



(12)发明专利申请

(10)申请公布号 CN 109902751 A

(43)申请公布日 2019.06.18

(21)申请号 201910160157.3

(22)申请日 2019.03.04

(71)申请人 福州大学

地址 350108 福建省福州市闽侯县上街镇
福州大学城学院路2号福州大学新区

(72)发明人 吴林煌 杨绣郡 陈志峰

(74)专利代理机构 福州元创专利商标代理有限公司 35100

代理人 蔡学俊

(51) Int. Cl.

G06K 9/62(2006.01)

G06K 9/38(2006.01)

G06N 3/04(2006.01)

G06N 3/08(2006.01)

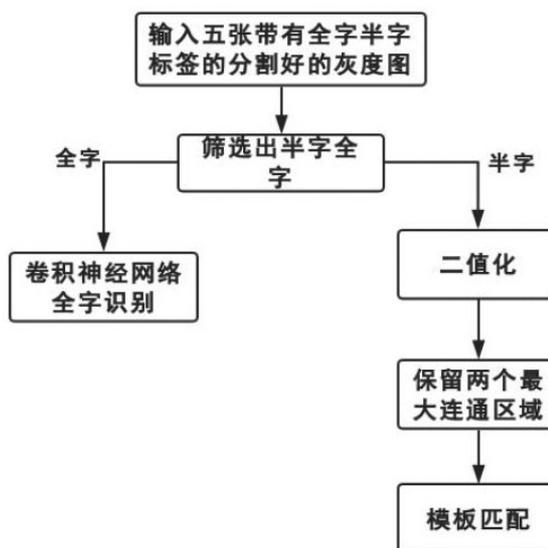
权利要求书2页 说明书10页 附图6页

(54)发明名称

一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法

(57)摘要

本发明涉及一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,首先获取带有全字半字标签的分割好的灰度图,接着将标签筛选分类,若标签为0则进行半字识别,若标签为1则进行全字识别;最后按照图像输入的顺序,依次输出灰度图的识别结果。本发明识别正确率高,为表盘数字字符半字全字的分开识别提供了新思路,并融合卷积神经网络和半字模板匹配,可广泛应用于需要进行数字字符识别的领域。



1. 一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,其特征在于:包括以下步骤:

步骤S1:获取带有全字半字标签的分割好的灰度图,并输入到步骤S2中;其中,标签0代表半字,标签1代表全字;

步骤S2:将标签筛选分类,若标签为0则进入步骤S4,若标签为1则进入步骤S3;

步骤S3:训练一个卷积神经网络,将标签为1的全字灰度图输入到训练好的神经网络中识别,得到全字识别结果;

步骤S4:将标签为0的半字灰度图二值化后,保留两个最大连通域,得到处理后的待识别半字二值图像 M_2 ;

步骤S5:将步骤S4得到的处理后的图像与半字模板进行基于像素的模板匹配,得到半字识别结果;

步骤S6:按照步骤S1中图像输入的顺序,依次输出灰度图的识别结果。

2. 根据权利要求1所述的一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,其特征在于:步骤S3具体包括以下步骤:

步骤S31:输入测试集图片和标签以及训练集图片和标签,其中全字共有0至9共10个字符,用10个独热码表示;

步骤S32:用TensorFlow训练一个前向传输的卷积神经网络;

步骤S33:保存该卷积神经网络的各项权重值;

步骤S34:将输入的全字灰度图输入到训练好的卷积神经网络中进行自动识别,得到识别结果。

3. 根据权利要求1所述的一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,其特征在于:步骤S4具体包括以下步骤:

步骤S41:将半字灰度图 M 进行二值化处理,得到初步待识别半字二值化图像 M_1 ;

步骤S42:将 M_1 用八领域的方式找出连通域,并以自然数标记连通域;

步骤S43:计算步骤S42标记的相同自然数的个数,比较得到两个最大个数的自然数,将这两个自然数所在的连通域的像素灰度值改为1,其余位置的像素灰度值改为0,得到保留最大连通域后的待识别半字二值化图像 M_2 。

4. 根据权利要求3所述的一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,其特征在于:步骤S41具体包括以下步骤:

步骤S411:设定一个 $t \times t$ 大小的候选框作为局部二值化每次选取像素的滑动窗口,设定两个初始阈值: T 、 S ;其中 T 为当窗口只存在背景或目标时判断是否是目标的阈值; S 为判断该滑动窗口中是否背景和目标同时存在的阈值;记 $f(x, y)$ 为图像 M 在第 x 行 y 列的像素灰度值;记 $f'(x, y)$ 为经过局部二值化处理后的图像在第 x 行 y 列的像素的值;

步骤S412:用 $t \times t$ 的候选框以步长1遍历所述灰度图像 M ;令 $T_1(x, y) = \frac{1}{2} \times (\max + \min)$, $T_2(x, y) = \max - \min$,其中, \max 为灰度图像在该候选框范围下的像素点灰度值最大值, \min 为灰度图像在该候选框范围下的像素点灰度值最小值, $T_1(x, y)$ 表示该候选框内灰度值最大值与灰度值最小值的均值, $T_2(x, y)$ 表示该候选框内灰度值最大值与灰度值最小值的差值;

步骤S413:比较 $T_2(x, y)$ 与 S 的大小;若 $T_2(x, y) > S$,则说明候选框中灰色像素值波动范围

大,目标和背景同时存在,则将灰度图像上该像素点的灰度值 $f(x,y)$ 与 $T_1(x,y)$ 对比,若 $f(x,y) > T_1(x,y)$,则令 $f'(x,y) = 1$;若 $f(x,y) < T_1(x,y)$,则令 $f'(x,y) = 0$;若 $T_2(x,y) < S$,则说明候选框中灰色像素值波动范围小,只存在背景或只存在目标,则将所述灰度图像上该像素点的灰度值 $f(x,y)$ 与阈值 T 值相比;若 $f(x,y) > T$,则 $f'(x,y) = 1$;若 $f(x,y) < T$,则 $f'(x,y) = 0$,得到初步待识别半字二值化图像 M_1 。

5. 根据权利要求1所述的一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,其特征在于:步骤S5具体包括以下步骤:

步骤S51:制作半字模板,个数为 z ;

步骤S52:将待识别半字二值图像 M_2 中各个像素点的值按照其各自所在的位置构成一个大小为 $m \times n$ 的矩阵 A ,其中 m 为图像的行数, n 为图像的列数;

步骤S53:将半字模板二值图中各个像素点的值按照其各自所在的位置构成一个 $m \times n$ 的矩阵 C ;

步骤S54:将所述矩阵 A 和所述矩阵 C 两个矩阵逐点进行相减取绝对值操作,然后将所有的值相加,记为 h ;

