云

$$P_{\text{in}} = \{p_i \in \mathbb{R}^3; i = 1, 2, \ldots, N\} \tag{1}$$

和其相应的特征

$$F_{\text{in}} = \{f_{p_i} \in \mathbb{R}^C; i = 1, 2, \ldots, N\} \tag{2}$$

我们可以用一个大小为 N ×（3+C）的张量 M_{in} 来表示输入的点云。

$$M_{\text{in}} = \{(p_i, f_{p_i}) \in \mathbb{R}^{3+C}; i = 1, 2, \ldots, N\} \tag{3}$$

采样层选择 N' 个点来定义输入点云 M_{in} 的局部区域的中心点。设 M_{cen} 为采样层的输出。这个

过程可以表述为：

$$\begin{aligned} M_{\text{cen}} &= \text{Sampling layer}\ (M_{\text{in}}) \\ M_{\text{cen}} &= \left\{ \left(p_j^{\text{cen}}, f_{p_j^{\text{cen}}} \right) \in \mathbb{R}^{3+C}; j = 1, 2, \ldots, N' \right\} \end{aligned} \tag{4}$$

其中 M_{cen} 是一个由中心点和其相应特征组成的张量。分组层通过寻找 K 个邻居来构建中心点周围的局部区域，每个中心点在球状半径 r 范围内，该过程可表述为：

$$M_{\text{group}} = \text{Grouping layer}\ (M_{\text{in}}, M_{\text{cen}}) \tag{5}$$

其中 M_{group} 是大小为 $N_{'}$ × K ×（3+C）的张量。特征学习层对每个局部区域模式进行编码和聚合，使之成为一个更聚合的表示。这个过程可以表述为：

$$\begin{aligned} M_{\text{out}} &= \text{Feature learning layer}\ (M_{\text{group}}) \\ M_{\text{out}} &= \left\{ \left(p_j^{\text{cen}}, f'_{p_j^{\text{cen}}} \right) \in \mathbb{R}^{3+C'}; j = 1, 2, \ldots, N' \right\} \end{aligned} \tag{6}$$

RS-CNN 旨在学习围绕某个中心点的局部区域中每个点的权重，并将其汇总以获得局部区域的浓缩表示。权重是通过使用 MLPs 将低级关系先验（如欧氏距离和相对位置）映射到高级关系向量来学习的。具体来说，让 p_j^{cen} 是由采样层采样的中心点之一，$\mathcal{N}\left(p_j^{\text{cen}}\right)$ 是其在球形半径 r 内的邻近区域，RS-CNN 可以被表述为:

$$f'_{p_j^{\text{cen}}} = \mathcal{A}\left(\left\{\mathcal{E}\left(f_{p_k}\right), \forall p_k \in \mathcal{N}\left(p_j^{\text{cen}}\right)\right\}\right) \tag{7}$$

其中，$\mathcal{A}$ 是一个对称的聚合函数（我们在这里使用 maxpooling），帮助卷积实现排列不变性并聚合局部特征；$\mathcal{E}$ 是一个学习 $\mathcal{N}\left(p_j^{\text{cen}}\right)$ 中所有点的特征的函数，$\mathcal{E}$ 被表述为。

$$\mathcal{E}\left(f_{p_k}\right) = \mathcal{M}\left(h\left(p_j^{\text{cen}}, p_k\right)\right) \otimes f_{p_k} \tag{8}$$

化简后，式子（7）变为：

$$f'_{p_j^{\text{cen}}} = \mathcal{A}\left(\left\{\mathcal{M}\left(h\left(p_j^{\text{cen}}, p_k\right)\right) \otimes f_{p_k}, \forall p_k \in \mathcal{N}\left(p_j^{\text{cen}}\right)\right\}\right). \tag{9}$$

注意，基于 RS-CNN 的特征学习层不同于 mlp。如果将 mlp 简单地用作特征学习层，则该过程可以表述为:

$$f'_{p_j^{\text{cen}}} = \mathcal{A}\left(MLPS\left(f_{p_k}\right), \forall p_k \in \mathcal{N}\left(p_j^{\text{cen}}\right)\right) \tag{10}$$

