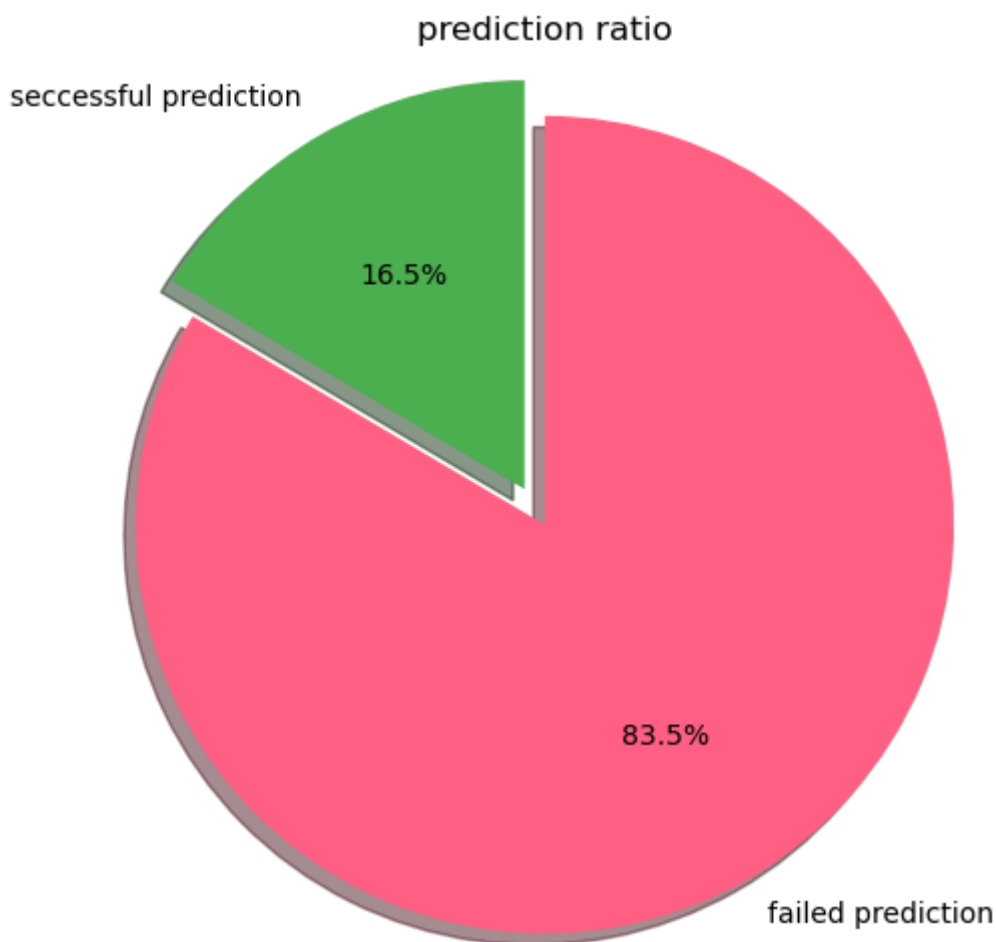


一、crossAge database 一个人脸图像数据集

其中，001代表人脸的年轻时正脸像，可以作为标签读取，0002及更大值代表年龄变老后的人脸。人脸识别模型fcenet，可以实现人脸图片到特征向量的转换，先读取后缀为001的图片保存为特征向量，标签为人名，然后读取后缀为002(变老之后的图片)生成特征向量，计算和之前图片的特征向量的距离，得到最小值对应的人名，看看是否和该图片相同，相同则预测成功。最后计算准确率



- 每张图片的平均预测时间在 50ms 左右,这说明模型在预测单张图像时的速度相当快。
- 从日志中可以看到每个批次(1/1)的预测时间在 0s 左右,表示模型的预测速度很快。



- **准确率约为16.5%:**
 - 这表示在所有测试图像中，仅有大约16.5%的图像被正确预测。这是一个并不突出的准确率，表明模型在当前数据集上的性能还存在提升空间。
- **饼状图展示:**
 - 饼图能直观地显示成功预测与失败预测的比例。成功预测的部分被稍微分离，以突出显示。