步骤S55:重复执行步骤S53-S54 z 次,找到最小的 h 值 h_{\min} ,即半字模板与待识别半字的最小汉明距离,该距离下的半字模板的标签即为待识别半字的识别结果。

6. 根据权利要求5所述的一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,其特征在于:步骤S51具体包括以下步骤:

步骤S511:根据表盘真实字符图像构造一张以0、1、2、3、4、5、6、7、8、9、0为顺序上下排列的图像;

步骤S512:以一个大小为 $m \times n$ 的窗口对步骤S511得到的图像进行步长为1的上下移动裁剪并保存,同时将其中全字的模板剔除,得到 z 张半字模板。

一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法

技术领域

[0001] 本发明涉及图像处理与识别中的表盘数字字符识别领域,特别是一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法。

背景技术

[0002] 由于现阶段大多数表盘是基于齿轮的滚动型表盘,比如水表、电表、时钟等等,由于齿轮带动常会出现进位不完全,即低位表盘框同时出现相邻两个数字的上下各一半的情况,而高位为整个数字都在表盘框内,这就将表盘里的数字字符分为两个类别,我们称之为全字和半字。正确识别是表盘数字字符识别的核心,由于出现两种数字字符类别,并且全字半字的属性各不相同,若用同一种方法识别则整体识别效果不是特别好。

发明内容

[0003] 有鉴于此,本发明的目的是提出一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,识别正确率高,为表盘数字字符半字全字的分开识别提供了新思路,并融合卷积神经网络和半字模板匹配,可广泛应用于需要进行数字字符识别的领域。

[0004] 本发明采用以下方案实现:一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,具体包括以下步骤:

[0005] 步骤S1:获取带有全字半字标签的分割好的灰度图,并输入到步骤S2中;其中,标签0代表半字,标签1代表全字;其中输入图像的大小均为 $m \times n$,其中, m 是所述输入图像的行数, n 是所示输入图像的列数;

[0006] 步骤S2:将标签筛选分类,若标签为0则进入步骤S4,若标签为1则进入步骤S3;

[0007] 步骤S3:训练一个卷积神经网络,将标签为1的全字灰度图输入到训练好的神经网络中识别,得到全字识别结果;

[0008] 步骤S4:将标签为0的半字灰度图二值化后,保留两个最大连通域,得到处理后的待识别半字二值图像 M_2 ;

[0009] 步骤S5:将步骤S4得到的处理后的图像与半字模板进行基于像素的模板匹配,得到半字识别结果;

[0010] 步骤S6:按照步骤S1中图像输入的顺序,依次输出灰度图的识别结果。

[0011] 进一步地,步骤S3具体包括以下步骤:

[0012] 步骤S31:输入测试集图片和标签以及训练集图片和标签,其中全字共有0至9共10个字符,用10个独热码表示;

[0013] 步骤S32:用TensorFlow训练一个前向传输的卷积神经网络;其中,该卷积神经网络的结构为“卷积层1-池化层1-卷积层2-池化层2-……-卷积层 x -池化层 x -全连接层”;

[0014] 步骤S33:保存该卷积神经网络的各项权重值;

[0015] 步骤S34:将输入的全字灰度图输入到训练好的卷积神经网络中进行自动识别,得

到识别结果。

[0016] 较佳的,所述卷积神经网络的架构如下:

[0017] 输入层:由于输入的是一个 $m \times n$ 的灰度图像,其在二维空间上的维度是 $m \times n$ 。一个像素点相当于一个神经元,故输入层的维度就是三维 $[32, 24, 1]$;

[0018] 卷积层1:若卷积层的尺寸是 $(2m_1+1) \times (2m_1+1)$,卷积深度为 n_1 ,设定步长为 s_1 。(卷积前先将原始输入图像的上下左右 m_1 以像素0填充)相当于用 n_1 个 $(2m_1+1) \times (2m_1+1)$ 的滑动窗口以步长 s_1 与输入图像在该窗口下的像素相卷积,得到的图像大小为 $[m, n, 1]$;

[0019] 池化层1:设定池化步长为 s_1 、池化尺寸为 $m_2 \times m_2$ 、池化方式:最大池化或平均池化。若采用最大池化,则以步长 s_1 在尺寸为 $m_2 \times m_2$ 的滑动窗口扫描卷积层1得到的图像像素,存储该窗口下的最大值;若为平均池化,则以步长 s_1 在尺寸为 $m_2 \times m_2$ 的滑动窗口扫描卷积层1得到的图像像素,存储该窗口下的像素平均值;故该层池化后得到的图像大小为 $[\frac{m}{m_2}, \frac{n}{m_2}, n_1]$;

[0020] 其余卷积层与池化层与上述所述原理一致,设最终得到图像大小为 $[x_1, y_1, z_1]$;

[0021] 全连接层1:先将得到的图像 $[x_1, y_1, z_1]$ 平坦化成 $[1, x_1 \times y_1 \times z_1]$,即 $x_1 \times y_1 \times z_1$ 个神经元与全连接层1的 $m \times n$ 个神经元全连接;

[0022] 全连接层2(即输出层):全字共有10个标签,用独热码表示,全连接层1的 $m \times n$ 个神经元与全连接层2的10个神经元相连接,输出应为 1×10 大小的向量;

[0023] 步骤S33:保存该卷积神经网络的各项权重值;

[0024] 步骤S34:将输入的全字灰度图输入到训练好的卷积神经网络中进行自动识别,得到识别结果。

[0025] 进一步地,步骤S4具体包括以下步骤:

[0026] 步骤S41:将半字灰度图M进行二值化处理,得到初步待识别半字二值化图像 M_1 ;

[0027] 步骤S42:将 M_1 用八领域的方式找出连通域,并以自然数(1、2、3、4、.....)标记连通域;

[0028] 步骤S43:计算步骤S42标记的相同自然数的个数,比较得到两个最大个数的自然数,将这两个自然数所在的连通域的像素灰度值改为1,其余位置的像素灰度值改为0,得到保留最大连通域后的待识别半字二值化图像 M_2 (默认在二值化图像中,值为0表示白色像素,值为1表示黑色像素)。

[0029] 进一步地,步骤S41具体包括以下步骤:

[0030] 步骤S411:(在灰度图中,将对识别有用的信息称为目标,如数字区域;将对识别无用的信息称为背景,如数字像素点以外的白色区域)设定一个 $t \times t$ 大小的候选框作为局部二值化每次选取像素的滑动窗口,设定两个初始阈值:T、S;其中T为当窗口只存在背景或目标时判断是否是目标的阈值;S为判断该滑动窗口中是否背景和目标同时存在的阈值;记 $f(x, y)$ 为图像M在第x行y列的像素灰度值(以图像M左上角为原点);记 $f'(x, y)$ 为经过局部二值化处理后的图像在第x行y列的像素的值;