公式（9）和公式（10）的主要区别在于，公式（10）没有引入低层关系先验向量 $h\left(p_j^{\text{cen}}, p_k\right)$，它为 $\mathcal{M}$ 提供了学习卷积权值的几何先验。

3.3 自注意力机制

基于 transformer 在 NLP 领域的卓越表现，我们考虑利用 Transformer 固有的顺序不变性，避免需要定义点云数据的顺序，通过注意机制进行特征学习。

首先将输入坐标嵌入到新的特征空间中。然后将嵌入的特征输入到 4 个堆叠的注意模块中，学习每个点的语义丰富和有区别的表示，然后用线性层生成输出特征，将其与 RS-CNN 拼接在一起，具体如图 3所示

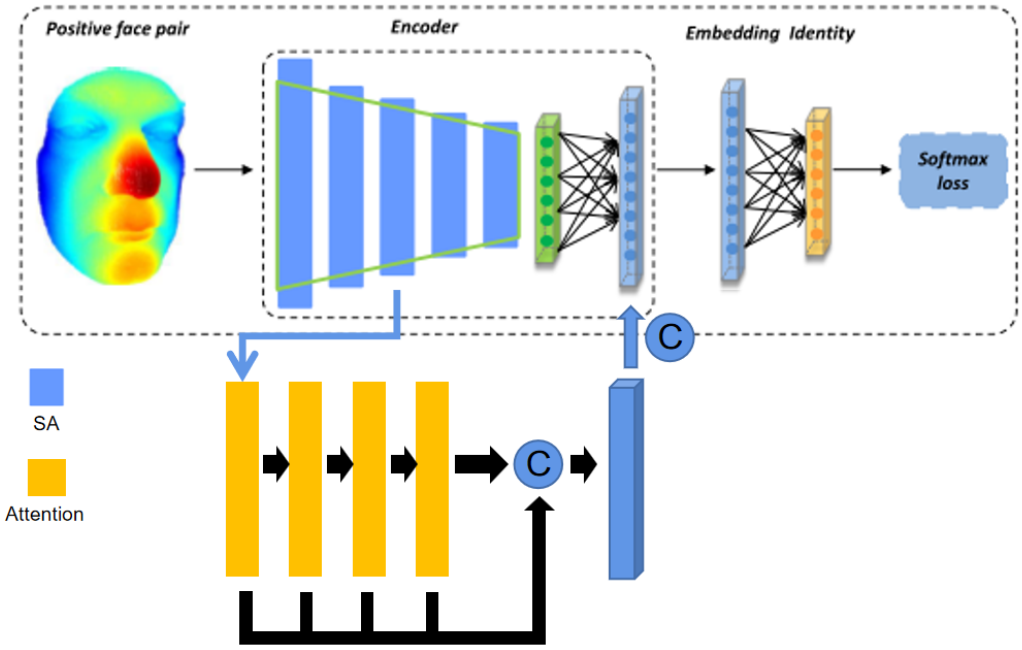

图 3: 自注意力模块

其中注意力层的具体结构如图 4所示

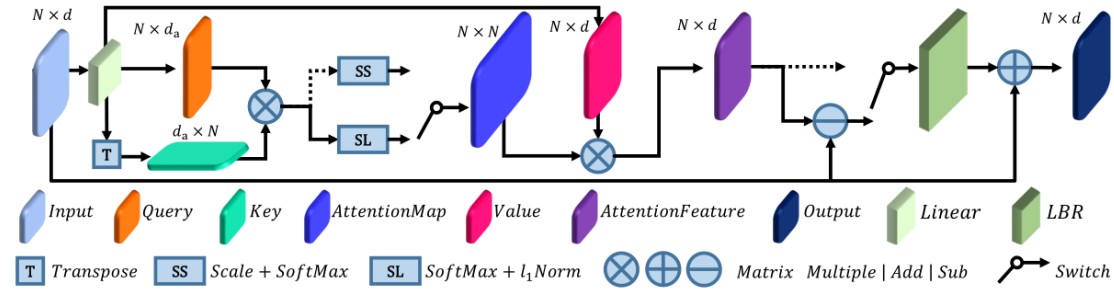

图 4: 自注意力结构图

3.4 特征相似度损失

为了将正样本对从负样本中分离出来，受三元组和对比学习的启发，我们设计了特征相似度损失来监督编码器的训练。我们的目标是最小化来自相同个体的人脸之间的相似性，并在高维特征空间中同时最大化不同个体人脸之间的差异。具体地说，给定 N_s 个 emb_i，emb_i^+ 和 emb_i^- (i = 1,2，…，N_s) 分别作为锚点、正样本和负样本的 L2 归一化嵌入，我们的特征相似度损失可以描述为:

$$L_{sim} = \sum_{i=1}^{N_s} \left[\mathcal{D}\left(emb_i, emb_i^+\right) + m - \mathcal{D}\left(emb_i, emb_i^-\right) \right] \tag{11}$$

其中 m 表示边缘，D(•，•) 表示计算两个向量之间距离的函数。与一般的三元组损失不同，我们使用余弦距离而不是 l2 距离来评估两个向量之间的相似性。原因是属于同一个体的不同样本在高维特征空间中具有相同的角度，余弦距离作为损失函数可以指导编码器实现这一目标，。由于余弦函数单调减小 (当角度在 [0，π/2] 范围内)，我们将公式（11）中的 L_sim 重新表示为:

