第32卷 第24期 计 算 机 工 程 2006年12月
Vol.32 No.24 Computer Engineering December 2006
·人工智能及识别技术· 文章编号：1000—3428(2006)24—0204—02 文献标识码：A 中图分类号：TP391

车牌识别中基于 Rough 集理论的字符识别

王希雷[1]，王 磊[2]
(1. 天津科技大学计算机科学与信息工程学院，天津 300222；2. 燕山大学机械学院 CAD 中心，秦皇岛 066004)

摘 要：用Rough集理论提取车牌中的文字、字母、数字、短横线的特征，再用这些特征进行模板匹配。该文中的基于Rough集可辨矩阵的特征选择算法，时间复杂度为$O(mn^2)$，改变了过去人们认为基于可辨矩阵的特征选择算法的时间复杂度不低于$O(m^2n^2)$的观点（其中m为数据集中特征/属性的个数，n为数据集中样本的个数）。给出了在车牌识别中的实验结果。

关键词：Rough 集；车牌识别；特征选择；二进制可辨矩阵

Character Recognition of License Plate Recognition Based on Rough Set Theory

WANG Xilei[1], WANG Lei[2]
(1. College of Computer Sci.& Info. Engineering, Tianjin Univ. of Sci. & Tech., Tianjin 300222;
2. Center of CAD, College of Mech. Eng., Yanshan Univ., Qinhuangdao 066004)

【Abstract】 This paper extracts the features of Chinese characters, letters, numbers and short across line based on Rough sets. Result by template matching can be obtained. The time complexity of feature selection algorithm based on Rough sets is $O(mn^2)$. Before, people think that the time complexity of feature selection algorithm based on Rough sets can not be under $O(m^2n^2)$ in which m is the number of features, n is the number of samples in datasets. Finally, the experiment and results in license plate recognition are presented.

【Key words】 Rough sets; License plate recognition; Feature selection; Binary discernibility matrix

模板匹配是计算机视觉的一个基本问题，在大量的实际应用中都有涉及，模板匹配有多种形式，一般是在具有先验知识的情况下进行的。这里的先验知识就是模板的典型形态或特征[1]。 Rough集是近年来发展起来的一种处理不确定和含糊信息的重要工具。在知识发现、机器学习、模式识别等领域得到成功应用。我们考虑使用Rough集进行车牌识别。特征选择和属性约简为同义词，在本文中将不加区别使用。

1 Rough 集基本知识

本文只简单叙述 Rough 集的一些基本知识，更详细的关于 Rough 集的知识请参见文献[2,3]，关于二进制可辨矩阵的详细知识请参见文献[4,5]。

信息表知识表达系统的基本成分是研究对象的集合，关于这些对象的知识是通过指定对象属性（属性、特征、知识是同义词）和它们的属性值来描述的。

决策表是一种特殊的信息表，它表示当满足某些条件时，决策行为应当如何进行。设决策表为T=(U,C,D,V,f)，其中论域$U=\{u_1,u_2,\ldots,u_n\}$，条件属性集$C=\{c_1,c_2,\ldots,c_m\}$，决策属性$D=\{d\}$，则决策表T对应的二进制可辨矩阵MT构造如下：

矩阵的每一列对应一个条件属性，共有m列，每一行对应一个对象对(u_i,u_j)，其中u_i,u_j的决策属性$d(u_i)\neq d(u_j)$，即这一对对象属于不同的决策类。设T对应的二进制可辨矩阵为MT，则MT的行对应在不可分辩关系IND(A)下的可分辨的对象对(u_i,u_j)，它的列对应属性集C中的属性c_{ij}，设$MT=(m_{(i,j)q})$，则：

$$m_{((i,j)q)}=\begin{cases}1 & c_q(u_i)\neq c_q(u_j)\\ 0 & c_q(u_i)=c_q(u_j)\end{cases}$$