[0031] 步骤S412:用 $t \times t$ 的候选框以步长1遍历所述灰度图像M;令 $T_1(x, y) = \frac{1}{2} \times (\max + \min)$, $T_2(x, y) = \max - \min$,其中, \max 为灰度图像在该候选框范围下的像

素点灰度值最大值, \min 为灰度图像在该候选框范围下的像素点灰度值最小值, $T_1(x, y)$ 表示该候选框内灰度值最大值与灰度值最小值的均值, $T_2(x, y)$ 表示该候选框内灰度值最大值与灰度值最小值的差值;

[0032] 步骤S413:比较 $T_2(x, y)$ 与 S 的大小;若 $T_2(x, y) > S$,则说明候选框中灰色像素值波动范围大,目标和背景同时存在,则将灰度图像上该像素点的灰度值 $f(x, y)$ 与 $T_1(x, y)$ 对比,若 $f(x, y) > T_1(x, y)$,则令 $f'(x, y) = 1$;若 $f(x, y) < T_1(x, y)$,则令 $f'(x, y) = 0$;若 $T_2(x, y) < S$,则说明候选框中灰色像素值波动范围小,只存在背景或只存在目标,则将所述灰度图像上该像素点的灰度值 $f(x, y)$ 与阈值 T 值相比;若 $f(x, y) > T$,则 $f'(x, y) = 1$;若 $f(x, y) < T$,则 $f'(x, y) = 0$,得到初步待识别半字二值化图像 M_1 。

[0033] 较佳的,步骤S411中,两个初始阈值的设置为: $T = \frac{256}{2} = 128$, $S = 15$; T 为0-255灰度值的中点。

[0034] 进一步地,步骤S5具体包括以下步骤:

[0035] 步骤S51:制作半字模板,个数为 z ;

[0036] 步骤S52:将待识别半字二值图像 M_2 中各个像素点的值按照其各自所在的位置构成一个大小为 $m \times n$ 的矩阵 A ,其中 m 为图像的行数, n 为图像的列数;

[0037] 步骤S53:将半字模板二值图中各个像素点的值按照其各自所在的位置构成一个 $m \times n$ 的矩阵 C ;

[0038] 步骤S54:将所述矩阵 A 和所述矩阵 C 两个矩阵逐点进行相减取绝对值操作,然后将所有的值相加,记为 h ;

[0039] 步骤S55:重复执行步骤S53-S54 z 次,找到最小的 h 值 h_{\min} ,即半字模板与待识别半字的最小汉明距离,该距离下的半字模板的标签即为待识别半字的识别结果。

[0040] 进一步地,步骤S51具体包括以下步骤:

[0041] 步骤S511:根据表盘真实字符图像构造一张以0、1、2、3、4、5、6、7、8、9、0为顺序上下排列的图像;

[0042] 步骤S512:以一个大小为 $m \times n$ 的窗口对步骤S511得到的图像进行步长为1的上下移动裁剪并保存,同时将其中全字的模板剔除,得到 z 张半字模板。

[0043] 与现有技术相比,本发明有以下有益效果:本发明将半字与全字分开识别,将全字放入训练好的卷积神经网络中识别,将半字放入半字模板中进行模板匹配,二者相结合,最后按顺序输出整个表盘的识别结果,从而提高表盘数字字符的识别正确率,为表盘数字字符半字全字的分开识别提供了新思路,并融合卷积神经网络和半字模板匹配,可广泛应用于需要进行数字字符识别的领域。

附图说明

[0044] 图1为本发明实施例的原理框图。

[0045] 图2为本发明实施例的输入的五张带有全字半字标签的灰度图。

[0046] 图3为本发明实施例的步骤S3卷积神经网络全字识别的结构框图。

[0047] 图4为本发明实施例的步骤S31训练集/测试集标签的独热码表示效果图。

[0048] 图5为本发明实施例的步骤S31部分训练集图片效果图。

- [0049] 图6为本发明实施例的步骤S32采用的卷积神经网络的结构框图。
- [0050] 图7为本发明实施例中步骤S32训练卷积神经网络的过程图。
- [0051] 图8为本发明实施例中图2中的(5)经过二值化后的效果图。
- [0052] 图9为本发明实施例中图8标记连通域的效果图。
- [0053] 图10为本发明实施例中图8保留两个最大连通区域后的效果图。
- [0054] 图11为本发明实施例中步骤S41局部二值化的结构框图。
- [0055] 图12为本发明实施例中步骤S55半字模板匹配中与模板最相似的模板二值图。
- [0056] 图13为本发明实施例中步骤S12部分半字模板效果图。
- [0057] 图14为本发明实施例中最后的输出结果效果图。

具体实施方式

- [0058] 下面结合附图及实施例对本发明做进一步说明。
- [0059] 应该指出,以下详细说明都是示例性的,旨在对本申请提供进一步的说明。除非另有指明,本文使用的所有技术和科学术语具有与本申请所属技术领域的普通技术人员通常理解相同含义。
- [0060] 需要注意的是,这里所使用的术语仅是为了描述具体实施方式,而非意图限制根据本申请的示例性实施方式。如在这里所使用的,除非上下文另外明确指出,否则单数形式也意图包括复数形式,此外,还应当理解的是,当在本说明书中使用术语“包含”和/或“包括”时,其指明存在特征、步骤、操作、器件、组件和/或它们的组合。
- [0061] 本实施例定义全字为整个数字字符图像都完整地在表盘内的数字字符,半字定义为表盘范围内出现一个数字的下半部分和另一位相邻数字的半部分的数字字符,由于在一副完整的表盘内出现两种不同类型的数字,所以将其分开识别,分别处理,可以大大提高数字字符的识别正确率。
- [0062] 如图1所示,本实施例提供了一种融合卷积神经网络和半字模板匹配的表盘数字字符识别方法,具体包括以下步骤:
- [0063] 步骤S1:获取带有全字半字标签的分割好的灰度图,并输入到步骤S2中;其中,标签0代表半字,标签1代表全字;其中输入图像的大小均为 $m \times n$,其中, m 是所述输入图像的行数, n 是所示输入图像的列数;
- [0064] 步骤S2:将标签筛选分类,若标签为0则进入步骤S4,若标签为1则进入步骤S3;
- [0065] 步骤S3:训练一个卷积神经网络,将标签为1的全字灰度图输入到训练好的神经网络中识别,得到全字识别结果;
- [0066] 步骤S4:将标签为0的半字灰度图二值化后,保留两个最大连通域,得到处理后的待识别半字二值图像 M_2 ;
- [0067] 步骤S5:将步骤S4得到的处理后的图像与半字模板进行基于像素的模板匹配,得到半字识别结果;
- [0068] 步骤S6:按照步骤S1中图像输入的顺序,依次输出灰度图的识别结果。
- [0069] 在本实施例中,步骤S3具体包括以下步骤:
- [0070] 步骤S31:输入测试集图片和标签以及训练集图片和标签,其中全字共有0至9共10个字符,用10个独热码表示;

