

# 武汉理工大学

数学建模暑期培训论文

第 1 题

基于 xxxxxxxx 模型

---

第 10 组

姓名

刘子川（组长）

程宇

祁成

方向

编程

建模

写作

2020 年 8 月 10 日

## 摘要

控制高压油管的压力变化对减小燃油量偏差,提高发动机工作效率具有重要意义。本文建立了基于质量守恒定理的微分方程稳压模型,采用二分法、试探法以及自适应权重的蝙蝠算法对模型进行求解。//

针对问题一,建立基于质量守恒定律的燃油流动模型,考察单向阀开启时间对压力稳定性的影响。综合考虑压力与弹性模量、密度之间的关系,提出燃油压力-密度微分方程模型和燃油流动方程。本文采用改进的欧拉方法对燃油压力-密度微分方程求得数值解;利用二分法求解压力分布。综合考虑平均绝对偏差等反映压力稳定程度的统计量,求得直接稳定于 100MPa 的开启时长为 **0.2955ms**,在 2s、5s 内到达并稳定于 150MPa 时开启时长为 **0.7795ms**、**0.6734ms**,10s 到达并稳定于 150MPa 的开启时长存在多解。最后对求解结果进行灵敏度分析、误差分析。//

针对问题二,建立基于质量守恒定律的泵-管-嘴系统动态稳压模型,将燃油进入和喷出的过程动态化处理。考虑柱塞和针阀升程的动态变动,建立喷油嘴流量方程和质量守恒方程。为提高角速度求解精度,以凸轮转动角度为固定步长,转动时间变动步长,采用试探法粗略搜索与二分法精细搜索的方法求解,求得凸轮最优转动角速度 **0.0283rad/ms** (转速 **270.382 转/分钟**),并得到该角速度下高压油管的密度、压力周期性变化图。对求解结果进行误差分析与灵敏度分析,考察柱塞腔残余容积变动对高压油管压力稳态的影响。//

针对问题三,对于增加一个喷油嘴的情况,改变质量守恒方程并沿用问题二的模型调整供、喷油策略,得到最优凸轮转动角速度为 **0.0522rad/ms** (**498.726 转/分钟**);对于既增加喷油嘴又增加减压阀的情况,建立基于自适应权重的蝙蝠算法的多变量优化模型,以凸轮转动角速度、减压阀开启时长和关闭时长为参数,平均绝对偏差 MAD 为目标,在泵-管-嘴系统动态稳压模型的基础上进行求解,得到最优参数:角速度 **0.0648 rad/ms** (**619.109 转/分钟**)、减压阀的开启时长 **2.4ms** 和减压阀的关闭时长 **97.6ms**。//

本文的优点为:1. 采用试探法粗略搜索与二分法精细搜索结合的方法,降低了问题的求解难度。2. 以凸轮转动角度为固定步长,对不同角速度按照不同精度的时间步长求解,大大提高了求解的精确度。3. 针对智能算法求解精度方面,采用改进的蝙蝠算法,使速度权重系数自适应调整,兼顾局部搜索与全局搜索能力。

关键词: 微分方程 微分方程 微分方程 微分方程

# 目录

一、 问题重述.....	1
1.1 问题背景.....	1
1.2 问题概述.....	1
二、 模型假设.....	2
三、 符号说明.....	2
四、 数据预处理.....	2
4.1 归一化处理.....	2
4.1.1 灰度值处理 .....	2
4.1.2 外框去除与尺度统一 .....	3
4.2 图像强化处理.....	4
4.2.1 滤波 .....	5
4.3 二值化处理.....	5
4.3.1 场方向估计 .....	5
4.3.2 二值化 .....	6
五、 问题一模型的建立与求解.....	6
5.1 问题描述与分析.....	6
5.2 模型的建立.....	7
5.3 模型的求解.....	7
5.4 实验结果及分析.....	7
六、 问题二模型的建立与求解.....	7
6.1 问题描述与分析.....	7
6.2 模型的建立.....	8
6.3 模型的求解.....	8
6.4 实验结果及分析.....	8
七、 问题三模型的建立与求解.....	10
7.1 结果分析.....	10
八、 灵敏度分析.....	10
九、 模型的评价.....	10
9.1 模型的优点.....	10