$$L_{sim} = \sum_{i=1}^{N_s} \left[1 - \mathcal{D}_{\cos}\left(emb_i, emb_i^+\right) + \mathcal{D}_{\cos}\left(emb_i, emb_i^-\right) - m \right] \tag{12}$$

其中 D_cos(•，•) 表示两个特征向量之间的余弦距离。

结合特征相似度损失和 softmax 分类损失 $L_{softmax}$, 总损失可以描述为:

$$L_{\text{total}} = L_{\text{softmax}} + \lambda L_{\text{sim}} \tag{13}$$

其中 λ 是用于平衡的一个常数

4 复现细节

4.1 与已有开源代码对比

基于已有的 RS-CNN 模型，我们将其运用于人脸识别中。RS-CNN 原本只用于 ModelNet 等拥有一定差异性的数据集中作分类或分割任务，作为特征提取模块最终获得的特征向量多集中于全局特征，当我们将其运用于人脸识别中时，我们更加关注于其获取的局部特征，因此我们对 sampling、grouping 和集合抽象模块进行了修改，同时加入三元组损失与对比学习共同监督整个训练过程。

在实验的后期，我们还根据 transformer 的启发在模型中加入了自注意力机制，使得模型在特征提取的过程中更加关注局部信息，将嵌入的特征输入到 4 个堆叠的注意模块中，学习每个点的语义丰富和有区别的表示，然后用线性层生成输出特征，将其与 RS-CNN 拼接在一起。

4.2 实验环境搭建

系统版本为 Ubuntu 14.04，编译器使用 Python 3.7，Pytorch1.6.0，CUDA 10.2 + cuDNN 8.2

4.3 使用说明

配置文件保存在 cfgs 中，可以根据使用者自己的需求修改配置文件，运行 train_face_siamese.py 训练网络，模型文件会保存在 cls 中。运行 evaluate_face_id_all.py 评估结果，评估日志文件将保存在对应模型文件夹的 log 中。

4.4 创新点

• 我们提出了一种轻量级但有效的框架，用于直接从 3D 点云中提取 3D FR 的特征。在训练阶段，使用两个权重共享编码器从人脸对中提取特征，然后学习样本之间的特征差异。在评估阶段，利用编码器获取 FR 的特征向量进行相似性评估。

• 为了有效地测量样本之间的特征相似度并获得细粒度的面部表示，受对比学习的启发，我们提出了一种对选择策略来促进训练，并设计了一个特征相似度损失来监督编码器的训练。