[0071] 步骤S32:用TensorFlow训练一个前向传输的卷积神经网络;其中,该卷积神经网络的结构为“卷积层1-池化层1-卷积层2-池化层2-……-卷积层x-池化层x-全连接层”;

[0072] 步骤S33:保存该卷积神经网络的各项权重值;

[0073] 步骤S34:将输入的全字灰度图输入到训练好的卷积神经网络中进行自动识别,得到识别结果。

[0074] 较佳的,在本实施例中,所述卷积神经网络的架构如下:

[0075] 输入层:由于输入的是一个 $m \times n$ 的灰度图像,其在二维空间上的维度是 $m \times n$ 。一个像素点相当于一个神经元,故输入层的维度就是三维 $[32, 24, 1]$;

[0076] 卷积层1:若卷积层的尺寸是 $(2m_1+1) \times (2m_1+1)$,卷积深度为 n_1 ,设定步长为 s_1 。(卷积前先将原始输入图像的上下左右 m_1 以像素0填充)相当于用 n_1 个 $(2m_1+1) \times (2m_1+1)$ 的滑动窗口以步长 s_1 与输入图像在该窗口下的像素相卷积,得到的图像大小为 $[m, n, 1]$;

[0077] 池化层1:设定池化步长为 s_1 、池化尺寸为 $m_2 \times m_2$ 、池化方式:最大池化或平均池化。若采用最大池化,则以步长 s_1 在尺寸为 $m_2 \times m_2$ 的滑动窗口扫描卷积层1得到的图像像素,存储该窗口下的最大值;若为平均池化,则以步长 s_1 在尺寸为 $m_2 \times m_2$ 的滑动窗口扫描卷积层1得到的图像像素,存储该窗口下的像素平均值;故该层池化后得到的图像大小为

$$\left[\frac{m}{m_2}, \frac{n}{m_2}, n_1\right];$$

[0078] 其余卷积层与池化层与上述所述原理一致,设最终得到图像大小为 $[x_1, y_1, z_1]$;

[0079] 全连接层1:先将得到的图像 $[x_1, y_1, z_1]$ 平坦化成 $[1, x_1 \times y_1 \times z_1]$,即 $x_1 \times y_1 \times z_1$ 个神经元与全连接层1的 $m \times n$ 个神经元全连接;

[0080] 全连接层2(即输出层):全字共有10个标签,用独热码表示,全连接层1的 $m \times n$ 个神经元与全连接层2的10个神经元相连接,输出应为 1×10 大小的向量;

[0081] 步骤S33:保存该卷积神经网络的各项权重值;

[0082] 步骤S34:将输入的全字灰度图输入到训练好的卷积神经网络中进行自动识别,得到识别结果。

[0083] 在本实施例中,步骤S4具体包括以下步骤:

[0084] 步骤S41:将半字灰度图M进行二值化处理,得到初步待识别半字二值化图像 M_1 ;

[0085] 步骤S42:将 M_1 用八领域的方式找出连通域,并以自然数(1、2、3、4……)标记连通域;

[0086] 步骤S43:计算步骤S42标记的相同自然数的个数,比较得到两个最大个数的自然数,将这两个自然数所在的连通域的像素灰度值改为1,其余位置的像素灰度值改为0,得到保留最大连通域后的待识别半字二值化图像 M_2 (默认在二值化图像中,值为0表示白色像素,值为1表示黑色像素)。

[0087] 在本实施例中,步骤S41具体包括以下步骤:

[0088] 步骤S411:(在灰度图中,将对识别有用的信息称为目标,如数字区域;将对识别无用的信息称为背景,如数字像素点以外的白色区域)设定一个 $t \times t$ 大小的候选框作为局部二值化每次选取像素的滑动窗口,设定两个初始阈值:T、S;其中T为当窗口只存在背景或目标时判断是否是目标的阈值;S为判断该滑动窗口中是否背景和目同时存在的阈值;记 $f(x, y)$ 为图像M在第x行y列的像素灰度值(以图像M左上角为原点);记 $f'(x, y)$ 为经过局部二

值化处理后的图像在第x行y列的像素的值；

[0089] 步骤S412：用 $t \times t$ 的候选框以步长1遍历所述灰度图像M；令

$T_1(x, y) = \frac{1}{2} \times (\max + \min)$, $T_2(x, y) = \max - \min$, 其中, \max 为灰度图像在该候选框范围下的像素点灰度值最大值, \min 为灰度图像在该候选框范围下的像素点灰度值最小值, $T_1(x, y)$ 表示该候选框内灰度值最大值与灰度值最小值的均值, $T_2(x, y)$ 表示该候选框内灰度值最大值与灰度值最小值的差值；

[0090] 步骤S413：比较 $T_2(x, y)$ 与S的大小；若 $T_2(x, y) > S$, 则说明候选框中灰色像素值波动范围大, 目标和背景同时存在, 则将灰度图像上该像素点的灰度值 $f(x, y)$ 与 $T_1(x, y)$ 对比, 若 $f(x, y) > T_1(x, y)$, 则令 $f'(x, y) = 1$; 若 $f(x, y) < T_1(x, y)$, 则令 $f'(x, y) = 0$; 若 $T_2(x, y) < S$, 则说明候选框中灰色像素值波动范围小, 只存在背景或只存在目标, 则将所述灰度图像上该像素点的灰度值 $f(x, y)$ 与阈值T值相比；若 $f(x, y) > T$, 则 $f'(x, y) = 1$; 若 $f(x, y) < T$, 则 $f'(x, y) = 0$, 得到初步待识别半字二值化图像 M_1 。

[0091] 较佳的, 在本实施例中, 步骤S411中, 两个初始阈值的设置为: $T = \frac{256}{2} = 128$, $S = 15$; T为0-255灰度值的中点。

[0092] 在本实施例中, 步骤S5具体包括以下步骤:

[0093] 步骤S51: 制作半字模板, 个数为z;

[0094] 步骤S52: 将待识别半字二值图像 M_2 中各个像素点的值按照其各自所在的位置构成一个大小为 $m \times n$ 的矩阵A, 其中m为图像的行数, n为图像的列数;

[0095] 步骤S53: 将半字模板二值图中各个像素点的值按照其各自所在的位置构成一个 $m \times n$ 的矩阵C;