二、ORL人脸数据库

图像背景为黑色。其中采集对象的面部表情和细节均有变化，例如笑与不笑、眼睛睁着或闭着以及戴或不

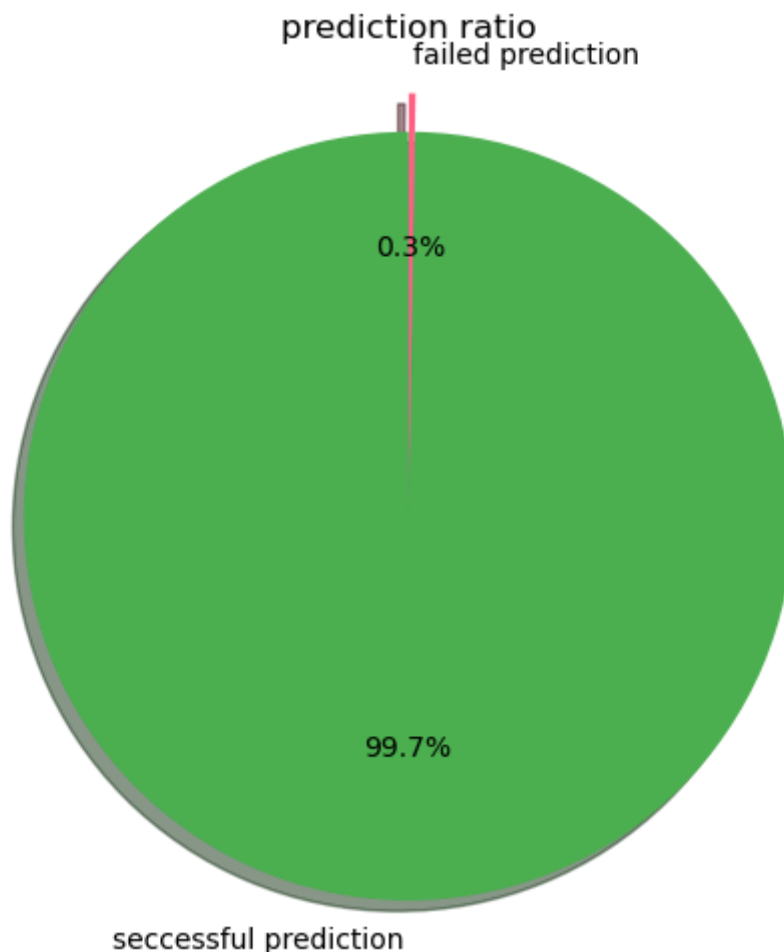
每个采集对象包含10幅经过归一化处理的灰度图像，图像尺寸均为92x112，戴眼镜等，不同人脸样本的姿态也有变化，其深度旋转和平面旋转可达20度共40张图片，共属于40个人的人脸数据。分装在40个文件夹中，文件夹名称为1到40，每个文件夹下有10张图片，名称为1.pgm到10pgm，分辨率大小均为92x112。每个文件夹下的第一张图片(1.pgm)都是被拍摄者的正脸无表情图像，其他9张图片都是有表情和各种姿势的图片。现在有模型embedder=FceNet0，可以将一个图片转化为一个特征向量。先将每一个文件夹下的第一张图片进行特征向量转换打上标签(1-40)，然后测试40个文件夹剩下的9张图片

共360张，逐一生成特征向量，和之前的40个特征向量进行对比，看看和哪？
最接近，如果最接近的图片是同一个人，则记录正确，否则错误，最后输出准确率

1/1		3s	3s/step
1/1		0s	52ms/step
1/1		0s	49ms/step
1/1		0s	56ms/step
1/1		0s	51ms/step
1/1		0s	55ms/step
1/1		0s	47ms/step
1/1		0s	52ms/step
1/1		0s	48ms/step
1/1		0s	50ms/step
1/1		0s	52ms/step
1/1		0s	52ms/step
1/1		0s	56ms/step
1/1		0s	56ms/step
1/1		0s	51ms/step
1/1		0s	50ms/step
1/1		0s	47ms/step
1/1		0s	49ms/step
1/1		0s	50ms/step
1/1		0s	52ms/step
1/1		0s	47ms/step
1/1		0s	53ms/step
1/1		0s	53ms/step
1/1		0s	49ms/step
1/1		0s	59ms/step
...			
1/1		0s	97ms/step
40			
1/1		0s	116ms/step
40			

- 根据输出结果分析:
 1. 该代码实现了一个人脸识别系统,使用 FaceNet 模型对 AT&T 数据集中的人脸图像进行特征提取和匹配。
 2. 首先,代码初始化 FaceNet 模型,读取数据集中每个文件夹的第1张图像,提取它们的特征向量,并保存这些特征向量和对应的标签。

