

车牌超分辨率重建与识别

曾超, 陈雨

(四川大学 电子信息学院, 成都 610065)

摘要: 为了从图片中快速准确地识别车牌, 提出一种结合图像超分辨率技术车牌识别方案; 车牌图片具有明显的特定的模式特征, 只是具体的字符编码不同; 因此车牌图片非常适合做超分辨率重建; 文章提出的系统主要由车牌检测定位、车牌超分辨率重建、字符分割、字符识别等模块组成; 综合基于边缘、基于颜色和基于最大稳定极值区域三种车牌检测策略并采用并行编程方法来综合检测结果得到候选车牌; 采用车牌图片正负样本训练支持向量机分类器; 得到分类器模型后对候选车牌判决得到真正的车牌; 随后对真实车牌图片进行超分辨率重建; 该部分主要由基于固定邻域回归的方法实现; 这种方法综合了稀疏字典学习和领域嵌入的方法, 比较好的兼顾了准确率和计算速度; 运用 OpenCV 提供的图像处理库来对重建后的图片做字符分割; 得到单独的字符图片后采用神经网络进行识别; 识别前先使用一定数量的字符图片对网络进行有监督训练获取识别模型; 采用一个单隐层的神经网络, 运用反向传播算法进行训练得到识别模型; 最后提取字符图片的特征并输入网络进行分类完成识别; 为了测试系统的表现, 在实际场景中采集了一百张车牌图片作为测试集; 实验表明, 该系统具有较高识别准确率和较快的处理速度。

关键词: 车牌识别; 超分辨率重建; OpenCV 库; 固定邻域回归; 支持向量机; 人工神经网络

License Plate Super-Resolution Reconstruction and Recognition

Zeng Chao, Chen Yu

(College of Electronics and Information Engineering, Sichuan University, Chengdu 610065, China)

Abstract: On purpose of identifying the license plate quickly and accurately from an well captured image, a license plate recognition scheme with the image super-resolution technology is proposed. License plate pictures have similar pattern features though coded with different numbers and characters. Therefore, license plate images are very suitable for super-resolution reconstruction. The system proposed in this paper is mainly composed of license plate detection and location, super-resolution reconstruction, character segmentation, character recognition and other modules. The three license plate detection strategies based on edge detection, color processing and maximum stability and extreme region algorithm are synthesized with parallel programming skills to get the candidate license plates. The support vector machine classifier is trained by using positive and negative samples of license plate images in advance. After the classifier is obtained, the real license plate is selected with the prediction model. The real license plate is then reconstructed with super-resolution technic. This stage is implemented mainly by the method based on the anchored neighborhood regression. This method combines the advantages of sparse dictionary learning and neighborhood embedding. Thus the accuracy and speed of calculation are both well taken into account. The OpenCV library is employed in the project to do character segmentation for the reconstructed image. An artificial neural network is then employed on the recognition stage. Before recognition, a certain number of positive and negative samples of character images are prepared to train the recognition models. In this paper, we use two single hidden layer neural networks and train with back propagation algorithm. After the network is fine tuned the features of the numbers and characters from test images are sent to the networks to complete the final recognition. In order to test the performance of the system, one hundred pictures of the license plate collected from the actual scenes serve as the test set. Experiments show that the system has high recognition accuracy and fast processing speed.

Keywords: license plate recognition; super-resolution reconstruction; OpenCV library; anchored neighborhood regression; support vector machine; artificial neural network

0 引言

随着信息科学的高速发展, 智能化场景在人们生产生活中变得越来越常见。其中智能化交通设施就是比较经典而常见的一个。汽车牌照是车辆身份的唯一标识, 通过智能化手段对现实场景中的汽车进行身份识别是智能交通提出的基本功能需求。这样的识别系统一般主要包含车牌检测定位、字符分割和字符识别 3 个部分。实际拍摄的图像往往存在噪声污染, 形变扭曲, 字符断裂, 牌照颜色、尺寸多样化等问题, 而且系统对算法的复杂度鲁棒性有较高要求, 所以单一地运用某种方法不能解决问题^[1]。本文研究的车牌识别方法基于一定预处理技术

提取车牌图块, 对其运用超分辨率的方法重建, 然后对重建后的图片执行车牌识别前处理流程, 提取到车牌并分割好字符后做特征提取, 最后送入识别网络进行识别。测试之前先使用来源于网络的 100 张全国各省的牌照图片以及若干数量的字符图片来训练 SVM 判决器和 ANN 识别网络以及学习车牌的稀疏字典对。测试时使用了实地采集的小汽车正面或背面图片样本, 共 100 张。样本拍摄时候的距离大都在三米之内, 拍摄角度大都倾斜较小, 光照均匀聚焦良好。经实验测试表明, 该方案是一套行之有效的车牌识别方案。