[0096] 步骤S54: 将所述矩阵A和所述矩阵C两个矩阵逐点进行相减取绝对值操作, 然后将所有的值相加, 记为h;

[0097] 步骤S55: 重复执行步骤S53-S54z次, 找到最小的h值 h_{\min} , 即半字模板与待识别半字的最小汉明距离, 该距离下的半字模板的标签即为待识别半字的识别结果。

[0098] 在本实施例中, 步骤S51具体包括以下步骤:

[0099] 步骤S511: 根据表盘真实字符图像构造一张以0、1、2、3、4、5、6、7、8、9、0为顺序上下排列的图像;

[0100] 步骤S512: 以一个大小为 $m \times n$ 的窗口对步骤S511得到的图像进行步长为1的上下移动裁剪并保存, 同时将其中全字的模板剔除, 得到z张半字模板。

[0101] 较佳的, 本实施例是将所述输入的带有全字半字标签的灰度图根据标签属性进行分开识别, 若为全字则用卷积神经网络进行识别, 若为半字则通过模板匹配进行识别: 全字一般位于表盘的高位, 若识别错误误差将会极大, 模板匹配具有局限性, 对于平移或旋转的图像的识别准确率较低, 所以应用识别准确率更高且能适应各种情况的卷积神经网络, 提高准确性和鲁棒性; 半字的情况较全字更为复杂, 若要用卷积神经网络则所需的样本较多, 考虑实际水表采集困难;

[0102] 特备的, 本实施例包括图像二值化、卷积神经网络全字识别、局部二值化、保留两个最大连通域、模板匹配半字识别这几个步骤;

[0103] 其中,卷积神经网络全字识别步骤是通过构建一个“卷积层-池化层-卷积层-池化层-……-卷积层-池化层-全连接层”的卷积神经网络对带有标签训练集和数据集进行训练,得到该网络的相关参数并保存,再将待识别的全字灰度图输入到该网络中,就能得到识别结果;

[0104] 局部二值化步骤采用 $k \times k$ 的候选框在灰度图中计算出最大值最小值的均值差值,通过最大值-最小值的差值与初始值 S 的比较判断该候选框像素值是否相差太大,即是否背景和目标同时存在,进而确定改点的阈值是最大值-最小值的一半,并利用最大值与最小值的差值判断窗口内的像素点是否处于同一背景\目标下,进而选择改像素点的阈值是最大值和最小值的均值还是初始值 T ,二值化使得图像在视觉上呈现黑或白的效果,可从整张半字图像中分离背景和目标;

[0105] 保留两个最大连通域步骤是通过八领域法将待识别的半字二值图的连通域以相同自然数标记出来,然后通过计算自然数的个数来确定两个最大连通域的位置,将该位置下的像素值改为1,其他位置的像素值改为0,根据半字在半字图像中的连通域最大这一特性来筛选出要进行识别的半字内容,更好地排除一些小连通域的杂质干扰,为后面的识别提供了良好的基础。

[0106] 模板匹配半字识别步骤是将已经保留两个最大连通域的待识别半字二值图与半字模板对应像素点一一相减取绝对值后相加,得到待识别半字与模板的距离,找到最小的距离即最相似的模板,该模板的标签即最终识别结果。

[0107] 较佳的,本实施例结合说明书附图对以下步骤进行进一步描述。

[0108] 步骤S1:如图2所示,获取五张带有全字半字标签的分割好的二值图,并输入到步骤S2中。其中,标签0代表半字,标签1代表全字。所述输入图像的大小均为 32×24 ,其中,32是所述输入图像的行数,24是所示输入图像的列数。

[0109] 步骤S2具体为:

[0110] 步骤S2:将标签筛选分类,标签0的半字灰度图进入步骤S4,标签为1的全字灰度图进入步骤S3,根据图2所示,将图2中的(1)(2)(3)(4)四张全字灰度图输入到步骤S3处理,将图2中的(5)的半字灰度图输入到步骤S4处理。

[0111] 如图3所示,在本实施例中,所述步骤S3具体包括以下步骤:

[0112] 步骤S31:输入测试集图片和标签以及训练集图片和标签,图片和标签一一对应,且将标签用独热码表示,即每一个数据所对应的标签为一个向量,全字的标签为0,1,2,3,4,5,6,7,8,9,如图4所示,在本实施例中,测试集图片和训练集图片均为106040张,部分训练集图片如图5所示;

[0113] 步骤S32:用TensorFlow训练一个结构为“卷积层1-池化层1-卷积层2-池化层2-全连接层”的前向传输的卷积神经网络,如图6所示。

[0114] 所述步骤S32所涉及的卷积神经网络架构如下:

[0115] 输入层:由于输入的是一个 32×24 的灰度图像,其在二维空间的维度是 32×24 。一个像素点相当于一个神经元,故输入层的维度就是三维 $[32, 24, 1]$;

[0116] 卷积层1:本实施例中第一个卷积层的尺寸是 5×5 ,卷积深度为32,设定步长为2(卷积前先将原始输入图像的上下左右2行以像素0填充)相当于用32个 5×5 的滑动窗口以步长2与输入图像在该窗口下的像素相卷积,得到的图像大小为 $[32, 24, 32]$;

[0117] 池化层1:本实施例中设定第一个池化步长为2,池化尺寸为 2×2 ,池化方式为最大池化,即以步长2在尺寸为 2×2 的滑动窗口扫描卷积层1得到的图像像素,存储该窗口下的最大值,故该层池化后得到的的图像大小为[16,12,32];

[0118] 卷积层2:本实施例中第二个卷积层的尺寸是 5×5 ,卷积深度为64,设定步长为2(卷积前先将池化层1得到的图像的上下左右2行以像素0填充)相当于用64个 5×5 的滑动窗口以步长2与输入图像在该窗口下的像素相卷积,得到的图像大小为[16,12,64];

[0119] 池化层2:本实施例中设定第一个池化步长为2,池化尺寸为 2×2 ,池化方式为最大池化,即以步长2在尺寸为 2×2 的滑动窗口扫描卷积层2得到的图像像素,存储该窗口下的最大值,故改层池化后得到的图像大小为[8,6,32];

[0120] 全连接层1:本实施例中先将池化层2得到的图像[8,6,32]平坦化成[1,8 \times 6 \times 32],即1536个神经元与 $32 \times 24 = 768$ 个神经元全连接;

[0121] 全连接层2(即输出层):在本实施例中全字共有10个标签,用独热码表示,全连接层1的768个神经元与全连接层2的10个神经元相连接,输出应为 1×10 大小的向量;

[0122] 在本实施例中,采用的学习率为0.001,丢弃率为0.5,训练过程如图7所示。

[0123] 步骤S33:保存该卷积神经网络的各项权重值;

