# 面部表情识别项目文档

## 一、项目概述

本项目是一个基于深度学习的面部表情识别系统，通过卷积神经网络（CNN）对人脸图像进行分析，可识别 7 种基本情绪（愤怒、厌恶、恐惧、快乐、悲伤、惊讶、中性）。系统包含数据处理、模型构建、训练、评估及实时识别等模块，实现了从数据预处理到实际应用的完整流程。

## 二、文件说明

### 1. 核心功能文件

| **文件名** | **功能描述** |
| --- | --- |
| preprocess.py | 数据预处理模块。读取fer2013.csv数据集，将像素值转换为 48x48 的灰度图像矩阵并归一化，对标签进行 one-hot 编码，按Usage列划分训练集和测试集。 |
| model.py | 模型构建模块。定义 CNN 模型结构，包含多组卷积层、批归一化层、池化层和 Dropout 层，使用 L2 正则化防止过拟合，输出层采用 softmax 激活函数实现多分类。 |
| train.py | 模型训练模块。加载预处理数据，通过ImageDataGenerator进行数据增强（旋转、缩放、翻转等），使用早停（EarlyStopping）和学习率衰减（ReduceLROnPlateau）优化训练，训练后保存模型至saved\_model目录。 |
| predict.py | 模型评估模块。加载训练好的模型，在测试集上评估准确率并输出结果。 |
| camera.py | 实时识别模块。调用摄像头捕获视频帧，通过 Haar 级联分类器检测人脸，使用训练好的模型预测表情并实时显示结果（含置信度）。 |
| FERDSProcess.py | 数据格式转换模块。将fer2013.csv中的像素数据提取为单独的 JPEG 图像文件，按Usage（如训练集、测试集）和表情类别（如 anger、happy）分类存储，方便可视化或其他模型使用。 |

## 三、核心流程说明

### 1. 数据处理

* preprocess.py和FERDSProcess.py均用于处理fer2013.csv，但用途不同：
  + preprocess.py：为模型训练直接生成可输入的数组格式数据（归一化图像矩阵 + one-hot 标签）。
  + FERDSProcess.py：将 CSV 中的像素数据转换为可视化图像文件。

### 2. 模型构建与训练

* model.py定义的 CNN 模型通过多层卷积提取图像特征，结合正则化和 Dropout 抑制过拟合。
* train.py使用数据增强扩展训练样本多样性，通过回调函数动态调整训练策略，提升模型泛化能力。

### 3. 应用场景

* 离线评估：通过predict.py验证模型在测试集上的性能。
* 实时识别：通过camera.py实现摄像头实时表情检测，可用于交互系统、情绪分析等场景。

## 四、使用说明

### 1. 环境依赖

需安装以下库：

tensorflow

pandas

numpy

opencv-python

pillow

### 2. 数据准备

将fer2013.csv数据集放在项目根目录，数据集的下载地址为：<https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data>。

### 3. 操作步骤

1. ****数据转换（可选）****：运行FERDSProcess.py生成图像文件，命令：python FERDSProcess.py。
2. ****模型训练****：运行train.py，训练完成后模型保存至saved\_model/emotion\_model.keras，命令：python train.py。
3. ****模型评估****：运行predict.py查看测试集准确率，命令：python predict.py。
4. ****实时识别****：运行camera.py启动摄像头，按q退出，命令：python camera.py。

## 五、注意事项

* 确保文件路径正确（如模型、数据集、人脸检测 XML 文件）。
* 训练过程可能需要较长时间，取决于硬件性能。
* 实时识别时，光线充足环境可提升人脸检测和表情识别效果。