1 车牌超分辨率重建与识别系统设计

该系统主要包括五个方面: 候选车牌检测、车牌 SVM 判决、车牌的超分辨率重建、车牌字符分割和字符识别。这前 4 个过程均可以视为识别的前处理, 这些流程均是为了最后的识别。系统输入为彩色图片, 一次能识别一个牌照。系统输出为

收稿日期: 2017-12-18; 修回日期: 2018-01-22。

作者简介: 曾超(1992-), 男, 湖北武汉人, 工学硕士, 主要从事计算机视觉和机器学习方向的研究。

牌照的识别结果, 即一个字符串。系统的结构如图 1 所示。图中的每个模块将在后文中详细描述。

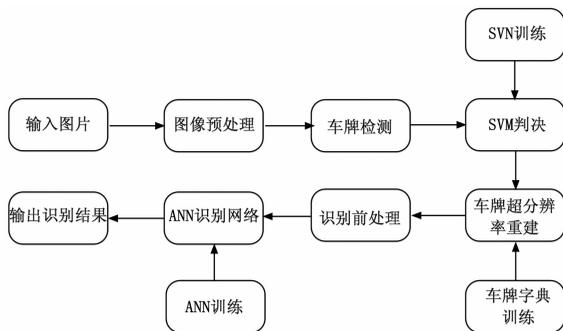


图 1 系统整体流程

1.1 车牌检测

车牌检测是车牌识别系统的关键步骤, 只有正确检测到图片中的车牌区域, 后续的车牌分割提取以及识别才有意义。本文综合基于颜色、基于边缘和基于 MSER 算法三种策略, 运用并行计算技术分别得到三种方法的检测结果, 综合之后来给出最后的车牌检测结果。

1) 基于车牌颜色的车牌检测。国内车辆牌照依据车辆类型分别有蓝底白字、黄底黑字、黑底白字、白底黑字和白底红字等 5 种标准。这种独特的颜色特征可以用来检测车牌的存在。

首先将 RGB 模型转换成 HSV 模式, 即色调、饱和度、亮度模型。当 H 的取值在 200~280 度之间, 颜色就可以确认在蓝色范畴; H 取值 30~80 度之间就确认为黄色; H 取值 0 度左右为红色; 以上颜色在 V 和 S 越接近 1 时候越纯正。S 接近 0 而亮度 V 接近 1 为白色, 而 S 和 V 都接近 0 时为黑色。这样, 将原图在 HSV 空间中遍历所有的像素。根据前面的分析, 若 H 值在 200~280 度之间, 且 S 和 V 的值在 0.30~1 之间的像素点颜色标记为白色, 此时 S 和 V 不在此范围则标记为黑色。于是可以得到原图像的一个经过处理后的二值图像。对这个二值图像做进一步的形态学处理, 再结合车牌先验知识, 最终确认车牌候选区域。

基于车牌颜色特征的检测时间效率很高, 对于高清彩色车牌照片效果较好。不足之处在于, 如果照片背景颜色过于复杂, 比如蓝色车牌照片中有大量蓝紫色的图像块或区域, 导致筛选提取受到干扰。

2) 基于形态学处理以及边缘检测的车牌检测。

车牌图像区域相对背景区域有个明显的特点就是, 车牌部分的矩形边框和里面的数字字母都包含着大量的横竖边缘, 尤其是竖直边。通过特定的一个图像基元块与图像做某种类似逻辑运算的作用, 可以得到这一边缘密集的区域。常见的这种操作有腐蚀膨胀、开操作和闭操作。具体地, 先将彩色图像转化为灰度图, 并且利用 5*5 高斯模糊去除摄像机和环境噪声。然后利用 sobel 滤波器和一个阈值滤波器来得到二值图像。所采用的阈值由 Otsu 算法得到。使用形态学处理方法闭操作来删除竖直边之间的空白区, 从而得到包含这些大量竖直边的连通闭合区域。这些区域就是车牌的候选区域。

3) 基于 MSER 的车牌检测。MSER 全称为 Maximally Stable Extremal Regions, 即最大稳定极值区域。这种区域特点是相对背景而言自成一个组织, 组织内灰度比较一致。通过多次尝试不同的阈值对原图像进行二值化滤波, 获取各个阈值时候的各区域并计算相应形态变化。在一定的阈值变化范围