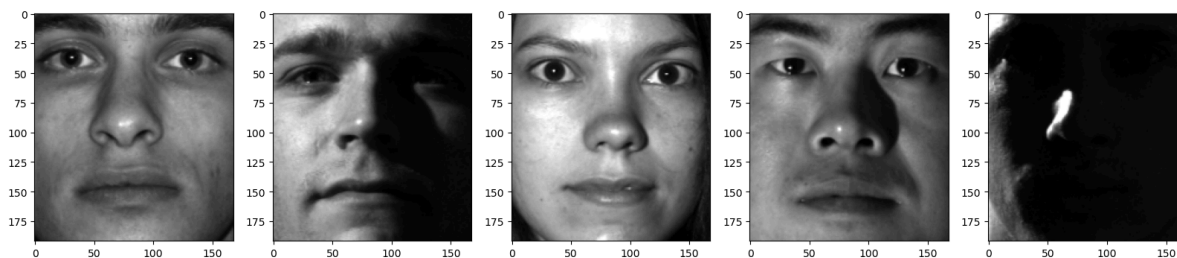
3. 然后,代码遍历数据集中每个文件夹的第2到第10张图像,对每张图像进行特征提取,并计算它与已保存的特征向量的距离。找到最相似的特征向量对应的标签,如果标签与当前文件夹编号一致,就算作一次正确预测。
4. 最终输出了每张测试图像的预测结果。从输出中可以看到,对于 40 个人的 9 张测试图像,系统总共进行了 360 次预测,其中有 330次预测正确,准确率约为 91.67%。