9.2 模型的缺点.....	10
9.3 模型改进.....	10
附录 A 数据可视化的实现.....	<b>12</b>

# 一、问题重述

## 1.1 问题背景

自动指纹识别系统 (automated fingerprint identification system, 简称 AFIS) 有着广泛的应用背景。目前对自动指纹识别系统的研究主要有 3 个方面, 即图像增强、指纹分类和细节匹配。一般可以分成” 离线部分” 和” 在线部分” 两个部分。如图 1 所示, 离线部分包括用指纹采集仪采集指纹、提取出细节点、将细节点保存到数据库中形成指纹模板库等主要步骤。在线部分包括用指纹采集仪采集指纹、提取出细节点、然后将这些细节点与保存在数据库中模板细节点进行匹配, 判断输入细节点与模板细节点是否来自同一个手指的指纹<sup>[1, 3]</sup>。指纹分类一般是用在大规模的指纹库中, 作为细节匹配中减少搜索范围的步骤使用。指纹图像一般占用较多的空间, 且图像中的像素信息并不适合计算机进行分析或匹配。为实现计算机自动识别, 需要有一种方法来描述指纹的内在结构、具体形态和其它特征并将其用最少的字节数来存储于计算机中。此计算机系统可扫描犯罪现场采集的指纹, 并且与州、地区、国家之间执法机关采集的数百万指纹档案互相比对<sup>[2]</sup>。指纹由专家追踪后, 经计算机扫描, 得到许多细节来和数据库里其它指纹比对, 列出相符合的百分比来让鉴识人员得知可能的相符人选<sup>1</sup>。任何计算机比对的结果, 都会经指纹专家比较与此指纹相关的样本来验证。



图 1 自动指纹识别系统框图

## 1.2 问题概述

围绕相关附件和条件要求, 试根据附件中的 16 幅指纹图像, 不借助现有的指纹相关软件, 依次提出以下问题:

<sup>1</sup> <https://baike.baidu.com/item/AFIS/2851410?fr=aladdin>

**编码：**给出一种用不超过 200 字节（下面称为“指纹密码”）来刻画描述指纹基本特征的表示方法，介绍其数学原理。

**匹配：**将你的方法编程实现，对附件中的每一幅指纹都给出其“指纹密码”的表示。基于你找到的这些指纹表示，你能否给出一种方法比较不同指纹间的异同及相似程度？

**应用：**你能否对附件中的 16 个指纹进行对比和归类？请给出你对比及分类的依据和结果。

## 二、模型假设

- (1)
- (2)
- (3)
- (4)

## 三、符号说明

符号	说明
$P_n$	20 个站点
$P_n$	20 个站点
$P_n$	20 个站点

注：表中未说明的符号以首次出现处为准

## 四、数据预处理

在正式搭建模型前，本文首先将图片进行预处理，以方便进行后续运算。经过归一化处理，??? 和??? 后，即可得到相同规格的二值化图片。

### 4.1 归一化处理

#### 4.1.1 灰度值处理

鉴于各个指纹图像的纹理深浅不统一，首先对每张图片做灰度值归一化处理。对于图片  $P_i$ ，其第  $x$  行第  $y$  列的像素点的灰度值可表示为  $P_i(x, y) \in [0, 255]$ 。即当  $P_i(x, y) = 255$  时， $P_i(x, y)$  为纯白色像素点；当  $P_i(x, y) = 0$  时， $P_i(x, y)$  为纯黑色像素点。对于任

意图片  $P_i$ , 有

$$P_{i,max} = \max_{1 \leq x \leq n, 1 \leq y \leq m} P_i(x, y),$$

$$P_{i,min} = \min_{1 \leq x \leq n, 1 \leq y \leq m} P_i(x, y),$$

其中  $n$  和  $m$  分别表示图片的行数和列数。即  $P_{i,max}$  与  $P_{i,min}$  分别表示图片中像素的最大灰度值和最小灰度值。对图片执行线性归一化操作如下