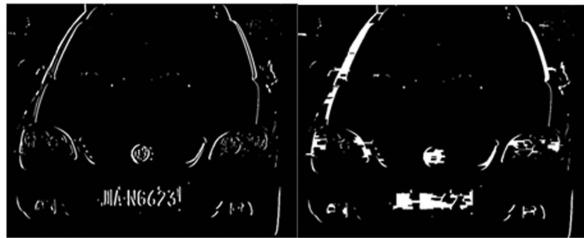


图 2 阈值处理

图 3 形态学处理

内如果某个区域的形状达到最大而且稳定(类似一个数学函数取得极大值的情况), 那么认定其为一个 MSER 区域^[2]。通过设置合适的阈值变化范围可以检测到车牌候选区域。

这样, 同时运行这三种算法就得到了一系列候选车牌区域, 调整其尺寸为 136*36 为下一步的 SVM 判决做准备。

1.2 候选车牌 SVM 判决

上面的方法提取到的矩形区域可能不是车牌, 需要经过判决得到真实车牌。SVM (support vector machine) 是一种基于统计学习理论的机器学习算法, 并且是一种有监督算法。监督学习是通过输入训练数据做运算并对照数据标签提供的相关信息, 利用梯度下降等策略来优化调整系统参数的方法。标签数据标识了每个数据样本对应所属的类别标号。SVM 算法是通过在样本空间通过计算来生成一个最优的判决平面来将所有样本分为两类。本文采用该方法解决一个二分类问题, 即候选矩形图块是或不是真实车牌图片。

对测试图片做判决工作之前, 需要先用一定数量的正负样本来训练 SVM 分类器。本文使用了在现实场景中拍摄以及从网络获取的数百张汽车牌照和非牌照照片作为原始素材, 通过程序提取车牌图片作为正样本, 使用相同尺寸的非车牌图片作为负样本。下面将详细介绍 SVM 具体建模过程。

1) 首先确定分类器个数。这里只有车牌这一个类别, 则分类器个数是 1。

2) 建立样本库。选取来源于互联网的各省车牌图片 100 张作为正样本负样本图像块数据来源。利用系统测试时产生的与车牌等大小的矩形候选图块按照是否为车牌作为正负样本。这些样本在做成训练数据之前需要先预处理, 比如灰度化, 图像滤波等。

3) 特征提取。本文使用大小为 136*36 的整张车牌图片作为训练数据输入, 因此特征就是车牌图片本身。特征值排序后对其归一化处理, 与标签数据一起做成训练数据。

4) 核函数选定。SVM 的构造效果受核函数选取影响很大, 合适的核函数选取可以得到显著的分类效果, 相反核函数选取不当则分类效果会大大降低。主要的核函数有线性内核, 多项式内核, 径向基内核 (RBF) 等, 其中 RBF 核应用最广泛。本文选取 RBF 高斯径向基核函数。

5) 训练 SVM。根据选定的核函数计算每个特征向量的映射值。SVM 分类问题可归结为寻找一个样本集的最优分割面。离这个分割面最近的正负样本点称为支持向量。优化目标则进一步转为去最大化这些支持向量到分割面的距离。通过一些数学转化, 优化问题可以表达为,

$$\min_{w,b} \frac{1}{2} \|\omega\|^2$$

$$s. t. y^{(i)} (\omega^T x^{(i)} + b) \geq 1, i = 1, \dots, m$$

ω 和 b 都是 SVM 的模型参数，也是决定输入样本所属类别的决策平面的特征参数。 y 是训练样本的类别标签。 m 为训练样本的个数。不等式约束条件是正负样本分别约定标示 y 为 1 和 -1 时的表达形式，等号当且仅当输入为支持向量时取得。

通过拉格朗日松弛引入松弛因子 α ，得到拉格朗日多项式，将约束转化到优化目标函数中去，问题转化为求以下式子的最小值：

$$L(\omega, b, \alpha) = \frac{1}{2} \|\omega\|^2 - \sum_{i=1}^m \alpha_i [y^{(i)} (\omega^T x^{(i)} + b) - 1]$$

最后运用 KKT 条件求解可得到模型参数以及预测表达式。此时 SVM 模型参数随之确定。

OpenCV 提供了一个 FileStorage 类可以来管理数据文件。我们借助这个类来存储训练数据矩阵和类别标签。以上步骤完成好了就可以进行训练了，这时候系统会按照以上数学原理来计算生成模型参数。借助 OpenCV 的 CvSVM 类，我们用之前生成的训练数据文件来初始化一个判决器实例来完成训练过程。训练完成后，系统就具备判决类似样本数据类别的能力了，这个称为训练过程。如图 4 所示为训练 SVM 的部分样本数据。



