

# 天津大学

## 基于 MATLAB 的实时手写数字识别系统 设计与测试报告



姓 名 \_\_\_\_\_ 薛茗阳 \_\_\_\_\_

学 号 \_\_\_\_\_ 3021004070 \_\_\_\_\_

学 院 \_\_\_\_\_ 医学院 \_\_\_\_\_

专业班级 \_\_\_\_\_ 生物医学工程 \_\_\_\_\_

2024 年 9 月 26 日

## 目 录

第一章 绪论 .....	1
1.1. 研究背景和意义 .....	1
1.2. 最新的机器学习和深度学习模型 .....	1
第二章 需求分析及解决方案 .....	2
2.1. 性能需求 .....	2
2.2. 硬件方案 .....	2
2.3. 软件方案 .....	2
2.4. 解决方案总结 .....	2
第三章 研究方法及数据集 .....	3
3.1. 数据集 .....	3
3.2. 图像采集以及预处理方案 .....	3
3.2.1. 实时图像采集 .....	3
3.2.2. 图像预处理 .....	3
3.3. CNN 网络 .....	4
3.3.1. CNN 网络的构建 .....	4
3.3.2. CNN 模型的训练 .....	5
3.4. 实际识别具体过程及处理策略 .....	6
3.5. 最终用户交互界面展示 .....	6
第四章 成果展示及总结分析 .....	8
4.1. 实际测试结果 .....	8
4.2. 未来改进方案 .....	9
4.3. 总结 .....	9

# 第一章 绪论

## 1.1. 研究背景和意义

随着信息技术的飞速进步，计算机视觉和人工智能正日益深入人们的生活和工作。其中，手写数字识别作为计算机视觉中的重要应用之一，广泛应用于诸如邮政编码识别、银行票据处理、自动化办公和智能设备输入等多个领域。手写数字识别的研究不仅有助于提高效率和准确性，还可推动各行业的智能化进程。

手写数字识别与打印字体识别相比，具有更大的难度和挑战。手写体的多样性、复杂性和个体差异显著增加了识别的难度。因此，开发一个基于 Matlab 的实时手写数字识别系统，能够大大促进相关领域的技术发展。本项目旨在利用 Matlab 的平台优势，通过深度学习模型，实现实时、高效、准确的手写数字识别。

研究手写数字识别系统具有重要的理论和应用意义。在理论上，手写数字识别涉及图像处理、模式识别、机器学习和深度学习等多学科交叉的内容，能够丰富和拓展这些领域的研究成果。在实际应用中，手写数字识别技术能极大地提升各类手写信息处理的效率，减少人工干预，提升系统的自动化水平。本项目基于 Matlab 实现，通过其强大的可视化和算法开发能力，为研究者提供了一套完整的手写数字识别解决方案。

## 1.2. 最新的机器学习和深度学习模型

近年来，机器学习和深度学习技术取得了飞速发展，为手写数字识别提供了更多先进的方法和工具。传统的机器学习方法如支持向量机（SVM）和 k 近邻（K-NN）等曾在手写数字识别中发挥重要作用，但深度学习模型，特别是卷积神经网络（CNN）的应用，使得识别性能得到了显著提升。

卷积神经网络（CNN）通过其对于图像特征的高效提取和表示能力，成为手写数字识别的主流方法。早期的 LeNet-5 模型在 MNIST 数据集上取得了显著的成功，其后续的改进模型如 AlexNet、VGG、ResNet 等在图像识别任务中表现出色。CNN 的层级架构、自适应特征提取和高效的计算性能使其在手写数字识别中具有显著的优势，这也是本项目的主要研究方法。

除了 CNN，递归神经网络（RNN）及其变种如长短时记忆网络（LSTM）在处理时序数据方面显示了其独特优势。尽管 RNN 更多应用于自然语言处理领域，但其变种模型与 CNN 相结合，如 CRNN。而最近，生成对抗网络（GAN）和变分自编码器（VAE）等生成模型在特征提取、数据增强和无监督学习方面取得了重要进展。此外，Transformer 模型及其在视觉任务中的应用（如 ViT）也逐渐引起了广泛关注，其自注意力机制在捕捉全局信息方面具有独特优势。