• 基于 transformer 在 NLP 领域的卓越表现，我们考虑利用 Transformer 固有的顺序不变性，避免需要定义点云数据的顺序，通过注意机制进行特征学习，然后将嵌入的特征输入到注意模块中，学习每个点的语义丰富和有区别的表示，然后用线性层生成输出特征，将其与 RS-CNN 特征拼接在一起。

5 实验结果分析

Lock3DFace[3]是由 Kinect V2 收集的低质量 3D 人脸组成的综合数据库，其中包括 509 个人的 5671 个 RGB-D 视频片段。数据集分为五个子集，涵盖表情变化 (FE)、中性脸 (NU)、遮挡 (OC)、姿势 (PS) 和时间推移 (TM)。点云格式的面部扫描样本如图 5所示。我们可以看到 Lock3DFace 的面部数据中有很多噪声，一些面部区域是模糊的。虽然数据集包含 RGB 信息，但我们在实验中只对所有模型使用深度信息。

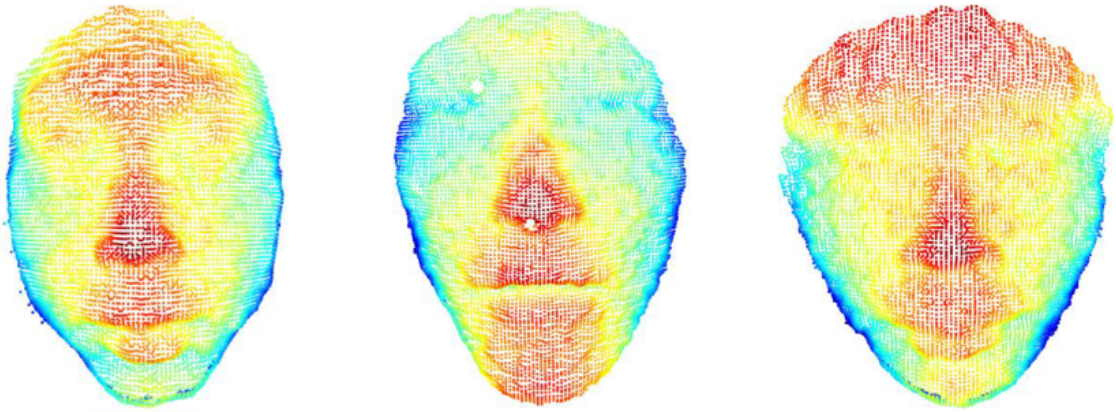

图 5: 面部扫描图

为了对 Lock3DFace 进行消融研究、识别和验证。首先，我们随机选择 340 名受试者进行训练，其余 169 名受试者用于测试。在训练集中，我们从每个视频中以相等的间隔采样 6 帧。在测试集中，我们也为每个受试者提取 6 帧，并将每个表情中立的人脸视频中的第一帧作为图库样本 (结果为 169 个图库样本)，其余帧用作探针。其次，对于训练集和测试集，我们采用基于鼻尖的对齐对所有样本进行预处理，进行非面部区域去除和面部裁剪。经过预处理后，所有数据都是点云格式，可以直接馈送到我们的网络中。对于二维 CNN 对应点的训练，我们按照从预处理的点云中获得精细化的深度图。在评估阶段，我们提取嵌入来自我们的网络结构编码器和其他 2D cnn 编码器的探测和画廊。然后，我们将每个探针的特征与图库中的所有身份进行匹配。探针的身份将被认为是具有最小余弦距离的画廊个体，最终评估结果如下所示