图 4 SVM 训练样本集

经过判决后选出真实的灰度车牌如图 5 所示。



图 5 提取的灰度车牌

1.3 车辆图片的超分辨率重建

这个步骤运用超分辨率的方法对提取到的灰度车牌进行处理。车牌图片质量得到了增强，大小和清晰度变得各加有利于下一步的处理。

目前研究的主流超分辨率重建方法有三种：基于插值、基于重建和基于字典学习的方法。其中基于插值的方法比较简单也最为实用，能满足一般的要求且能应用于实时场景。基于重建的方法是利用前向观测模型对输入的低分辨率图像进行一致性约束，同时结合图像先验信息进行求解，适合和运动估计相关的多帧图像的超分辨率重建场合。为了加入更多的先验信息，出现了基于学习的方法。基于学习的方法最早可追溯到 Freeman 的 VISTA 方法^[3]。这种方法主要思路是分别选定高低分辨率训练图像集，在其上建立一种对应关系。近年来兴起了基于字典学习的方法，建立高低分辨率图像块的基元组合即字典。字典的选取有邻域嵌入方法 (Neighborhood Embed-

ding) 和稀疏编码法 (Sparse Coding)。

邻域法认为同一幅图的低分辨率和高分辨率版本中，流形对象的几何结构是一样的。因此每一个图像块由 K 最近邻的其他图块重建的关系也应该是一样的，即这里的重建系数是一样的。这种方法直接使用采样图块作为字典组成元素。但是随着图像块数目增多，即这里采用的字典的原子数增大，计算复杂度会大大增加。

为了解决这个计算复杂度增加的问题，Yang 等结合信号的稀疏编码理论提出稀疏字典学习方法^[4]。

这种方法主要包括两个阶段，先学习稀疏字典然后重建。首先分别建立高分辨率和低分辨率图块的字典 D_h 和 D_l 。问题归结为寻找低分辨率块 y_l 在字典 D_l 下的最稀疏表示：

$$\min_{\alpha} \|\alpha\|_0, s. t. \|FD_l\alpha - Fy_l\|_2 \leq \epsilon$$

式中， α 表示低分辨率字典下的稀疏表达， F 表示特征提取作用矩阵。 L_0 范数优化属于 NP 难问题，于是转化成 $L1$ 范数最小化：

$$\min_{\alpha} \|\alpha\|_1, s. t. \|FD_l\alpha - Fy_l\|_2 \leq \epsilon$$

引入正则化因子 λ ，进一步转化为：

$$\min_{\alpha} \|\alpha\|_1 + \frac{1}{2} \|FD_l\alpha - Fy_l\|_2^2$$

这是一个 $L1$ 范数线性回归问题，可以运用优化理论求解。解出最优系数 α 后进行超分辨率重建：

$$x_l = D_h\alpha$$

这种基于稀疏编码的方法解决了字典数据集体量急剧增加的问题。但是这个稀疏字典集的建模过程仍然需要耗费大量时间。

在以上研究的基础之上，Radu 等人提出一种结合邻域和稀疏编码的方法——固定邻域回归法 (Anchored Neighborhood Regression)^[5]。本文将使用这种方法来完成车牌图片的超分辨率重建。

运用岭回归的思想，该问题首先可以归结为：

$$\min_{\beta} \|y_F - N_l\beta\|_2^2 + \lambda \|\beta\|_2$$

其中： N_l 代表 LR (low resolution) 空间的邻域字典，与之对应地邻域法中则是输入特征 y_F 的 K 个最近邻，在稀疏编码中则是 LR 字典。 λ 可以消除奇异解问题，使问题能有稳定解。通过线性代数的方法可以解得图块的邻域表达 β 为：

$$\beta = (N_l^T N_l + \lambda I)^{-1} N_l^T y_F$$

在该方法中，LR (Low Resolution) 和 HR (High Resolution) 版本的邻域关系视为等同，于是可以据此并结合 HR 的邻域矩阵可以得出重建的高分辨率图块：

$$x = N_h\beta$$

这里 x 即为 HR 输出的图像块。 N_h 是 HR 版本的图块邻域矩阵，和 N_l 对应。至此求得了由输入 LR 版本的特征向量 y_F 得到 HR 版本图块的公式：

$$x = N_h (N_l^T N_l + \lambda I)^{-1} N_l^T y_F$$

该方法可以从一个学习到的稀疏字典开始。对稀疏字典中的每个原子通过计算与其他原子的相关系数来找到 K 个最相近的邻居。这样原来图块的稀疏表达可以转换成这些邻居原子的表达。以这种方式，对于特定的图块而言，其表达空间维度大大减小，原来需要整个字典集来查找表达系数，现在只需要在一个近邻空间里去寻找表达，这样计算效率将大大提高。