## 第二章 需求分析及解决方案

### 2.1. 性能需求

项目需要动态连续识别手写字符。出于实际使用体验考虑，动态识别界面的帧率应高于 25Hz，程序识别准确率应足够高，程序应适用于多种不同手写字符环境。

### 2.2. 硬件方案

笔记本电脑自带的前置摄像头的清晰度为高清 HD720P，足以满足手写字符识别的测试需求，用户也可自己接入外置的更为清晰便携的摄像头。

### 2.3. 软件方案

基于 matlab 内置的 AppDesigner 编译器进行编译，一方面 matlab 便于编写多种深度学习算法和图像的预处理算法，另一方面 AppDesigner 便于实现用户交互界面，实现面向对象的程序设计。

### 2.4. 解决方案总结

依赖于笔记本摄像头或其他外部摄像头，实时采集手写字符画面，通过 matlab Appdesigner 编写程序对其进行预处理，再利用以训练好的 CNN 模型对其进行识别判断，最终实时反馈识别结果。

## 第三章 研究方法及数据集

### 3.1. 数据集

本项目使用了 MATLAB 中提供的内置 MNIST 数据集，该数据集包含大量手写数字的灰度图像，用作训练和测试深度学习模型。MNIST 数据集的图像尺寸为  $28 \times 28$  像素，单通道灰度图。训练集包含 60000 个样本，测试集包含 10000 个样本。数据集中的标签被转换为分类标签，以用于模型的训练和评估。

### 3.2. 图像采集以及预处理方案

本项目使用 MATLAB App Designer 设计了一个专门用于图像采集和预处理的应用程序。该应用利用摄像头实时捕获图像，并通过一系列预处理步骤优化输入数据，提升模型的识别精度。

#### 3.2.1. 实时图像采集

通过调用 Matlab 官网工具包中的 webcam 接口，我们首先获取当前系统中可用的摄像头，此时可以选择笔记本电脑自带的摄像头或者用户外接的摄像头。随后，通过 snapshot 函数，实时抓取当前视频流中的帧。为最大化图像处理效率，确保获取的图像数据为 uint8 格式，以提高处理速度。

#### 3.2.2. 图像预处理

图像获取后，应用程序会提取图像的中心正方形区域，确保该区域的尺寸在  $200 \times 200$  像素之内，以便于聚焦关键视野内容。进一步，对该提取区域进行灰度化处理，采用 rgb2gray 函数将彩色图像转换为灰度图，为后续的二值化步骤奠定基础。

接下来，对灰度图像执行二值化，通过 imbinarize 将灰度图转换为二值图，提高了特征的对比度。随后，以二值图像为基础执行形态学操作。其中，图像首先经历了膨胀处理，使用 imdilate 函数增强字符的线宽和完整性，以抵御因笔迹断裂带来的信息损失。

最终，将图像调整为  $28 \times 28$  的标准尺寸，以适应较小输入尺寸的神经网络，并转换为双精度型数据以便进一步的深度学习处理。整合后的图像数据形成了供分类网络使用的标准输入格式。为提高识别的稳定性和鲁棒性，应用程序每秒执行一次识别操作，确保在实时性的要求内保持一定的计算资源利用率。

该预处理流程，结合了图像的空间转换、增强和标准化，为后续分类模型的准确性提供了有力支持。通过合理的设计和 optimization，整个应用流程实现了较高的处理效率和识别准确度。

### 3.3. CNN 网络

项目中设计并实现了一个卷积神经网络（CNN）模型，用于手写字符的分类任务。该模型的设计结合了常规的卷积层、激活层、池化层和全连接层，以实现高效的特征提取和分类。

#### 3.3.1. CNN 网络的构建

首先，网络的输入层被设计为能够接收  $28 \times 28$  的单通道灰度图像格式，完美适配 MNIST 数据集。这样的设计允许进行批量数据处理，从而提升了模型的训练效率。

