===========================================================

CNN\_Recognition.m

这段代码实现了一个基于CNN的交通标志分类器。代码首先使用imageDatastore函数创建一个imageDatastore对象，用于存储训练图像数据，其中 'Database' 参数指定数据集的根目录，'IncludeSubfolders' 和 'LabelSource' 参数分别指定是否包含子文件夹以及标签的来源。

然后通过定义CNN模型的各个层，包括卷积层、批量归一化层、ReLU层和池化层，以及全连接层、softmax层和分类层，来搭建一个分类器。

定义好CNN模型后，代码定义了训练参数，包括学习率、迭代次数和批次大小等，并使用trainNetwork函数开始训练模型。

最后，代码使用imread函数读取一张交通标志图像，并使用训练好的模型进行分类预测，返回标签结果。

该代码实现了一个基本的交通标志分类器，可用于实际的交通标志识别应用。

===========================================================

myGUI.m

这是一个交通标志识别系统的 MATLAB GUI 程序，程序界面包括一个显示原图的图形区域和五个按钮。该程序使用了卷积神经网络（CNN）来进行图像分类，通过加载已训练好的 CNN 模型对输入图像进行分类识别。

具体来说，程序的操作流程如下：

点击“打开待识别图片”按钮，选择要识别的交通标志图像，程序将其显示在原图区域中。

点击“灰度变换”按钮，将原图转换为灰度图像，并在灰度变换结果区域中显示。

点击“滤波结果”按钮，对灰度图像进行高斯滤波，并在滤波结果区域中显示。

点击“二值化”按钮，将滤波结果进行二值化，并在二值化结果区域中显示。

点击“边缘检测结果”按钮，对二值化结果进行边缘检测，并在边缘检测结果区域中显示。

点击“识别标志”按钮，程序将原图缩放为 128x128 大小的图像，然后使用已加载的 CNN 模型对其进行分类识别，并将识别结果显示在程序界面的“识别结果”文本框中。

该程序使用了 MATLAB 的图形用户界面（GUI）工具箱来构建交互式的界面，用户可以通过点击按钮完成对图像的预处理和识别操作。同时，程序还使用了 MATLAB 的深度学习工具箱中的 CNN 神经网络来实现图像分类，使得程序具有较高的分类准确率。

===========================================================