车牌图片可以看做一种特有的模式，其上的纹理是十分相

似的。因此车牌图片适合单独训练一个字典然后完成高分辨率重建。通过学习一定数目的较清晰车辆图片和较模糊车辆样本, 来训练这个字典。

进行超分辨率重建后, 车辆图片分辨率明显提高, 字符骨干得到强化, 从而各加利于下一步的处理。超分辨率重建后的车牌如图 6 所示。



图 6 灰度车牌超分辨率重建

1.4 字符分割

经过超分辨率重建、滤波、二值化等过程, 下一步采用基于连通域的方法来提取字符。字符分割的目的就是为了得到单独的字符图片作为识别模块的输入。常见的车牌分割方法有以下三种。

1) 垂直投影法。得到车牌的二值图后, 从左到右扫描车牌的每一个纵列并统计非零像素个数。不难发现当扫描到字符间隙的时候这个统计值达到极小值, 且相邻极小值之间就是字符区域。这种方法分割速度快, 满足实时性要求, 不足之处在于对字符粘连处理效果不好。

2) 模板匹配法。这种方法需要首先根据标准车牌设计一个同比例的模板, 再调整到实验中的车牌尺寸。在该模板上用矩形框标记好字符区域的位置。当使用该模板从左到右在实验车牌上滑动时, 标记区域内的白色像素值(字符像素)将在某一时刻达到峰值。此时矩形区域内选中的部分即为实验车牌的字符区域。该方法对于车牌字符有缺损有较好的分割效果, 但是计算复杂度较高, 每次滑动的时刻要统计区域内的像素, 相邻时刻还要计算变化趋势。

3) 连通域方法。车牌上除了汉字, 其余字符均是连通的。运用这个特点, 使用 OpenCV 的库函数在二值车牌上标记连通域并用外接矩形来包围这些区域, 以此分割得到车牌上的连通区域。通过尺寸校验字符像素空占比等可以去掉分隔符等干扰。此方法对于车牌图片质量要求较高, 否则容易受到背景要素干扰。

本文结合了连通域方法和基于模板匹配的分割方法。OpenCV 提供了一个查找轮廓的函数, 可以方便的找到二值图中的连通区域的轮廓。对连通域方法检测到的车牌字符以水平位置排序。经过尺寸校验去除干扰后, 取后面六位字符存入字符向量。最后根据汉字到第一个英文字符的距离制作模板提取汉字字符存入单独的汉字字符向量。

字符分割定位效果如图 7 所示。



图 7 字符分割结果

1.5 字符识别

识别之前先选取适当结构的神经网络, 将有标签的训练样本输入并使用 BP 算法训练。得到网络模型后对测试样本提取特征送入神经网络进行识别。

1.5.1 字符特征选取

字符图片送入神经网络进行识别之前首先要做特征提取。

传统的特征有字符图像的水平 and 垂直累计直方图, 低分辨率字符图像样本, SIFT (Scale-invariant feature transform) 特征, LBP (Local Binary Pattern) 特征等。本文同时采用 LBP 特征和水平垂直累积直方图特征。LBP 是一种用来描述图像纹理特征的算子。初级的 LBP 算子反映每个像素与周边像素的大小关系(比中心像素大则置 1, 否则置 0), 对所有像素和其邻域执行这种统计和比较便可以得到这种特征, 特征维数和原图大小相同。

example	thresholded	weights
6 5 2	1 0 0	1 2 4
7 6 1	1 0 0	128 8
9 8 7	1 1 1	64 32 16

$$LBP=1+16+32+64+128=241$$

图 8 LBP 特征示意图

对于数字和字母图片尺寸归一化到 10×10 , LBP 特征占去 100 个数据位, 加上横竖的累计直方图信息 20 位, 整个特征将占用 120 位数据。这也决定了非汉字网络的输入神经元为 120 个。类似地, 汉字字符考虑到其复杂性, 尺寸归一成 20×20 , 对应的特征将占用 440 位, 因此汉字网络的输入神经元个数为 440。特征收集好了之后, 将使用人工神经网络来完成字符分类即识别过程。

1.5.2 人工神经网络选取

人工神经网络的本质是一个从输入映射到输出的多层复合函数。

如果节点之间不构成循环, 就称为前馈网络。前馈网络也是最基本的神经网络。本文就使用这种网络来做字符识别。

一般而言, 层数多则网络模拟现实世界复杂关系的能力就强, 但与此同时网络的训练方法会随着层数增加而变得非常复杂。如图 9 所示, 本文采用一个单隐层的网络结构, 为简化表达, 输出层只画了一个节点作为代表。输入层不计, 总层数为两层, 即隐层和输出层。