[0124] 步骤S34:将输入的全字二值图中输入到训练好的卷积神经网络中进行识别,得到图2中的(1)(2)(3)(4)的识别结果:9、9、5、7。

[0125] 在本实施例中,所述步骤S4具体包括以下步骤:

[0126] 步骤S41:将所述半字灰度图M,即图2中的(5)进行二值化处理,得到图像待识别半字二值化图像 M_1 ,如图8所示,大小为 32×24 ;

[0127] 步骤S42:将步骤S41得到的用 M_1 八领域的方式找出连通域,并以自然数1、2、3、4、6、7、8标记连通域,如图9所示,大小为 32×24 ;

[0128] 步骤S43:计算步骤S42标记的相同自然数的个数,自然数1、2、3、4、5、6、7、8的个数分别30、17、16、100、99、25、28、18,个数最多的自然数为4、5,将该两个自然数所在的连通域的像素改为1,其余位置的像素改为0,得到所述图像 M_2 ,如图10所示,大小为 32×24 ;

[0129] 如图11所示,在本实施例中,所述步骤S41具体包括以下步骤:

[0130] 步骤S411:(在灰度图中,将对识别有用的信息称为目标,如数字区域;将对识别无用的信息称为背景,如数字像素点以外的白色区域)设定一个 7×7 大小的候选框作为局部

二值化每次选取像素的滑动窗口,设定两个初始阈值: $T = \frac{256}{2} = 128$, $S = 15$ 。T为0-255灰度

值的中点,用来作为当窗口只存在背景或目标时判断是否是目标的阈值;S为判断该滑动窗口中是否背景和背景同时存在的阈值,记 $f(x,y)$ 为图像M在第x行y列的像素灰度值(以图像M左上角为原点)。记 $f'(x,y)$ 为经过局部二值化处理后的图像 M_1 在第x行y列的像素后的值。

[0131] 步骤S412:用 7×7 的候选框以步长为1遍历所述灰度图像M;令

$T_1(x,y) = \frac{1}{2} \times (\max + \min)$, $T_2(x,y) = \max - \min$,其中,max为灰度图像在该滑动窗口范围下的

像素点灰度值最大值,min为灰度图像在该候选框范围下的像素点灰度值最小值, T_1 表示该候选框内灰度值最大值与灰度值最小值的均值, T_2 表示该候选框内灰度值最大值与灰度值最小值的差值;

[0132] 步骤S413:比较 T_2 与 S 的大小;若 $T_2(x,y) > S$,则说明候选框中灰色像素值波动范围大,目标和背景同时存在,则将灰度图像上该像素点的灰度值 $f(x,y)$ 与 $T_1(x,y)$ 对比,若 $f(x,y) > T_1(x,y)$,则 $f'(x,y) = 1$;若 $f(x,y) < T_1(x,y)$,则 $f'(x,y) = 0$;若 $T_2(x,y) < S$,则说明候选框中灰色像素值波动范围小,只存在背景或目标,则将所述灰度图像上该像素点的灰度值 $f(x,y)$ 与阈值 T 值相比;若 $f(x,y) > T$,则 $f'(x,y) = 1$;若 $f(x,y) < T$,则 $f'(x,y) = 0$ 。

[0133] 在本实施例中,所述步骤S5具体包括以下步骤:

[0134] 步骤S51:制作半字模板,个数为 k ;

[0135] 步骤S52:将所述待识别半字二值图 M_2 (即图10)中各个像素点的值按照其各自所在的位置构成一个大小为 32×24 的矩阵 A ;

[0136] 步骤S53:将所述半字模板二值图中各个像素点的值按照其各自所在的位置构成一个 32×24 的矩阵 C ;

[0137] 步骤S54:将所述矩阵 A 和所述矩阵 C 两个矩阵逐点进行相减取绝对值操作,然后将所有的值相加,记为 h ;

[0138] 步骤S55:重复执行步骤S53-S54 k 次,找到 $h_{\min} = 45$ 即半字模板与待识别半字 P 的最小汉明距离,该最小汉明距离下的半字模板的标签5即为待识别半字 P 的识别结果,如图12所示。

[0139] 在本实施例中,所述步骤S51具体包括以下步骤

[0140] 步骤S511:根据表盘真实字符图像构造一张以01234567890按顺序上下排列的图像;

[0141] 步骤S512:以一个窗口为 32×24 的大小对步骤S511所述的图像进行上下移动步长为1的裁剪并保存,同时将其中全字的模板剔除,得到260张半字模板,如图13所示。

[0142] 在本实施例中,所述步骤S6具体包括以下步骤:

[0143] 步骤S6:按照步骤S1输入的图像顺序,依次输出识别结果,如图14所示,本实施例识别结果为:99575,识别结果与实际相符合,识别正确。

[0144] 本领域内的技术人员应明白,本申请的实施例可提供为方法、系统、或计算机程序产品。因此,本申请可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且,本申请可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。

[0145] 本申请是参照根据本申请实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器,使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0146] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中,使得存储在该计算机可读存储器中的指令产生包括指令装置的制品,该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0147] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上,使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0148] 以上所述,仅是本发明的较佳实施例而已,并非是对本发明作其它形式的限制,任何熟悉本专业的技术人员可能利用上述揭示的技术内容加以变更或改型为等同变化的等效实施例。但是凡是未脱离本发明技术方案内容,依据本发明的技术实质对以上实施例所作的任何简单修改、等同变化与改型,仍属于本发明技术方案的保护范围。