$$P_i(x, y)' = \frac{255}{P_{i,max} - P_{i,min}} (P_i(x, y) - P_{i,min})$$

即将图片  $P_i$  像素点灰度值线性放缩于  $[0, 255]$  间, 得到灰度值归一化图片  $P_i'$ 。

#### 4.1.2 外框去除与尺度统一

对于图片  $P_i$ , 我们将去除其空白外框以方便后续学习策略运算。设定边界阈值为  $threshold_N$ , 当图片  $P_i$  最上方的某一行和最下方的某一行的非空白像素数大于  $threshold_N$  时, 将其定义为新的上边界或下边界, 及可得到行边界位置如下

$$fringe_{x1} = \max(x)_{(\sum_{i=1}^m (P'(x,i) < 255)) > threshold_N},$$

$$fringe_{x2} = \min(x)_{(\sum_{i=1}^m (P'(x,i) < 255)) > threshold_N}$$

其中  $fringe_{x1}$  与  $fringe_{x2}$  分别为新定义上边界与下边界, 同理有列边界位置如下

$$fringe_{y1} = \min(y)_{(\sum_{i=1}^n (P'(i,y) < 255)) > threshold_N},$$

$$fringe_{y2} = \max(y)_{(\sum_{i=1}^n (P'(i,y) < 255)) > threshold_N}$$

其中  $fringe_{y1}$  与  $fringe_{y2}$  分别为新定义左边界与右边界。即去除外框后的图片可表示为  $P'' = P'(fringe_{x1} : fringe_{x2}, fringe_{y1} : fringe_{y2})$ 。即只保留新定义边界内的像素点并删除剩余像素点即可得到去除外框后的图片  $P''$ 。定义标准行数为  $N$ , 标准列数为  $M$ , 将去除外框后的图片  $P''$  进行放缩处理可得归一化图片  $P_i^s$  如下

$$P_i^s(x, y) = P''(\text{floor}(\frac{fringe_{x1} - fringe_{x2}}{N}x), \text{floor}(\frac{fringe_{y2} - fringe_{y1}}{M}y))$$

$$x \in [1, N], y \in [1, M]$$

其中  $\text{floor}$  表示向下取整，即通过灰度值处理、外框去除与尺度统一后即可将原始指纹图片数据  $P_i$  转换为灰度值范围与图片尺度统一的归一化图片数据  $P_i^s(x, y)$ 。

## 4.2 图像强化处理

利用 Grunward-Letnikov 微分算子，构造自适应函数，采集梯度信息和计算信息熵，从而确定微分阶数。为了确定自适应分数阶微分阶数  $v$ ，需要确定每个像素的梯度  $G$  和信息熵  $E$ 。梯度  $G$  反映指纹像素特征在感兴趣区域 (ROI, Region of Interest) 上的突变情况，取近似梯度模值为

$$G[f(x, y)] = \max \left\{ \left| \frac{\partial f}{\partial x} \right|, \left| \frac{\partial f}{\partial y} \right| \right\} \quad (1)$$

另一个描述图像边缘纹理变化的指标是信息熵  $E$ ，即所有可能发生事件的信息量期望之和，其数学描述为

$$E = - \sum_{i=1}^n p(x_i) \cdot \log_2(p(x_i)) \quad (2)$$

其中， $p(x_i)$  是事件  $x_i$  发生的概率， $n$  是可能发生事件的个数。对信息熵进行归一化处理，可以获得取值范围为  $[0, 1]$  的信息熵

$$E' = \frac{E - E_{\min}}{E_{\max} - E_{\min}} \quad (3)$$

综合梯度模值和信息熵，构造关系函数  $v = w_1 \cdot G + w_2 \cdot E' + w_3$ ，即微分阶数  $v$  由梯度和信息熵共同决定，梯度模值和信息熵越大，反映该指纹像素需要增强的程度越大，更可能为纹理区域或边缘，分数阶微分阶数取值也越大。此问题中，可算得  $v = 3$ 。将阶数代入利用 Grunward-Letnikov 分数阶微分近似表达式，即

$$\frac{\partial^v f(x, y)}{\partial x_i^v} \doteq \sum_{k=0}^{t-\alpha} (-1)^k \binom{v}{k} f(x_i - k) \quad (4)$$