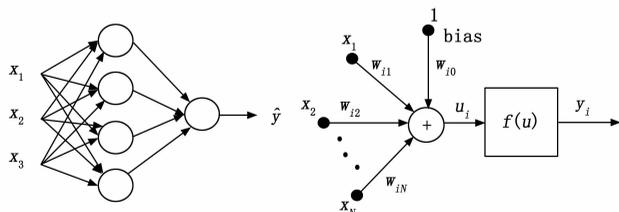


图 9 神经网络表示

图 10 神经元表示

如图 10 所示, 对于每一个神经节点, 来自上层的输出首先作为本层的输入被线性加权且带有一个额外的偏置量。线性加权后的结果随后进入激励函数, 作用后传入下一层。以第一层即隐层为例, 用公式表达为:

$$z^{[1]} = w^{[1]} a^{[0]} + b^{[1]}$$

这里 $a^{[0]}$ 表示上一层的输出, 这里实际上是输入向量。 $w^{[1]}$ 为第一层的权重矩阵, $b^{[1]}$ 为第一层的线性偏置。 $z^{[1]}$ 为第一层的输出向量。对于整个网络而言, 信息的传导方向从网络输入到输出层, 如图 11 所示。输入经过第一层节点线性加权后经过激励函数后被第二层再次线性加权, 随后经过第二层即输出层激励函数作用, 最后在输出层计算损失函数。这称为信息流的前馈过程, 如图 11 所示。

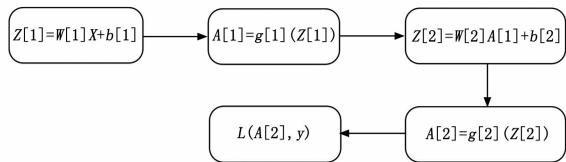


图 11 信息流前馈过程

中括号中的 1 和 2 分别表示当前矩阵参数所在的层数标识。X 表示输入，Z 表示线性加权结果，A 表示经过激励函数后的结果，L 表示损失函数。

1.5.3 BP 算法流程与网络训练

BP 算法 (Back Propagation) 的提出极大促进了神经网络的发展，使得神经网络的训练变得切实可行，是一种被广泛采用的网络学习方法。本文使用这种方法来更新网络参数形成识别模型。这个学习过程实际是一个优化过程，优化目标是一个损失函数：

$$L(a, y) = -(y \log(a) + (1 - y) \log(1 - a))$$

a 是实际输出， y 是期望的输出 (类别标签)。实际计算损失函数时要逐个样本求和。BP 算法则本质上是通过对这个函数求解对网络连接权重的梯度然后迭代更新这个权重系数 w ：

$$w_{ij}(k + 1) = w_{ij}(k) - \alpha \frac{\partial L}{\partial w_{ij}}$$

其中： α 是学习因子，决定向局部最优点的接近速度，选取太小训练速度很慢，选取过大不易收敛，需要通过实验来逐步调整确定。 i 表示层数标识， j 表示层中的节点标识，每个节点的 w 向量包含若干个元素，其个数等于与上层的连接数。这个迭代一直持续到输出优化到预定的水平或者迭代次数达到限制为止。因此为迭代得到各个权值系数，需要计算对应的微分量。为了推导表示的方便，下面约定 dL/dz 这一微分量就用 dz 简化表达，其他微分量也都这样类比来表达。

输出层采用的激励为 Sigmoid 函数。结合以上公式以及求导的相关知识可以导出，

$$dz^{[2]} = a^{[2]} - y$$

z 表示上一层的输出 a 经过本层线性加重的结果。本层的 z 经过激励函数将又得到本层输出的 a 。

由输出层的线性加权关系可以导出，

$$dw^{[2]} = dz^{[2]} a^{[1]T}$$

$$db^{[2]} = dz^{[2]}$$

由链式法则可知，

$$\frac{dL}{dz^{[1]}} = \frac{dL}{da^{[1]}} * \frac{da^{[1]}}{dz^{[1]}}$$

式中， dz 表示真实的微分而不是上面约定的比值。不难发现上式右边第二项即为第一层激励函数 g (双曲正切) 的导数，第一项可以由输出层的 z 与 a 线性关系式转化成 L 对 $z^{[2]}$ 的导数，同时结合矢量微积分可以导出，

$$dz^{[1]} = w^{[2]T} dz^{[2]} * g^{[1]'}(z^{[1]})$$

第一层的 $dw^{[1]}$ 和 $db^{[1]}$ 和第二层表达关系形式上是类似的，都是线性加权关系决定的。

计算了以上微分量之后于是可以代入权值迭代公式：

$$w^{[i]}(k + 1) = w^{[i]}(k) - \alpha dw^{[i]}, i = 1, 2$$

$$b^{[i]}(k + 1) = b^{[i]}(k) - \alpha db^{[i]}, i = 1, 2$$

实际编码时，为了启动迭代还需要对权值进行随机初始化得到初始权值向量。每得到一次新的权值向量，信息流进行一

次前馈过程直到输出层，在输出层计算偏差；信息流到达输出层后又进行一次反向传导计算各权值对应的微分量直到第一层，微分量更新后又计算新的权值向量。整个迭代过程周而复始，信息流的前馈和反向传播交替进行，直到输出层得到的偏差收敛于预设的水平或者迭代达到最大限度。这便是用反向传播算法训练神经网络的基本过程。对于每个训练样本都经历上述迭代过程，最后平均各样本得到的结果即得到整个训练完成后的分类模型。