接下来是卷积层的设计。网络包含三层卷积层，每层分别使用 16、32 和 64 个  $3 \times 3$  的卷积滤波器。卷积层的主要功能是在输入图像上滑动滤波器，以提取局部的空间特征，如边缘和角点信息。这里我们选择 ‘same’ 填充以确保卷积后的输出尺寸保持不变，从而在不损失图像边界信息的情况下提取更深层次的特征。

为提高网络性能，卷积层之后设置了批归一化层。批归一化的目的是对激活值进行标准化处理，这样做能有效地减小因权值初始化差异引起的不利影响，从而加速模型的训练速度，同时还提升了其稳定性。

在批归一化层之后，使用 ReLU 激活函数进行非线性转换。这种激活函数有助于增强网络的非线性表达能力，同时有效地避免了梯度消失的问题，使得信号能更为顺利地通过深度网络。

随后，采用了最大池化层进行下采样处理。这个部分使用  $2 \times 2$  的窗口和步长为 2，通过降低特征图的空间分辨率，减小了计算复杂度，同时保留了重要的特征信息。这种池化策略提高了模型对局部变化和噪声的抵抗能力。

最后，网络通过全连接层将所提取的特征整合，并使用 softmax 层将输出映射为概率分布。通过这个结构，模型能够在多类别分类任务中输出准确的分类结果。

如下为具体 CNN 网络参数：

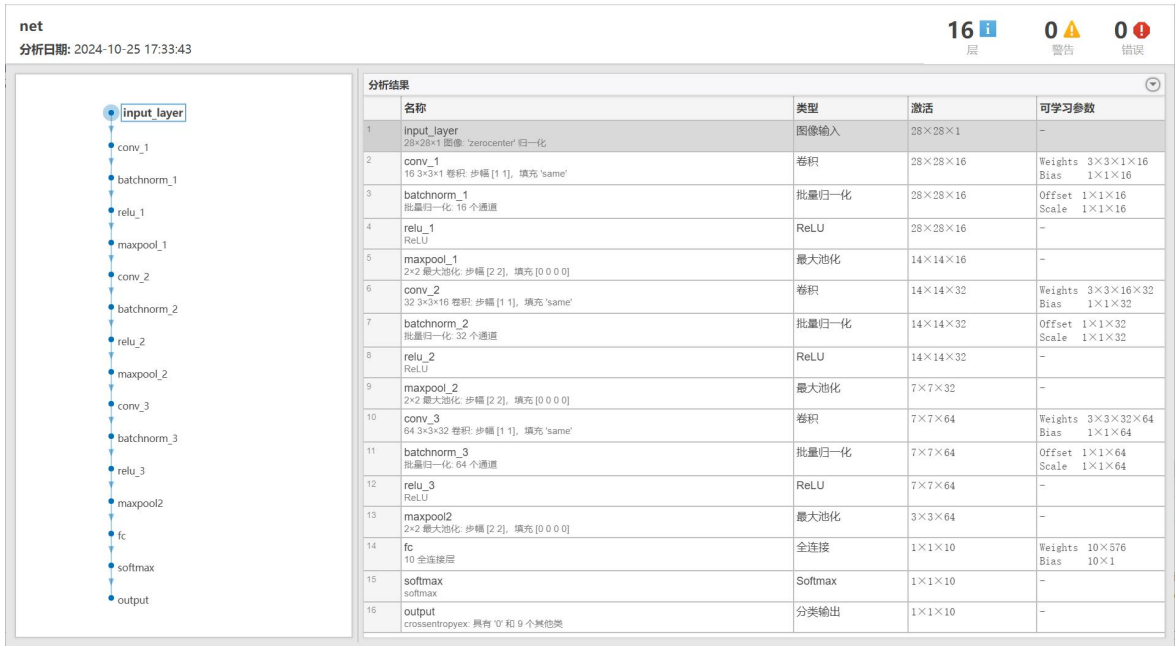


图 1 CNN 网络具体结构以及参数

3.3.2. CNN 模型的训练

训练过程中使用了 Adam 优化器，这种方法结合了动量和自适应学习率调整的特点，可以快速获得良好的收敛性和训练效果。

初始学习率设为  $5 \times 10^{-4}$ ，此选择在保证学习速度的同时，能够有效地稳定训练过程，避免出现震荡或过慢的学习速度。最大训练周期数设定为 20，旨在合理利用计算资源，确保在计算效率和模型性能之间达到最佳平衡。

