

人工智能原理与方法

技术报告

|  |  |  |  |
| --- | --- | --- | --- |
| **题 目:** | 针对低质量数据的人脸识别 | | |
| **姓 名:** | 马照桅 | **学 号：** | 122106010730 |

2022 年 10 月

# **背景介绍**

随着深度学习的发展，人脸识别算法已经得到了很大的进步。现代算法在高质量数据集上的表现已经可以达到95%以上的准确度。然而，人们一直在回避针对低质量数据集的人脸识别。这种数据集的挑战远高于高质量数据集，因为随着分辨率的下降，图片会无法避免的损失人脸属性。图1展示了高质量和低质量人脸图像的示例。

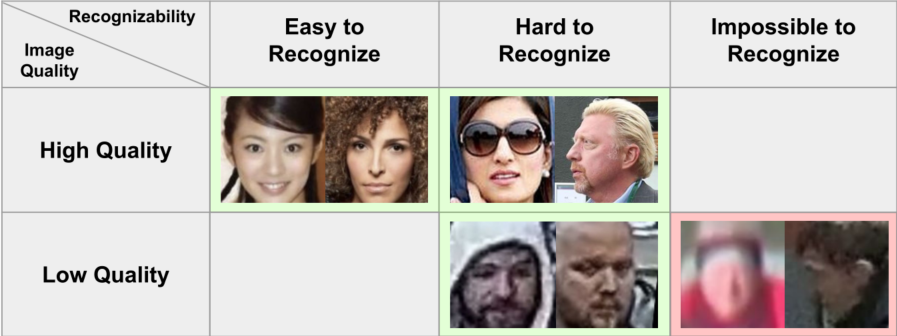


图1.不同质量和不同可识别性的人脸图像示例

如图1所示，高质量和低质量的图像都包含姿势、遮挡和分辨率的变化，这会使人脸识别任务变得困难。像图1的底行的这样的低质量图像正日益成为人脸识别数据集的重要组成部分，因为它们在监控视频和无人机镜头中大量出现。为了降低这种低质量图像的识别难度，本文使用超分辨率方法对数据集进行分辨率提升，进而提高低质量人脸的识别准确度。

# **本文使用的方法**

本文使用超分辨率方法，将原本难以识别的低质量图像进行提高分辨率，尽可能还原出图像中人脸特征，接着参考ArcFace的损失函数和MobileNet的网络结构，提出了一种轻量级的人脸识别模型。该模型的整体结构如图2所示。

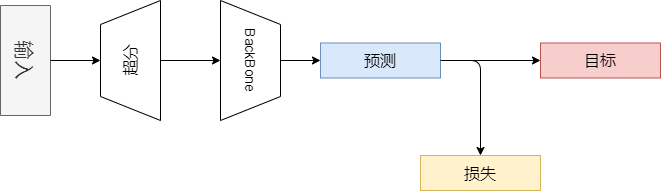


图2.模型结构图

## 2.1超分辨率方法

本文使用的超分辨率方法是RealSR。目前很多先进的超分辨率算法在忽略模糊和噪声的数据集上取得了不错的成绩，但是在解决实际超分辨率问题时表现并不理想，因为这些算法的训练数据集大多采用简单的高质量数据集经过多次下采样来构造低分辨率图片的。使用这种方法的好处是，图像损失了部分高频信息并强化了低频信息，这不但保留了原图像大部分信息，而且低分辨率图像是干净且无噪声的。

为了解决这一现象带来的在真实世界低分辨率图像上效果不理想的问题，RealSR设计了一种新的图像退化框架，通过估计各种模糊核和真实的噪声分布,生成低分辨率图像。基于这一新退化框架，可以获得与真实世界图像共享一个共同域的低分辨率图像。然后，作RealSR提出了一个真实世界的超分辨率模型，以获得更好的感知效果。在合成噪声数据和真实世界图像上的广泛实验表明，这种方法优于最先进的方法，能过获得更低的噪声和更好的视觉质量。