其中  $f(x, y)$  是图像坐标为  $(x, y)$  的像素灰度值， $t$  和  $\alpha$  分别是微分上限和下限， $x_i (i = 0, 1, \dots, 7)$  是不同的偏微分方向，此处取以  $x$  轴正向为起始方向，顺时针旋转一周均匀分部的 8 个方向。为方便计算，对 8 个方向的分数阶偏微分近似分别保留前三项进行三层  $5 \times 5$  掩膜构造，并进行归一化赋权。

对每个方向  $x_i$ ，以掩膜中心点为作用点， $5 \times 5$  掩膜  $w(x, y)$  中三层权重分别为分数阶微分近似的前三项系数，则掩膜中总权重为分数阶微分近似的前三项系数与每层掩膜元素个数的乘积之和，即  $4v_2 - 12v + 8$ 。对权重归一化，可得中心点、第一层、第二层



掩膜元素权重分别为  $w_0 = \frac{8}{4v_2 - 12v + 8}$ ,  $w_1 = \frac{-v}{4v_2 - 12v + 8}$ ,  $w_2 = \frac{v}{16(v - 2)}$ 。在规模为  $M \times N$  的图像上, 对距离中心点小于  $M - \frac{n-1}{2}$  和  $N - \frac{n-1}{2}$  的像素进行  $n \times n$  掩膜操作, 其他像素进行灰度保留。输出图像可以表示为:

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b f(x+s, y+t) \cdot w(x, y) \quad (5)$$

其中,  $w(x, y)$  是构造的  $5 \times 5$  掩膜,  $a = b = \frac{n-1}{2}$ 。

#### 4.2.1 滤波

### 4.3 二值化处理

#### 4.3.1 场方向估计

一般指纹图像都有较为清晰的场方向, 对于图像上的每一个像素  $P(x, y)$ , 为了确定其在该像素处脊线的方向, 以该像素为中心的  $9 \times 9$  像素矩阵  $u_{P(x, y)}$  内, 分别计算 8 个方向上的灰度平均值, 即对于

$$u_{P(x, y)} = \begin{bmatrix} P(x-4, y-4) & \cdots & P(x, y-4) & \cdots & P(x+4, y-4) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ P(x-4, y) & \cdots & P(x, y) & \cdots & P(x+4, y) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ P(x+4, y+4) & \cdots & P(x, y+4) & \cdots & P(x+4, y+4) \end{bmatrix}$$

若  $u_{P(x, y)}$  中存在若干像素点超出原图像边界时, 记超出边界的像素点为  $P_o$  并使得  $\{P_o\} = 255$ 。并依照图??中的方向, 分别计算 8 个方向上的灰度平均值得到  $Gmean[i]$ , 然后将这 8 个平均值按两两垂直的方向分成 4 组, 0 和 4 一组、1 和 5 一组、2 和 6 一组、3 和 7 一组, 计算每组中两个平均值的差值。

$$Gdiff[j] = |Gmean[j] - Gmean[j+4]|, j = 1, 2, 3, 4,$$

差值绝对值相差最大的方向即为脊线方向与其法向, 即

$$iDir = \begin{cases} \arg(\max_i(Gdiff[j])), \left| P(x, y) - \max_i(Gdiff[j]) \right| < \left| P(x, y) - \max_i(Gdiff[j+4]) \right| \\ \arg(\max_i(Gdiff[j])) + 4, otherwise \end{cases}$$

武汉理工大学

图 2 问题一思维流程图

#### 4.3.2 二值化

得到图像每个像素处的方向场后,再依据方向场来对图像进行二值化。若像素  $P(x, y)$  处的脊线方向为  $iDir$ , 先用估计方向场的方法计算该像素处在方向  $iDir$  和其法线  $iVar = iDir + 4$  的灰度平均值  $Gmean[iDir]$  和  $Gmean[iVar]$ , 基于此将该像素二值化为

$$P(x, y) = \begin{cases} 255, Gmean[iDir] \geq Gmean[iVar] \\ 0, otherwise \end{cases}$$

其中 255 为二值图像中图像背景和谷线的灰度值, 0 为二值图像中图像脊线的灰度值。

## 五、问题一模型的建立与求解

### 5.1 问题描述与分析

问题一要求

其思维流程图如图 3 所示:

武汉理工大学

图 3 问题一思维流程图

## 5.2 模型的建立

$$d(p_i, p_j) = |x_i - x_j| + |y_i - y_j|,$$

## 5.3 模型的求解

---

### Algorithm 1: Procedure of Apriori

---

**Input:** item data base:  $D$   
minimum Support threshold:  $Sup_{min}$   
minimum Confidence threshold:  $Conf_{min}$   
**Output:** frequent item sets  $F$

```

1 Initialize
   iteration  $t \leftarrow 1$ 
   The candidate FIS:  $C_t = \emptyset$ 
   The length of FIS:  $length = 1$ 
   for  $i=1$  to  $sizeof(D)$  do
2      $I_i = D(i)$ 
      $n = sizeof(I_i)$ 
     for  $j=1$  to  $n$  do
3       if  $I_i(j) \notin C_t$  then
4          $C_t = C_t \cup I_i(j)$ 
5       end
6     end
7 end
8  $F_t = \{f | f \in C_t, Sup(f) > Sup_{min}\}$ 
   while  $F \neq \emptyset$  do
9      $t = t + 1$ 
      $length = length + 1$ 
      $C_t \leftarrow$  all candidate of FIS in  $F_{t-1}$ 
      $F_t = \{f | f \in C_t, (Sup(f) > Sup_{min}) \cap (Conf(f) > Conf_{min})\}$ 
10 end
11 return  $F_{t-1}$ 

```

---

## 5.4 实验结果及分析

# 六、 问题二模型的建立与求解

## 6.1 问题描述与分析

问题二要求给出比较不同指纹异同的方法。本问中,对第一问所获得的所有行向量的每一维度进行归一化处理,分别计算归一化后的行向量间的谷本系数 (Tanimoto Coefficient)、皮尔森相关系数 (Pearson Correlation Coefficient) 和马哈拉诺比斯距离 (Mahalanobis Distance),最后构造关系函数确定行向量的相似程度。

- 谷本系数 (Tanimoto Coefficient)

Tanimoto Coefficient 主要用于计算符号度量或布尔值度量的个体间的相似度，因为个体的特征属性都是由符号度量或者布尔值标识，因此无法衡量差异具体值的大小，只能获得“是否相同”这个结果，所以 Tanimoto Coefficient 只关心个体间共同具有的特征是否一致这个问题。其数学描述为

$$T(A, B) = \frac{A * B}{\|A\|^2 + \|B\|^2 - A * B} \quad (6)$$

- 皮尔逊相关系数（Pearson Correlation Coefficient）
- 马哈拉诺比斯距离（Mahalanobis Distance）

其思维流程图如图 4 所示：



图 4 问题二思维流程图

## 6.2 模型的建立

## 6.3 模型的求解

## 6.4 实验结果及分析

结果如下表??所示：

表 1 XXXXXXXXXXXXXXXXXXXX

XXXXXXX	XXXXXXX
XXXXXXX	909.80
XXXXXXX	852.60

由表1可知

其各个小车的运输细节图下图所示：

武汉理工大学 武汉理工大学

武汉理工大学 武汉理工大学

图 5 xxxxxxxxxxxxxxxxxxxxxxxxxxxx

## 七、 问题三模型的建立与求解

### 7.1 结果分析

## 八、 灵敏度分析

## 九、 模型的评价

### 9.1 模型的优点

(1)

(2)

### 9.2 模型的缺点

### 9.3 模型改进

## 参考文献

- [1] Davies S G. Touching Big Brother: How biometric technology will fuse flesh and machine[J]. Information Technology & People, 2014, 7(4): 38-47.
- [2] Moses K R, Higgins P, McCabe M, et al. Automated fingerprint identification system (AFIS)[J]. Scientific Working Group on Friction Ridge Analysis Study and Technology and National institute of Justice (eds.) SWGFAST-The fingerprint sourcebook, 2011: 1-33.
- [3] Dror I E, Wertheim K, Fraser-Mackenzie P, et al. The impact of human – technology co-operation and distributed cognition in forensic science: biasing effects of AFIS contextual information on human experts[J]. Journal of forensic sciences, 2012, 57(2): 343-352.
- [4]
- [5]

## 附录 A 数据可视化的实现

第一问画图-python 源代码

---

第二问画图-python 源代码

---