考虑到数字字母与汉字形态的差异性，使用两个独立的网络分开进行训练。该训练过程为有监督训练，因此需要提供样本数据的类别标签信息。为此引入 $N * M$ 的训练矩阵， N 表示训练样本的数目， M 是样本的类别数。比如对于单纯的数字网络， M 等于 10，即十个待识别的数字类别数。加上车牌 24 个英文字母 (不考虑字母 I 和 O) 的个数，对于非汉字网络类别数为 34。对于汉字网络类别数为 31。如果第 i 行样本属于第 j 类那么该矩阵 (i, j) 位置标记为 1。以这种方式来存储各个字符样本的类别标签。系统识别完成后会给出字符属于每个类别的概率从而选取最大的那个完成分类。

训练过程即神经网络的构建过程。通过前处理，将提取的字符特征输入神经网络先进行训练，再利用测试集测试识别性能。

1.5.4 神经网络配置

对于非汉字网络输入神经元 120 个，汉字网络输入神经元 440 个。隐层神经元 40 个。非汉字网络输出节点 34 个，汉字网络输出节点 31 个。学习率取 0.01。激励函数隐层选取常用的双曲正切 \tanh 函数，输出层激励则选择 Sigmoid 函数。训练算法为 Back Propagation 算法。

2 实验效果与评估

系统由 VS2013 开发平台采用 C++ 编码，并链接了 OpenCV 库，在 CPU 主频 2.3 GHz，8 G 内存的 Win10 家庭版笔记本电脑平台上开发实现并测试识别效果。采用来源于网络的训练样本 100 张，拍摄距离 2 米左右，车牌倾斜较小。SVM 使用这些样本在实验时产生的分辨率为 $136 * 36$ 的车牌图片作为正样本，使用程序在源图中提取相同大小的非车牌区域块作为负样本，各 150 张。ANN 训练字符样本通过对车牌样本图片执行字符分割算法得到。汉字分辨率均为 $20 * 20$ ，数字和字母为 $10 * 10$ 。实验结果如表 1 所示。

定位准确率为所有检测到真实车牌的样本占测试集的比例。识别准确率表示得出正确识别结果的样本占比。拒识率是由于车牌未检测到或者车牌曝光条件等不符合要求等而导致系统未给出识别结果的样本比例。测量时统计每次处理的时间，最后取平均数作为系统运行过程花费的时间。

表 1 实验结果

定位准确率	识别准确率	拒识率	处理时间
95%	93%	5%	134 ms

对误识和拒识的样本通过逐个反复测试分析发现：1) 个别车牌倾斜程度较大，汉字分割不准从而导致识别偏差；2) 对于少数远距离拍摄的样本发生拒识现象，主要原因是此时背景的干扰要素与近距离样本相比明显变多，系统无法检测到车牌。对于近距离且无明显倾斜形变的绝大部分车牌样本，系统表现良好。对于这些样本，由于识别前处理加入超分辨率模块改善

了字符的可辨识度, 从而保证了较高识别准确率。对于较远距离的复杂场景则需要研究更加灵敏并且鲁棒性好的车牌检测方法。

3 总结与展望

本文讨论了如何结合超分辨率领域的最新成果, 对车牌进行预处理进而较准确地完成识别。从至少三个方面增强了一个基本的车牌识别系统: 一是引入了超分辨率重建技术来强化车牌识别的前处理流程; 二是增加了训练良好的 SVM 判决器来确认识别前得到的是真实车牌; 三是引入并行计算来综合不同的车牌检测策略, 增加检测准确率的同时提高效率。未来考虑统一检测和识别过程, 建立端到端的识别框架, 使得二者共用特征空间并建立统一提取汉字和其他字符的机制。

(上接第 243 页)

2) 剔除跳点, 保证数据的正确性。采用多种方法实现: a) 判断数据的合法性; b) 通过相关的计算判断数据的合理性, 如相邻两点之间距离过大, 超过理论上飞机所能达到的最大速度, 则可认为该数据点为跳点; c) 采用常见的均方根值法判断跳点。