该模型的训练流程如图3所示。算法分为两个阶段：（1）将原图片使用降级算法降低为低分辨率图像；（2）使用降级前的图像作为高分辨率图像，进行监督学习。

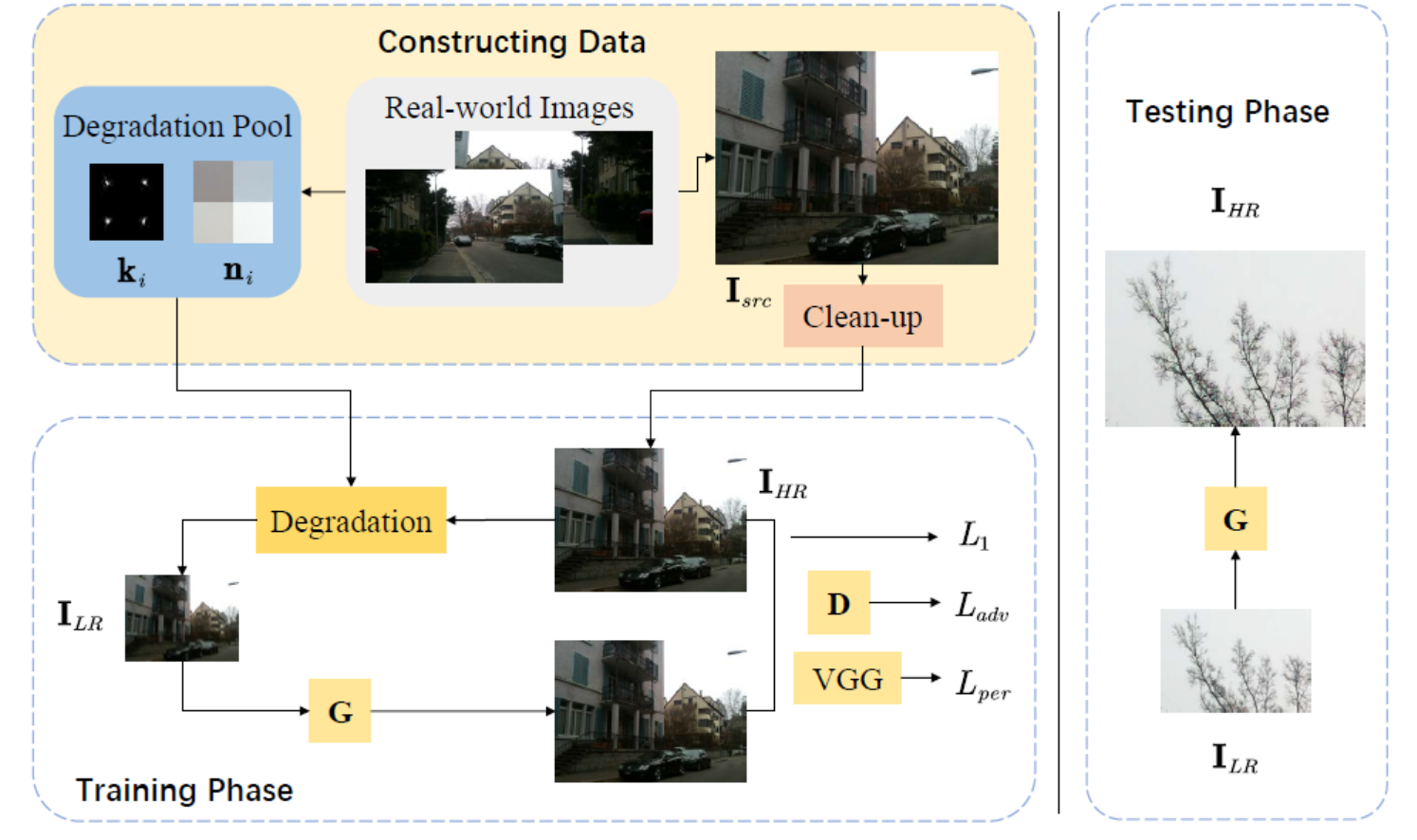


图3.RealSR训练流程

## 2.2人脸识别模型

由于本文旨在解决监控、无人机等条件下的人脸识别，所以本文使用的是基于MobileNet的网络框架，训练的轻量级模型。同时参考了ArcFace的损失函数。

# **实验细节**

## 3.1实验使用的数据集

本次实验使用的训练数据集是emore数据集，共有85742个类别，共5822653张图片。测试数据集使用的是lfw-align-128数据集。数据集结构如图4所示。

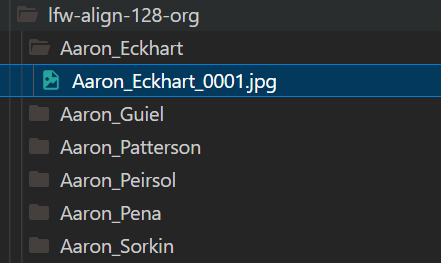


图4.数据集结构

## 3.2人脸检测方法

本文使用的人脸检测算法是MTCNN模型，使用该模型可以将待检测图像中的人脸首先检测出来，从而提高人脸识别的效率。

## 3.3测试方法

为了验证超分辨率方法的有效性，本次实验对测试数据集进行了手动降低图像分辨率和质量的操作，以及对降低后的图像进行超分辨率的操作。得到三个测试数据集：原图像数据集，图像分辨率为128\*128；降低质量后的数据集，分辨率为32\*32；对低分辨率图像进行超分辨率后的数据集，分辨率为128\*128。

三种数据集的图像示例如图5所示。

|  |  |  |
| --- | --- | --- |
| 原图像 | 低分辨率图像 | 超分辨率图像 |

图5.三种数据集图像展示