选择小批量大小为 16，以提升模型的泛化能力和计算效率。每个周期后对数据进行随机化，以减少模型过拟合，并增强网络对不同数据分布的适应能力。在训练过程中，每 30 步进行一次验证，以确保及时监控模型在验证集上的表现，及时发现和调整潜在的过拟合问题。

最终面对 10000 张手写字符图片的测试集样本，模型的预测正确率为 99.56%，满足项目需求，如下为实际预测的混淆矩阵：

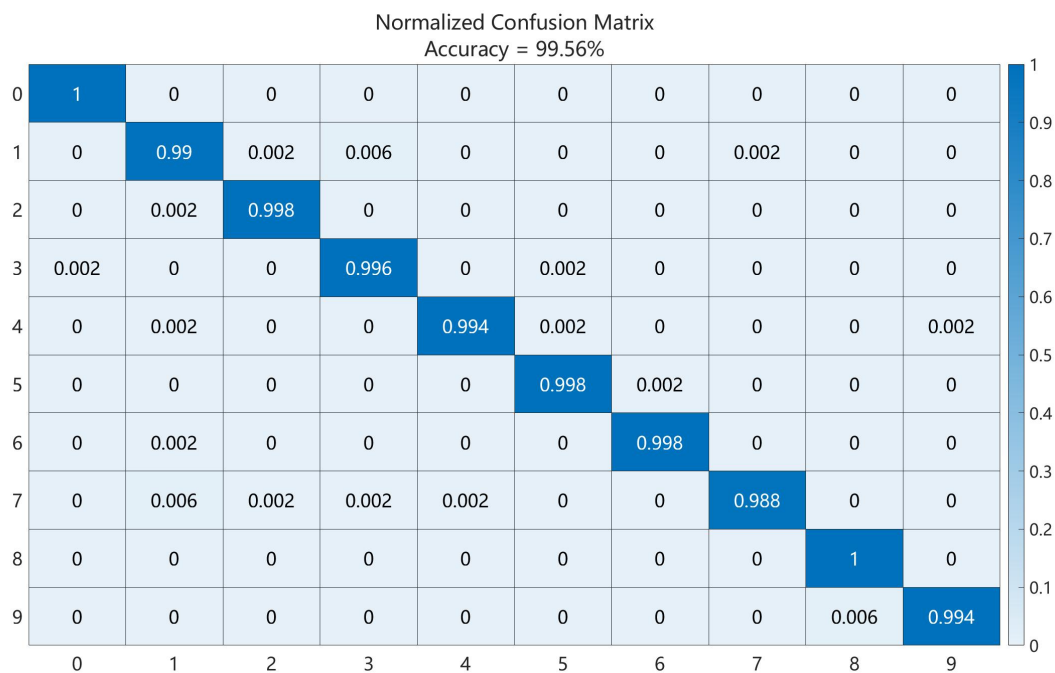


图 2 CNN 模型面对测试集的混淆矩阵

### 3.4. 实际识别具体过程及处理策略

在最终的识别方案中，程序先进行本文 3.2. 中提到的采集以及预处理，随后加入训练好的 CNN 模型进行识别。在实际测试中，发现“1, 4, 7, 9”这几个字符容易混淆，因此实际训练的模型有两个，第一个模型可以识别全部十个数字，第二个模型是专用于识别“1, 4, 7, 9”这四个数字。

当第一个 CNN 模型识别到目标字符属于上述字符其中之一时，通过第二个 CNN 模型进行二次识别，以最后一次识别结果作为最终预测结果，在识别区域的左上角显示预测结果和模型预测的置信度。

### 3.5. 最终用户交互界面展示

用户界面的中心由一大块矩形黑色区域，该区域显示摄像头画面，该区域的左上角显示摄像头的画面帧率，用户点击“打开摄像头”按钮，即可启动摄像头并进行后续识别操作，程序会自动识别目标并给出识别结果。当用户识别结束后，点击“关闭摄像头”按钮即可关闭摄像头。