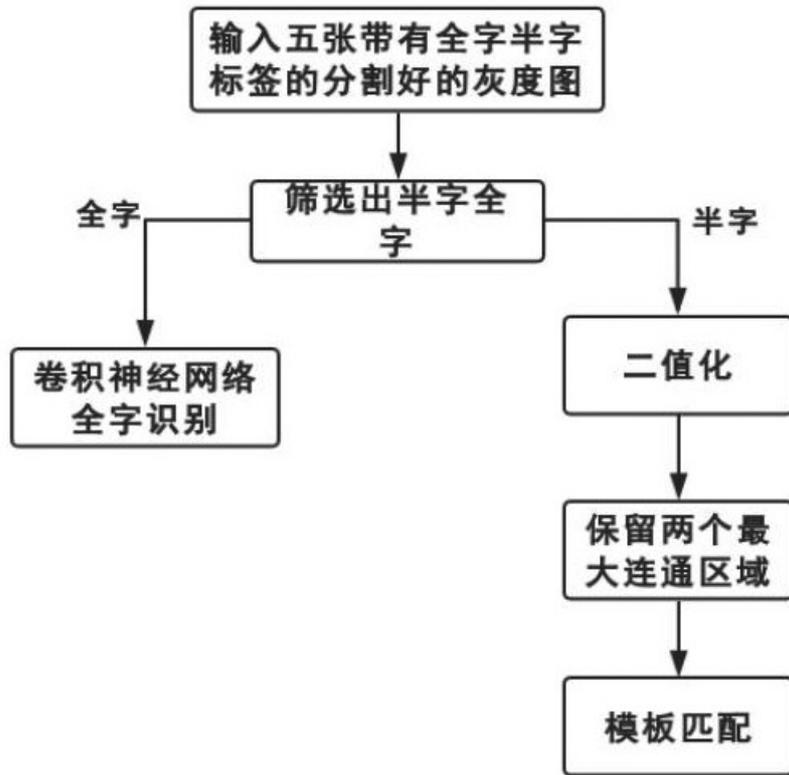


图1

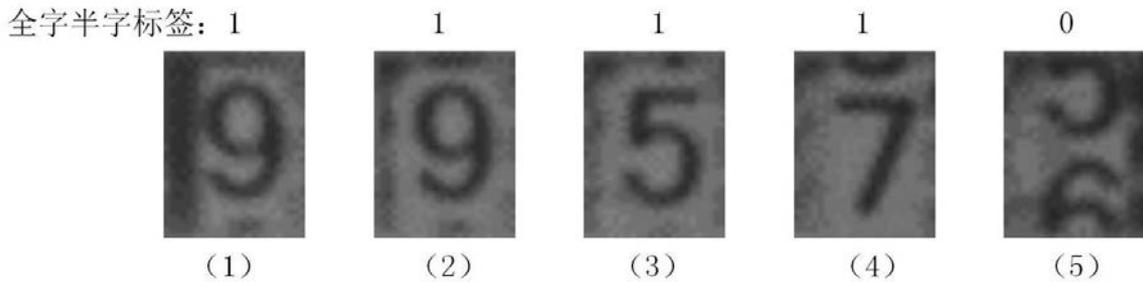


图2

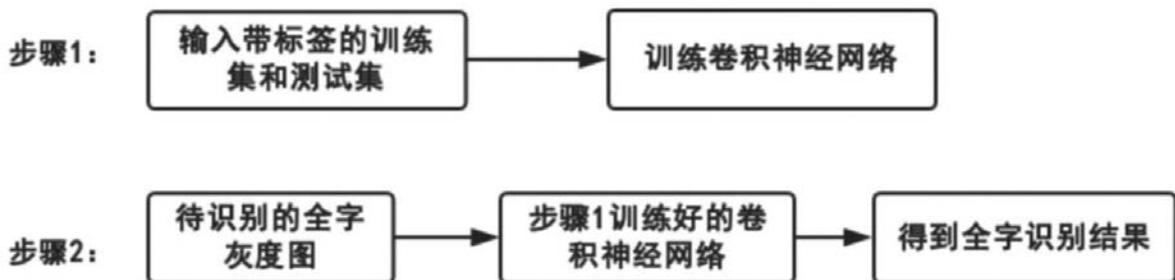


图3

0 : [1,0,0,0,0,0,0,0,0,0]
1 : [0,1,0,0,0,0,0,0,0,0]
2 : [0,0,1,0,0,0,0,0,0,0]
3 : [0,0,0,1,0,0,0,0,0,0]
4 : [0,0,0,0,1,0,0,0,0,0]
5 : [0,0,0,0,0,1,0,0,0,0]
6 : [0,0,0,0,0,0,1,0,0,0]
7 : [0,0,0,0,0,0,0,1,0,0]
8 : [0,0,0,0,0,0,0,0,1,0]
9 : [0,0,0,0,0,0,0,0,0,1]

图4



图5

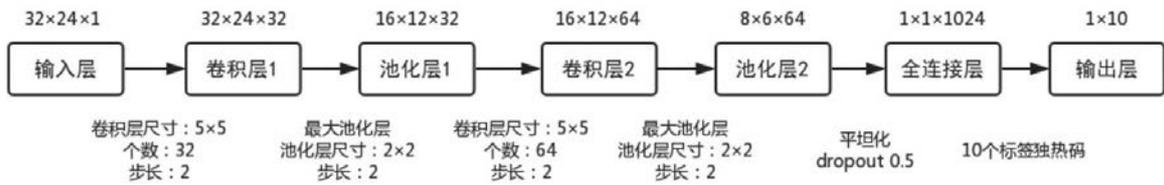


图6

```

第 3100 步的 训练损失=0.00052, 测试精度=0.98531
第 3200 步的 训练损失=0.00120, 测试精度=0.98697
第 3300 步的 训练损失=0.00096, 测试精度=0.98853
第 3400 步的 训练损失=0.00041, 测试精度=0.99000
第 3500 步的 训练损失=0.06956, 测试精度=0.99139
第 3600 步的 训练损失=0.00012, 测试精度=0.99270
第 3700 步的 训练损失=0.00122, 测试精度=0.99395
第 3800 步的 训练损失=0.00101, 测试精度=0.99513
第 3900 步的 训练损失=0.00078, 测试精度=0.99625
第 4000 步的 训练损失=0.00279, 测试精度=0.99732

```

图7



图8

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	1	1	1	0	0	5	5	5	5	5	5	5	5	5	5	5	0	0	0	0	0	0	6
3	0	1	1	1	1	0	0	5	5	5	5	5	5	5	5	5	5	5	0	0	0	0	0	0	6
4	0	1	1	1	1	0	0	5	5	5	5	5	5	0	5	5	5	5	5	0	0	0	0	6	6
5	0	1	1	1	1	0	0	0	5	5	5	0	0	0	0	0	5	5	5	5	0	0	0	6	6
6	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	5	5	5	5	0	0	0	6	6
7	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	5	5	5	0	0	0	6	6
8	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	5	5	5	0	0	0	6	6
9	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	5	5	5	0	0	0	6	6
10	1	1	0	0	0	0	0	5	5	5	0	0	0	0	0	0	5	5	5	5	0	0	0	6	6
11	1	1	0	0	0	0	0	5	5	5	5	0	0	0	0	0	5	5	5	5	0	0	0	0	6
12	0	0	0	0	0	0	0	5	5	5	5	0	0	0	0	5	5	5	5	0	0	0	0	0	6
13	0	0	0	0	0	0	0	0	0	5	5	5	5	5	5	5	5	5	0	0	0	0	0	7	7
14	0	0	0	0	0	0	0	0	0	0	5	5	5	5	5	5	5	0	0	0	0	0	0	7	7
15	2	2	2	0	0	0	0	0	0	0	5	5	5	5	5	0	0	0	0	0	0	0	0	7	7
16	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	7
17	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	7
18	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	7
19	2	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	0	0	0	0	0	0	7	7
20	2	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	0	0	0	0	0	7	7
21	2	2	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	0	0	0	7	7
22	2	2	0	0	0	0	0	0	0	0	4	4	4	0	0	0	4	4	4	4	0	0	0	7	7
23	2	2	0	0	0	0	0	0	0	4	4	4	0	0	0	0	4	4	4	4	0	0	0	0	6
24	0	0	0	0	0	0	0	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	6
25	0	3	3	0	0	0	0	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	8	8
26	3	3	3	3	0	0	4	4	4	4	4	0	0	4	4	4	4	0	0	0	0	0	0	8	8
27	3	3	3	3	0	0	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	8	8
28	3	3	3	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	8	8
29	3	3	0	0	0	0	4	4	4	4	4	0	0	0	0	4	4	4	4	4	0	0	0	8	8
30	3	0	0	0	0	0	4	4	4	4	0	0	0	0	0	4	4	4	4	4	0	0	0	8	8
31	0	0	0	0	0	0	0	4	4	4	0	0	0	0	0	0	4	4	4	4	0	0	0	0	8
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6