# **实验结果与分析**

## 4.1实验结果

模型在原测试数据集上的表现如图6所示。



图6.原测试数据集上的表现

模型在低质量测试数据集上的表现如图7所示。



图7.低质量测试数据集上的表现

模型在超分辨率测试数据集上的表现如图8所示。



图8.超分辨率测试数据集上的表现

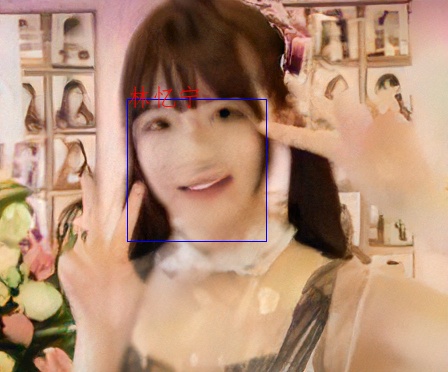
模型的实际使用效果如下所示。原图分辨率为60\*64，图像质量较低。无法检测到人脸，导致识别失败。但是在使用超分辨率技术后，可以成功识别出该人脸信息。



低分辨率原图



低分辨率识别图



超分辨率识别图

## 4.2分析

从结果上来看，超分辨对于低质量数据集的人脸识别效果是有提升的。但是在原有算法流程中加入一个额外的模型，会导致整体模型的运行时间加大。在本次实验中，未使用超分辨率模型时的总识别时间为24ms，但是加入超分辨率模型后的总识别时间为35ms。这会影响到实际使用中检测效果。

此外本次实验使用的超分辨率模型并不是专门正对人脸的模型。如果将该模型换为处理人脸的超分辨率模型，整体模型的识别效果因该会进一步增加。

# **程序说明文档**

dataset文件夹用于存放训练，测试用数据集。

face\_db文件夹用于存放识别时的人脸数据

models文件夹用于存放训练好的模型

creat\_dataset.py程序用于将数据集解压并转换为二进制文件

infer.py程序用于进行人脸识别

eval.py程序用于评估模型在数据集上的表现

train,py程序用于训练模型

gan.py程序用于实现超分辨率