3) 轨迹的动态变化, 保证界面的整洁性。多架飞机同时飞行, 如果显示每架飞机的完全轨迹则会显得地图太乱, 系统通过创建动态数组, 将飞行轨迹保持在一定范围内, 从而保证界面的整洁性。

3 应用实例

目前系统已经在各个遥测站以及监控大厅中使用, 对保障重点型号的试飞起到了重要的作用。系统主要软件包括 PCM 数据发送软件、二次雷达数据解析发送软件和多目标电子地图软件。

3.1 服务器端软件

图 7 为 PCM 飞参数据发送软件参数配置界面, 通过配置飞机关键参数名来提取参数物理量, 并通过网络组播方式发送至多目标电子地图。该软件即可独立运行, 也可嵌入到其他大型软件中, 作为一个功能使用。本系统就采用在前端实时处理软件嵌入 PCM 飞参数据转发功能模块实现了 PCM 数据到电子地图的传输。



图 7 PCM 数据发送软件

3.2 客户端软件

图 8 为多目标电子地图软件, 上部分电子地图显示飞机的位置信息, 下部分列表显示飞机部分关键参数信息。在显示信息最右边选择遥测站后, 会显示飞机相对该遥测站的水平和垂直角度。图中以机场为中心显示坐标轴, 以每隔 100 公里绘制飞行半径。如图 8 所示, 地图同时显示四架飞机, 其中飞机 1

参考文献:

[1] 吴 昊, 丁庆生. 一种完整的汽车牌照识别算法 [J]. 计算机测量与控制, 2008, 16 (3): 392-394.

[2] Matas J, Chum O, Urban M, et al. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions [A]. BMVC [C]. London, 2002: 384-393.

[3] William T. Freeman, Egon C. Pasztor. Learning low-level vision [A]. ICCV [C]. 1999, 40 (1): 1182.

[4] Jianchao Yang, Image Super-Resolution via Sparse Representation [J]. IEEE Transactions on Image Processing, 2010, 19 (11): 2861-2873.

[5] Timofte R, De V, Gool L V. Anchored Neighborhood Regression for Fast Example-Based Super-Resolution [J]. IEEE international Conference on Computer Vision, 2013: 1920-1927.

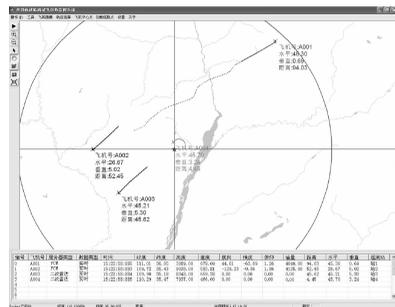


图 8 多目标电子地图

显示了完整轨迹, 并用虚线画出, 其余三架仅显示最近两分钟内的轨迹。用户可通过菜单栏的“设置”功能编辑地图基本信息以及选择数据源的类型。当 PCM 数据中断时, 用户可通过可在此项设置中切换数据源为二次雷达类型, 从而快速搜索到当前跟踪目标。使用表明, 使用该软件后, 将丢失目标的搜索时间由原来的 3~5 分钟缩短至 20~30 秒, 大大提高了跟踪效率。

4 结论

系统采用面向对象程序设计语言 VC++ , 集成 MapX 控件, 并结合单位型号试飞特点, 设计并实现的多目标试验机试飞态势监视系统的开发。经实际使用表明, 软件简单易用, 完全满足多目标监视的需求, 由于引入了二次雷达数据以及计算了遥测天线角度, 解决了当前飞行试验中测试跟踪目标丢失后的重新搜索和跟踪问题, 缩短了目标锁定时间, 提升了遥测信号的质量, 为试飞安全提供了保障。

参考文献:

[1] 于 艳, 等, 分布式网络遥测数据处理系统软件设计 [J]. 测控技术, 2006 (6): 62-64.

[2] 刘升护, 王 鹏, 叶 冰. 多目标电子地图的设计与实现 [J]. 现代电子技术, 2015, 38 (12): 94-97.

[3] 秦 浩, 徐景硕, 高 扬, 等. 基于 VC++ 与 MapX 的导航电子地图的实现 [J]. 航空计算技术, 2011, 41 (5): 98-101.

[4] MapInfo Corporation. MapX 5.0 开发人员手册 [Z]. 2004.

[5] 颜辉武. VC++ 环境下基于 MapX 控件的 GIS 应用软件的开发 [J]. 昆明理工大学学报: 理工版, 2001 (6): 12-17.

[6] 王 庆, 周喜宁, 宋 磊. 基于 MapX 的模拟器电子海图软件设计研究 [J]. 现代电子技术, 2014, 37 (1): 119-122.

[7] 王延路, 刘语乔. 试飞监控电子地图设计与实现 [J]. 现代电子技术, 2013, 35 (13): 131-133.