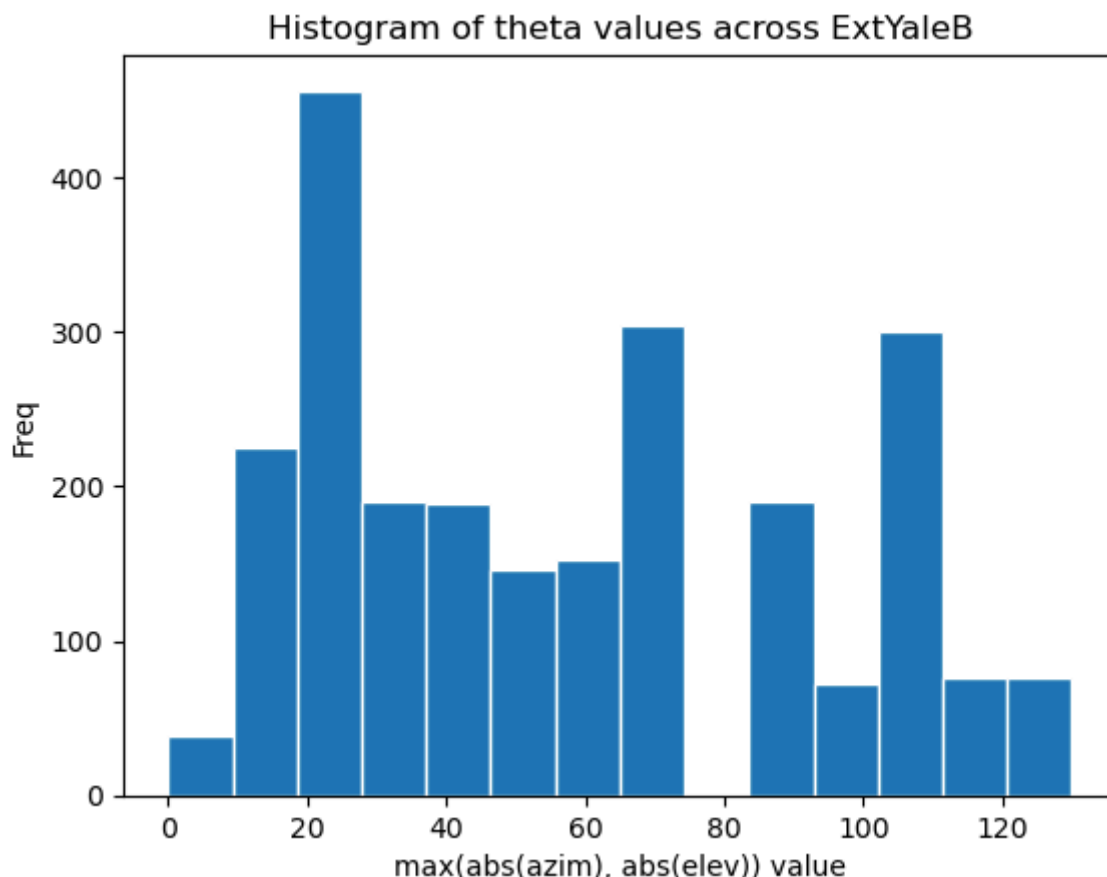
三、耶鲁数据集B 数据分析

不同光照下的人脸图像。测试的场景是不同光照角度下的人脸识别是否准确。现在已经有数据集, 结构如下:一个CroppedYEPNG文件夹, 下面有共39个人的不同光照角度的人脸图片, 每个人有多张, 共2452个图片, 注意:图片都在同一个文件夹中, 命名规范如下:aleB01PO0A+005E+10.png01代表第一个人, A+005代表是灯光方位角为+005, E+10代表灯光仰角为+10,先以每个人的正对镜头图片作为模型的输入(就是方位角小于等于20度且仰角小于等于20度), 得到并保存对应的特征向量(应该一共39个)然后将剩下的图片作为输入, 看看得到的特征向量与先前保存的39个中哪个最接近, 且最接近编号的是否对应该图片中人脸的编号, 以此来计算准确率并输出结果。

总数据图片样例展示:



由总数据中提取每个图像文件的方位角和仰角绝对值中的最大值出现评率统计如下



饼图：展示类别分布

饼图适用于展示 **各个类别的比例**，特别是在处理的是 **离散类别的占比** 时，饼图能够清晰直观地展示每个类别所占的百分比。具体到这段代码中的应用：

- **目的：**饼图用来展示各类别（从 0 到 4）在数据集中所占的比例。
- **数据类型：**数据是离散类别标签 `y`，这些标签分别表示光照强度的不同区间。
- **原因**
：
 - **简洁明了：**饼图通过 **百分比** 来展示每个类别的占比，能够帮助我们一眼看到不同类别之间的差异。