由于二进制可辨矩阵采用了二进制的表达形式，其计算比施行等价类计算要快得多，灵活得多。

对于决策表，由于不一致问题的存在，基于可辨矩阵求出的核和约简是存在冗余属性的[6,7]。我们按照文献[5]中方法建立二进制可辨矩阵。

2 基于 Rough 集的特征选择算法

目前的属性约简算法从信息系统中直接求得约简，本文方法，分两步求得约简，有效地降低了算法的时间复杂度。

2.1 第 1 步算法

输入：二进制可辨矩阵 MT；

输出：一个近似约简；

Step1 M ← MT；

Step2 core ← （M中每行仅有一个“1”的行中“1”所在的列的属性）；并消去这些列及这些列中元素“1”所对应行；reduction ← core；计算每行和每列“1”的个数，分别放入Row_i和Col_i中；

Step3 if $M\neq\phi$ then对 $\forall$ $a_i+a_j=a_i$ （$i\neq j$），去掉a_j(a_i,a_j为M中列对应属性) else goto step7；

Step4 reduction ← （M 中每行仅有一个“1”的行中“1”所在的列的属性）；并消去这些列及其元素“1”所对应行；

Step5 reduction ← （M 中含“1”个数最少的行对应的列）；并消去此行及此行中“1”对应列；

基金项目：天津科技大学科学研究基金资助项目（20050226）；天津市科技发展计划基金资助项目（04310951R）

作者简介：王希雷(1973－)，男，硕士，主研方向：Rough 集理论及应用，知识发现；王 磊，学士

收稿日期：2006-03-28 **E-mail：**wxl-cn@163.com

注：若有两行或多行中的“1”的个数最少，则选择“1”对应列中“1”的总数最多的行

Step6 if M≠ ϕ then goto step5 else goto step8；

Step7 输出 reduction，结束，并且不执行第二步算法，reduction 为最终约简；

Step8 输出 reduction。

2.2 第 2 步算法

输入：二进制可辨矩阵；一个近似约简；

输出：一个约简；

Step1 MT $\rightarrow$ B；

Step2 reduction1 $\leftarrow$ reduction-core ；假设 reduction1= $\{a_1,a_2,\ldots,a_{m1}\}$；

Step3 for i= 1 to m1；

Step3.1 reduction-$\{a_i\}$ $\rightarrow$reduction2, 消去B中reduction2 中所有属性对应列，及所有元素“1”对应行；

Step3.2 if B= ϕ then reduction $\leftarrow$ reduction-$\{a_i\}$；

Step4 输出 reduction 为最终结果。

假设决策表T中有m个属性，n个对象，本文算法最坏情况下时间复杂度为$O(mn^2)$（建立二进制可辨矩阵的时间复杂度为$O(mn^2)$，第 1 步算法时间复杂度为$O(m^2)$，第 2 步算法的时间复杂度为$O(m^2)$），在此以前人们普遍认为基于可辨矩阵的属性约简算法的时间复杂度不低于$O(m^2n^2)$。而同类算法中以二进制可辨矩阵为基础的BDMR算法的时间复杂度为$O(n^4+m^2)$，以等价类计算为主的CEBARKCC算法、CEBARKN算法、MIBAK算法时间复杂度分别为$O(mn^2)+O(n^3)$、$O(mn^2)+O(mn^3)$、$O(mn^2)+O(n^3)$，本文的算法的时间复杂度比以上算法都低。而且本文算法采用的是二进制布尔运算，与CEBARKCC、CEBARKN、MIBARK等使用的等价类计算相比，计算速度要快得多。