如下为该界面的展示：



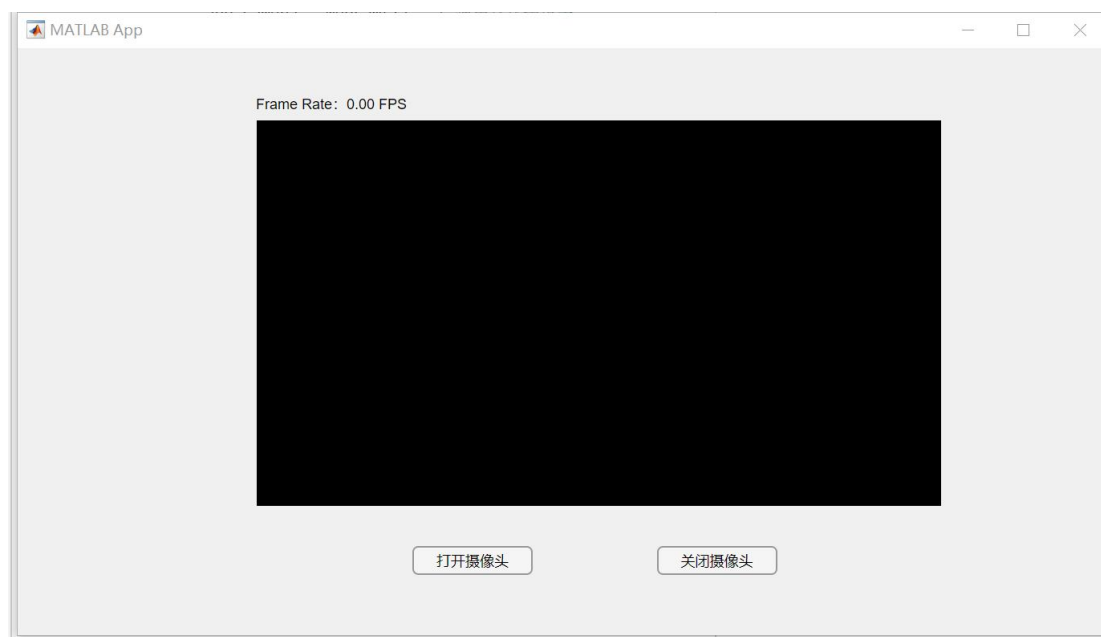


图 3 AppDesigner 用户界面展示

## 第四章 成果展示及总结分析

### 4.1. 实际测试结果

实际测试部分示例如图 4，图 5 和图 6 所示：

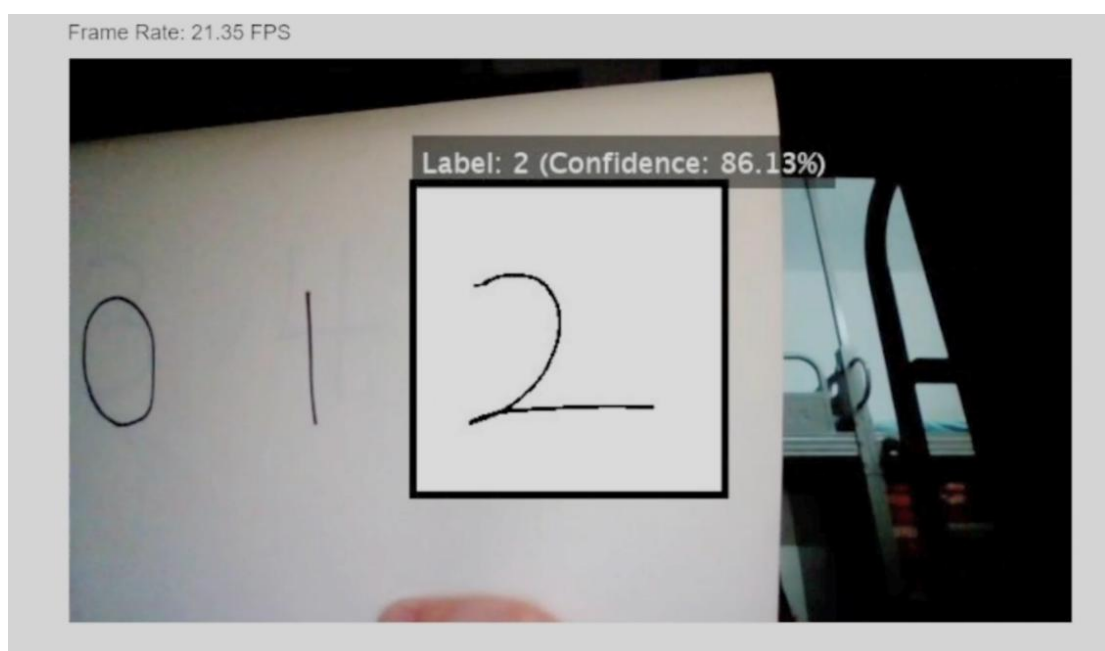


图 4 手写字符“2”的识别测试

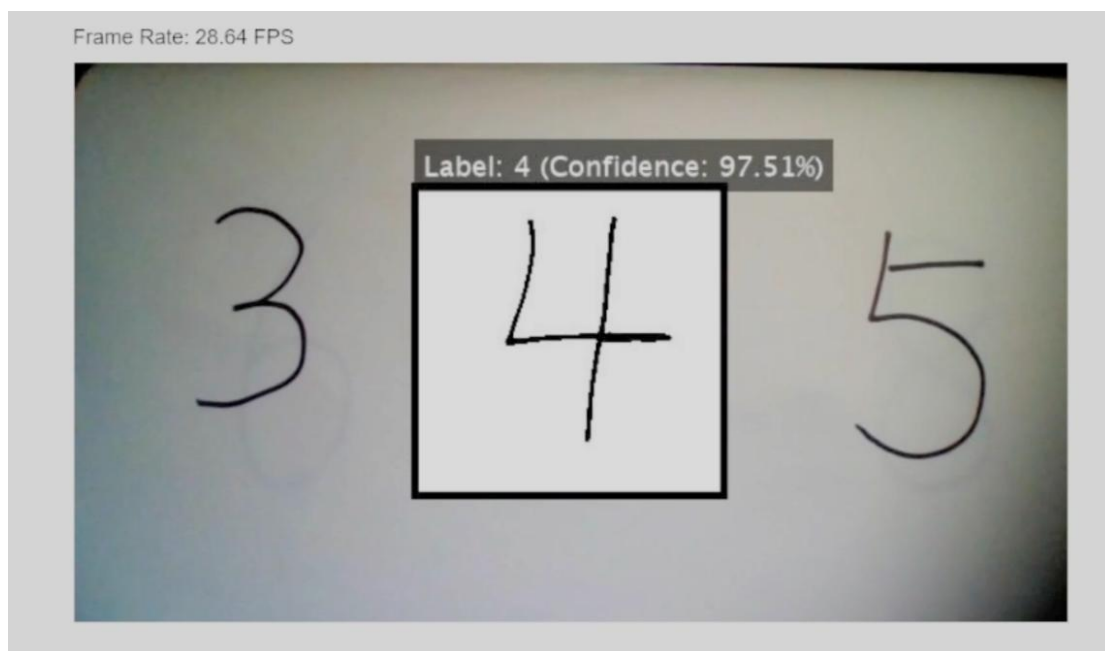


图 5 手写字符“4”的识别测试

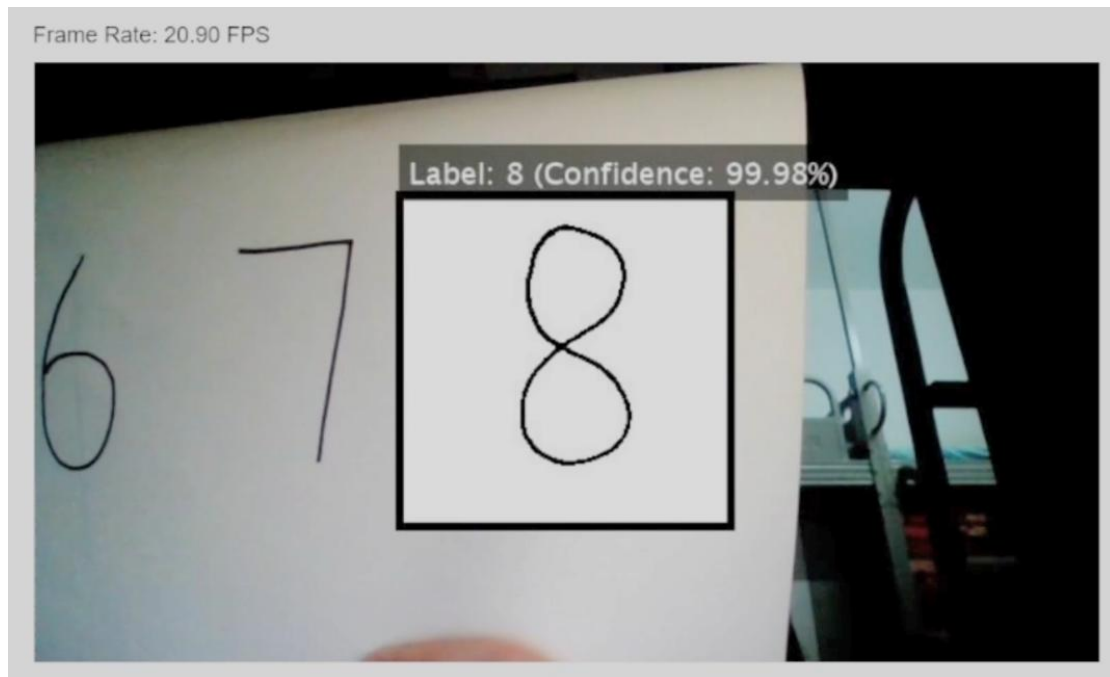


图 6 手写字符“8”的识别测试

最正如上述测试示例一样，用户需要将被测目标移动至摄像头画面所提示的，位于中心的黑色方形框内，在该方框上方会显示出识别的字符以及置信度。更多详细测试信息，请查看附件的视频文件：