表 1: Lock3DFace 中 Rank-one 识别率

Method	Input	Lock3Dface					
		FE	NU	OC	PS	TM	Total
VGG-16	Depth	94.76	99.57	44.68	49.21	34.50	70.58
ResNet-34	Depth	96.09	99.29	54.91	61.39	45.00	76.56
Inception-v3	Depth	93.56	98.97	56.98	54.14	42.17	74.44
MobileNet-v2	Depth	95.74	98.91	61.44	69.92	43.00	79.49
Led3D	Depth	97.62	99.62	68.93	64.81	64.97	81.02
	Depth&Normal	98.17	99.62	78.10	70.38	65.28	84.22
RS-CNN	XYZ	97.93	99.35	77.06	72.03	66.33	84.78
	XYZ&Normal	98.52	99.46	80.67	73.69	67.00	87.18
Ours	XYZ	98.36	99.03	76.59	71.97	66.59	84.34
	XYZ&Normal	98.94	99.62	76.50	72.48	64.37	85.96
Ours(attention)	XYZ&Normal	98.63	99.40	79.16	75.92	70.66	87.56

6 总结与展望

本文在 PointNet++ 框架的基础上复现并加入了 RS-CNN（Relation-Shape Convolutional Neural Network）用于特征提取，用于直接处理 3D 人脸的点集数据。为了进一步提高 FR 性能，我们从 transformer 中得到灵感，在其中加入了自注意力机制，在 Lock3DFace 上的实验表明，所提出的网络模型可以达到较好的性能。

未来可能的方向有高质量的点云合成网络，以便进一步提高使用低成本的 3D 硬件 (如 Kinect) 捕获的 3D 面部数据的识别精度，或者对特征提取模块再进行修改以取得更好的验证精度。

参考文献

[1] PHILLIPS P J, FLYNN P J, SCRUGGS T, et al. Overview of the face recognition grand challenge[C]//2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05): vol. 1. 2005: 947-954.

[2] SAVRAN A, ALYÜZ N, DIBEKLIOĞLU H, et al. Bosphorus database for 3D face analysis[C]//European workshop on biometrics and identity management. 2008: 47-56.

[3] ZHANG J, HUANG D, WANG Y, et al. Lock3dface: A large-scale database of low-cost kinect 3d faces [C]//2016 International Conference on Biometrics (ICB). 2016: 1-8.

[4] DRIRA H, AMOR B B, SRIVASTAVA A, et al. 3D face recognition under expressions, occlusions, and pose variations[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(9): 2270-2283.

[5] SAMIR C, SRIVASTAVA A, DAOUDI M. Three-dimensional face recognition using shapes of facial curves[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(11): 1858-1863.

[6] MIAN A S, BENNAMOUN M, OWENS R. Keypoint detection and local feature matching for textured 3D face recognition[J]. International Journal of Computer Vision, 2008, 79(1): 1-12.

[7] GILANI S Z, MIAN A, SHAFAIT F, et al. Dense 3D face correspondence[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 40(7): 1584-1598.

[8] KIM D, HERNANDEZ M, CHOI J, et al. Deep 3D face identification[C]//2017 IEEE international joint conference on biometrics (IJCB). 2017: 133-142.

[9] GILANI S Z, MIAN A. Learning from millions of 3D scans for large-scale 3D face recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 1896-1905.

[10] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition [J]. arXiv preprint arXiv:1409.1556, 2014.

[11] MU G, HUANG D, HU G, et al. Led3d: A lightweight and efficient deep approach to recognizing low-quality 3d faces[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 5773-5782.

[12] MATURANA D, SCHERER S. Voxnet: A 3d convolutional neural network for real-time object recognition[C]//2015 IEEE/RSJ international conference on intelligent robots and systems (IROS). 2015: 922-928.

[13] QI C R, SU H, NIESSNER M, et al. Volumetric and multi-view cnns for object classification on 3d data [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 5648-5656.

[14] QI C R, SU H, MO K, et al. Pointnet: Deep learning on point sets for 3d classification and segmentation [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 652-660.

[15] QI C R, YI L, SU H, et al. Pointnet++: Deep hierarchical feature learning on point sets in a metric space [J]. Advances in neural information processing systems, 2017, 30.

[16] WANG Y, SUN Y, LIU Z, et al. Dynamic graph cnn for learning on point clouds[J]. Acm Transactions On Graphics (tog), 2019, 38(5): 1-12.

[17] ZHOU H, FENG Y, FANG M, et al. Adaptive graph convolution for point cloud analysis[C]// Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 4965-4974.