3 字符识别部分的模型描述

本文的方法把字符识别分为 3 个部分：(1)利用 Rough 集得出分类规则；(2)用分类规则识别出具体字符；(3)把新字符加入字符库，更新规则。

3.1 分类规则的生成

图 1 描述的是分类规则生成模型。

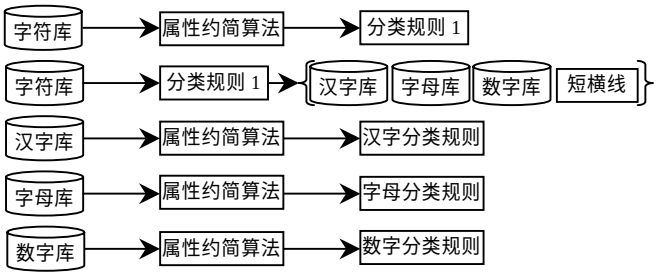

图 1 分类规则生成模型

其中‘字符库’是由经过字符预处理后的字符组成的；‘属性约简算法’是第 2 节中描述的算法（其中的决策属性值是 0、1、2、3、分别表示汉字、英文字母、阿拉伯数字、短横线；条件属性为图形上的点，属性值为 0 或 1）；‘分类规则 1’是通过属性约算法得到的分类规则，分类规则 1 可以把字符库中的字符分为汉字、字母、数字、短横线（新车牌中有一个短横线）；使用分类规则 1 对字符库进行分类，可以把字符库分为‘汉字库’、‘字母库’、‘数字库’、‘短横线’；然后分别应用第 3 节中的属性约简算法（决策属性值为具体字符），即可得出相应的分类规则——汉字分类规则、字母分类规则、数字分类规则（短横线的分类规则包含于分类规则 1 之中）。

分类规则生成算法如下：

输入：字符库中字符形成的二进制可辨矩阵 M；

输出：分类规则 1、汉字分类规则、字母分类规则、数字分类规则；

Step1 对字符库形成的可辨矩阵 M 调用第 3 节中的属性约简算法，得出分类规则 1；

Step2 对字符库中字符应用分类规则 1 进行分类，将其分成 0、1、2、3，分别表示汉字库、字母库、数字库、短横线；

Step3 对汉字库构造可辨矩阵 M0，应用第 3 节中的属性约简算法，得出汉字分类规则；

Step4 对字母库、数字库分别执行 Step3 的操作，得出字母分类规则、数字分类规则；

Step5 分别把分类规则 1，汉字分类规则，字母分类规则，数字分类规则放入相应的规则库。

3.2 字符识别

字符识别部分用‘分类规则 1’对待判断字符进行第 1 次分类，此时，得到此字符是 0、1、2、3 中的哪一类；然后应用对应的汉字、字母或数字分类规则对此字符进行判断，得到此字符具体是哪一个字符。

字符识别算法如下：

输入：经字符预处理后的待判断字符；

输出：判断结果；

Step1 对待判断字符使用分类规则１，得出此字符属于 0、1、2、3 中的哪一类；

Step2 if 此字符属于 0,then 使用汉字分类规则，得出此字符是哪个汉字 goto Step6；

Step3 if 此字符属于 1，then 使用字母分类规则，得出此字符是哪个字母 goto Step6；

Step4 if 此字符属于 2，then 应用数字分类规则，得出此字符是哪一个数字 goto Step6；

Step5 如果此字符属于 3，则此字符是短横线；

Step6 输出结果（即此字符是具体哪一个汉字、字母、数字或短横线）。

3.3 更新规则

当已知新字符（即我们知道结果的新字符）增加到一定量时，就要把新字符加入到字符库中，然后应用属性约简算法得到新的分类规则，来替换旧的分类规则，再重复图 1 中的步骤，更新汉字、字母、数字分类规则。然后使用更新后的规则，利用 3.2 中算法进行字符识别。

4 试验结果

试验总共收集了 1 551 个车牌与文献[8]中的方法，比较结果如下。下面的图表中用 B 表示上述用于比较的算法。表 1、表 2 是含大量模糊、残缺车牌的实验数据，表 3、表 4 是几乎不含模糊、残缺车牌的实验数据。其中，车牌数量表示测试用的车牌的数量， a 表示汉字识别率/时间，b 表示字母识别率/时间，c 表示数字识别率/时间，d 表示短横线识别率/时间（%/ms）。试验分两部分，表 1、表 2 表示的是第 1 部分实验结果；表 3、表 4 表示的是第 2 部分实验的结果。

第 1 部分实验采用的数据是从 1 551 个车牌中随机抽取的 3 组数据，分别为 200，400，800，再把这些数据分别分为 2 组，一组用于计算分类规则，另一组用于测试。即计算分类规则的数据分别为 100，200，400；测试数据分别为 100，

(下转第 253 页)