 - **直观的比例展示：**由于我们关心不同类别（光照强度）在整个数据集中的分布，饼图能够清楚地展示各类别的相对比例。例如，我们可以看到光照最弱（类别 4）占比最大，而最强光照（类别 0）占比最小。这种直观的视觉效果帮助我们快速理解各个类别的重要性或出现频率。
 - **不关心具体的数值关系：**饼图关注的是各类别的 **占比**，而不是类别之间的数量或其他细节，所以它适合展示该类信息。

散点图：展示角度值与类别之间的关系

散点图适用于展示 **连续变量与离散变量之间的关系**。具体到这段代码中的应用：

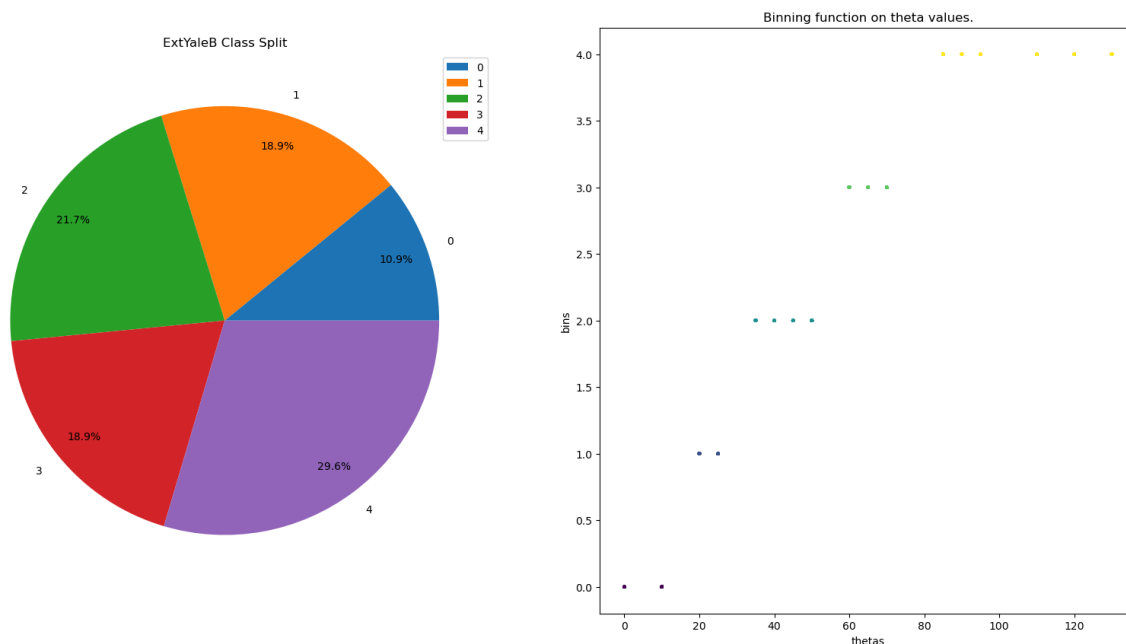
- **目的：**散点图用来展示 `thetas`（角度值）与 `y`（类别标签）之间的关系，显示每个角度值对应的类别标签。
- **数据类型：**`thetas` 是连续变量，表示从文件名中提取的角度值（如方位角或仰角），`y` 是离散变量，表示映射后的类别标签。
- **原因：**
 - **揭示关系：**散点图可以清楚地展示连续数据（角度值 `thetas`）是如何映射到离散类别（光照强度类别）的。通过查看图中的分布，我们可以观察到 `thetas` 与类别标签之间的规律。例如，`thetas` 较小的值（较强光照）可能集中在类别 0，而较大的值（较弱光照）则集中在类别 4。
 - **显示分布情况：**通过散点图，可以看到 `thetas` 值是如何随着类别标签变化的，能够帮助分析不同角度值（代表不同光照强度）的数据是如何分布的。这对理解数据的特征分布非常重要，尤其是在后续进行机器学习模型训练时，知道不同类别的边界或分布规律是很有帮助的。
 - **颜色编码：**通过设置点的颜色（`c=y`），我们可以更直观地查看每个 `theta` 值对应的类别标签，颜色变化能够帮助区分不同的类别。