“基于 matlab 的实时手写字符识别系统展示.mp4”

摄像头画面清晰，帧率可以保持 20Hz 以上，平均帧率为 25Hz 左右，最大帧率为 30Hz，满足实时连续识别的需求；测试视频中的所有字符均被成功识别，置信度均高于 95%，可以认为该系统具有识别手写字符的功能。

## 4.2. 未来改进方案

首先是识别准确率，当被测目标被部分遮挡或涂抹破坏，CNN 卷积模型的识别准确率会大幅下降，为此可以尝试有别于卷积的递归机器学习办法，如 RNN 等模型，可能会改善这种现象。

若被测目标的光影环境过于暗淡，则会导致画面采集失真或者预处理出错，因此还需进一步优化画面采集和预处理方案，可以尝试使用自带灯光的摄像头进行图像采集。

## 4.3. 总结

本项目成功开发并实现了一套基于 MATLAB 的实时手写字符识别系统。通过综合利用现代深度学习技术和 MATLAB 的 AppDesigner 工具，实现了高效、准确的手写数字识别。项目从图像的实时采集到复杂的预处理，再到最终卷积神经网络的识别，全流程统一且高效。

系统的设计不仅证明了卷积神经网络（CNN）在图像识别任务中的卓越性能，还展示了 MATLAB 在快速构建、测试和部署计算机视觉应用中的便利性。通

过多阶段的图像处理和分层次的识别策略,我们大幅提高了对复杂手写字符的识别准确率,对容易混淆的字符进行多次验证,进一步提升了结果的可靠性。

在实际测试中,系统表现出色,能够在各种环境下以较高的准确率和稳定的帧率识别手写字符。平均置信度高于 95%的识别结果印证了系统的有效性。这一成果不仅为特定字符识别提供了解决方案,还为更加复杂的应用奠定了基础。

展望未来,项目可通过引入更为先进的深度学习模型提升系统在复杂场景下的适应能力。同时,改进图像采集的硬件配置和优化预处理算法也将进一步提高识别的稳定性与准确性。总的来说,本项目的设计满足了项目需求,未来仍有改进空间。