图9



图10

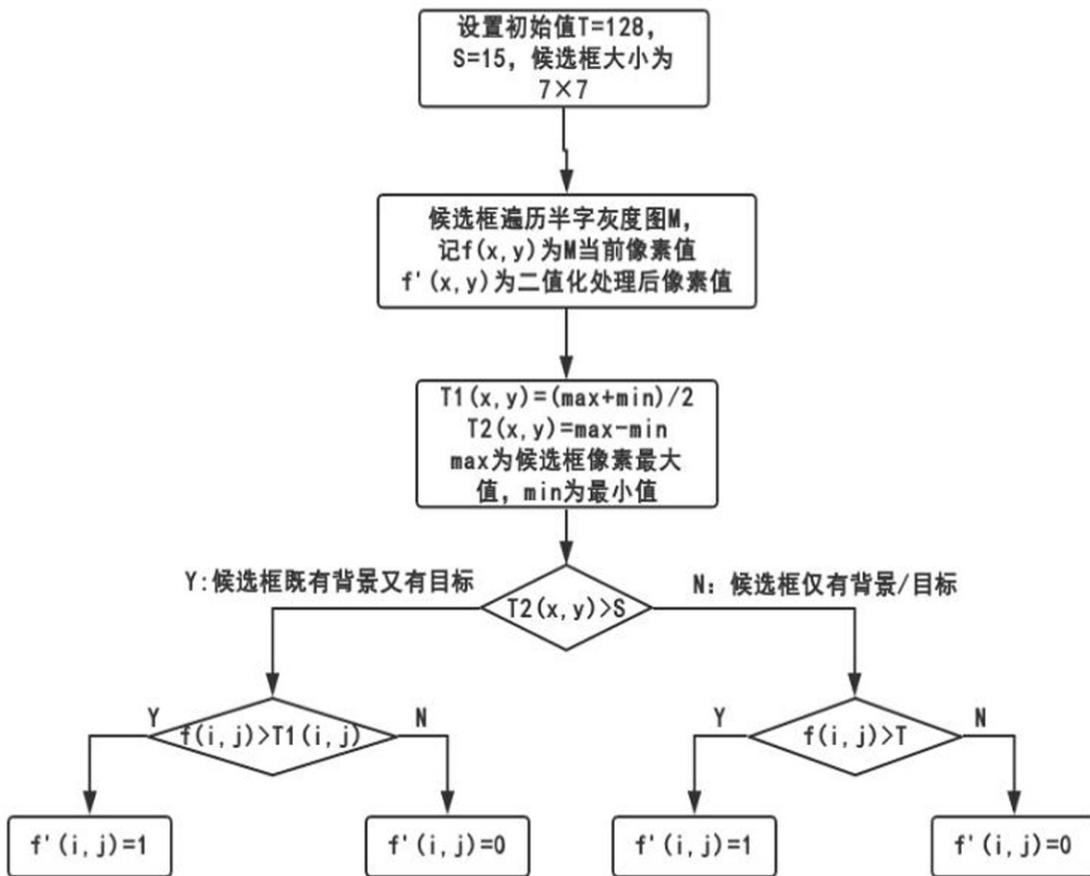


图11



图12

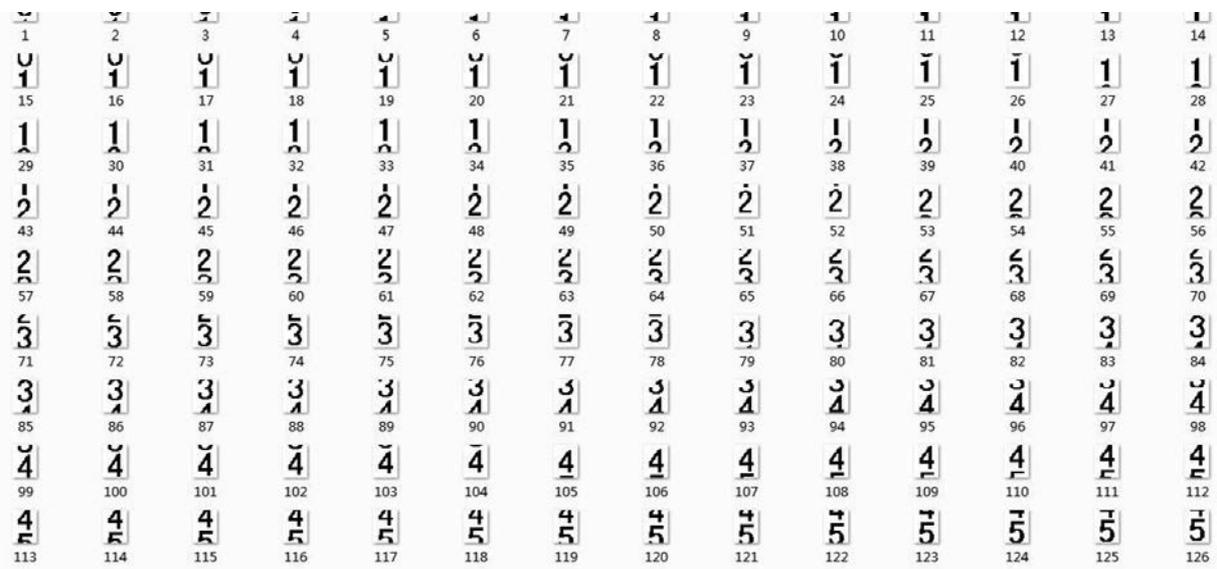


图13

识别结果：9 9 5 7 5



图14