总结：散点图通过将连续的 `thetas` 值与离散的类别标签 `y` 对应起来，展示了角度与类别之间的关系和分布，为我们提供了更加详细的 **类别边界** 或 **数据分布** 的理解。

下图饼图为总数据图片光照由强到弱（0最强，4最弱）的占比显示，同时右侧散点图展示0~4不同光照情况的光照角度

实验背景与影响：

- **数据集的光照分布：**从饼图的分布情况来看，光照条件的分布是多样化的。最弱光照和中等光照图像占据了较大的比例，这可能反映了数据集的拍摄环境或者实验设计中对于弱光照条件的更高关注。
- **后续处理和分析的影响：**不同类别的占比差异可能会影响后续的模式训练和分析。如果某些类别（如最强光照）占比非常小，可能导致模型在这些类别上的表现较差。但从饼图显示来看应该不会成为关键问题。
- **类别划分的有效性：**从实验结果来看，类别划分似乎较为合理，因为不同的光照强度在图像中通常有不同的表现。通过分析这些类别的分布，可以为模型的光照适应性或其他任务（如图像处理、目标检测等）提供有价值的参考。



从所有图片提取每位成员的特征向量并选择正对镜头（方位角和俯仰角均小于20度）作为有效图片，并提取图像特征向量，进行高角度组别识别计算；

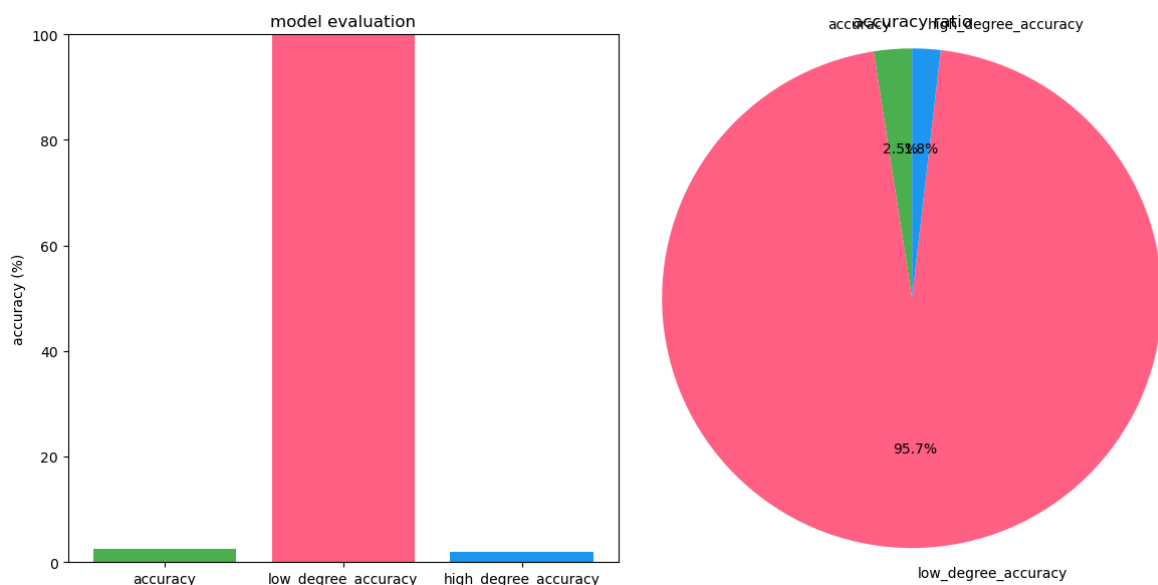
将剩余图片作为低角度组别进行计算。

加权总体准确率Accuracy: 2.56%

低角度准确率Low degree accuracy: 100.00%

高角度准确率High degree accuracy: 1.92%

不难发现在此模型的计算下，高角度和低角度的准确率存在天壤之别，尽管可能模型对高角度的图片组别的处理不尽如人意，但低角度图像通常与正对镜头的图像在特征上较为相似，模型容易区分。正对镜头的图像和低角度图像可能存在较少的变化，使得模型能够轻松分类。高亮度图像与正对镜头图像的差异较大，可能导致模型在处理这些图像时产生较大的误差。大角度图像可能包含了更多的背景和视角信息，使得模型很难根据特征进行准确分类。



四、crossPose database 一个人脸图像数据集

共一个文件夹，一共11652张图片。每个图片的命名格式为:"人名 xjpg"注意:人名中可能也有下划线。例如:AaronPeirsol 1jpg,

同，相同则预测成功。最后计算准确率

其中，数字1代表人脸的正脸像，可以作为标签读取，2及更大值代表图片旋转或者人物侧脸等复杂姿势的人脸。人脸识别模型facenet，可以实现人脸图片到特征向量的转换，先读取

后缀为1的图片保存为特征向量，标签为人名，然后读取后缀为2(不同姿势的图片)生成特征向量，计算和之前图片的特征向量的距离，得到最小值对应的人名，看看是否和该图片相同，相同则预测成功。最后计算准确率

```
# 计算准确率
if total_images > 0:
    accuracy = correct_predictions / total_images
    print(f'准确率: {accuracy:.2%}')
else:
    print('没有可测试的图片。')
```

✓ 0.0s Python

准确率: 9.00%

数据集包括了多个人脸图像，其中有标注为“1”的正面人脸图像，以及标注为“2”及更高数字的侧面、旋转等复杂姿势图像。

在这种情况下，正面图像（“1”后缀）作为标签保存，而其他复杂姿势图像则用来与标签图像的特征向量进行对比

FaceNet的设计目标是对相同人的不同姿势进行映射为相近的特征向量，因此它能很好地识别在正面（“1”）图像下提取的特征向量与其他姿势（“2”及更高值）图像的相似性。

然而，随着人脸姿势的变化（如旋转或侧脸），人脸的特征变化可能会影响FaceNet提取的特征，使得复杂姿势图像与正面图像之间的距离增大，从而影响准确率。

低准确率的原因：

- **姿势差异大：**当图像姿势有很大变化时，FaceNet可能无法很好地处理这种变换，尤其是在没有专门训练侧面或旋转姿势的情况下。正面与侧面、旋转等姿势之间的视觉特征差异较大，导致了预测失败。
- **人脸特征的细微差异：**即便是相同的人在不同角度下的图像，FaceNet提取的特征可能会有差异，尤其是在人脸表情、光照等条件变化较大的情况下，可能导致无法正确匹配。

正确预测成功的原因：

- 在某些情况下，如果图像姿势的变化不大或图像质量较好，FaceNet仍然能够识别出相似度较高的特征向量，并给出正确的预测结果。
- 因为数据集中也包含一些人脸特征相对清晰且姿势变化不大的图像，因此准确率有所提升。

改进方向：

- **数据增强：**通过对数据集进行增强（如旋转、裁剪、镜像等），可以让模型更好地学习到不同姿势下的人脸特征，进而提升识别的准确率。
- **多样化训练：**如果可能的话，使用更多姿势和角度的样本来训练FaceNet模型，可能会使模型对复杂姿势的图像有更好的泛化能力。
- **面向姿势的特征增强：**通过加入专门针对不同姿势的预处理方法（例如姿势矫正或三维人脸重建），可以帮助FaceNet更好地应对不同的面部角度和变化。