



大连海事大学  
DALIAN MARITIME UNIVERSITY

# 本科毕业论文

题目: 基于低分辨率视频的手势运动方向检测  
Title: Detection of Gesture Motion Direction based on  
Low-Resolution Video

姓名: 阿海  
学号: 222020xxxx  
学院: 理学院  
专业年级班级: 数学与应用数学 2020-1  
指导教师: 海老师 (教授)  
第二导师: 无  
完成日期: 2024 年 5 月

## 摘 要

本文针对手机摄像头所获取的视频文件，进行手势运动方向的检测。针对低端摄像头视频图像的特点，本文采用了基于背景去除和肤色模型的方法对手部区域进行检测，并判别手部运动的方向。

首先，获取视频图像序列，即从视频文件中获取每一帧图像作为待检测的视频图像序列；其次，对获取的视频图像序列中的每一帧图像进行颜色模型转换、背景去除、图像二值化、形态学处理等预处理；然后，利用区域增长方法来检测视频图像序列中的手部连通区域，并计算每帧图像中手部区域的中心；最后根据图像序列中手部区域中心位置的变化来判断手部运动方向。

本文在 Visual c++6.0 开发环境下，借助于 OpenCV 开放平台，设计并实现了基于低端摄像头视频手势运动检测系统，得到了较好的检测效果。

**关键词:** 运动目标检测；颜色模型；区域增长

## ABSTRACT

In this paper, the detection of hand motion direction is performed for the video files acquired by cell phone camera. Aiming at the characteristics of low-end camera video images, this paper adopts a method based on background removal and skin color model to detect the hand region and to discriminate the direction of hand motion.

Firstly, a video image sequence is acquired, i.e., each frame from the video file is acquired as a video image sequence to be detected; secondly, each frame in the acquired video image sequence is preprocessed with color model conversion, background removal, image binarization, morphological processing, etc.; then, a region growing method is used to detect the hand connectivity region in the video image sequence and to calculate the center of the hand region in each frame; finally, the hand region is detected based on the color model in the image sequence and the direction of hand movement is detected based on the color model. The center of the hand region; finally, the hand movement direction is judged based on the change of the center position of the hand region in the image sequence.

In this paper, under Visual c++6.0 development environment, with the help of OpenCV open platform, we designed and realized the hand gesture motion detection system based on low-end camera video, and got a better detection effect.

**Keywords: Moving Target Detection; Color Model; Regional Growth**

# 目 录

|                           |    |
|---------------------------|----|
| 第 1 章 绪论 .....            | 1  |
| 1.1 课题研究的背景及意义.....       | 1  |
| 1.1.1 视频运动目标检测的研究现状 ..... | 1  |
| 1.1.2 运动目标检测技术 .....      | 2  |
| 1.2 本章小结.....             | 3  |
| 第 2 章 基础知识 .....          | 4  |
| 2.1 视频图像预处理.....          | 4  |
| 2.1.1 常用颜色模型 .....        | 4  |
| 2.2 本章小结.....             | 6  |
| 第 3 章 视频图像预处理 .....       | 7  |
| 3.1 引言.....               | 7  |
| 3.2 图像的多种显示方式.....        | 7  |
| 3.3 本章小结.....             | 8  |
| 第 4 章 系统性能分析 .....        | 9  |
| 结论 .....                  | 10 |
| 参考文献.....                 | 11 |
| 致谢.....                   | 12 |
| 附录 1 企业信息表.....           | 1  |
| 附录 2 相关定理证明 .....         | 1  |
| 附录 3 程序代码 .....           | 1  |

# 基于低分辨率视频的手势运动方向检测

## 第 1 章 绪论

本模板只是作为本科论文格式示例作用，为尽可能涵盖《毕业论文撰写规范》规定的内容，部分图片或表格与论文内容无关，该模板论文无研究意义，师生只做格式参考。

### 1.1 课题研究的背景及意义

摄像头 (camera) 又称为电脑相机、电脑眼等，它作为一种视频输入设备，在过去被广泛的运用于视频会议、远程医疗及实时监控等方面。近些年来，随着互联网技术的发展，网络速度的不断提高，再加上感光成像器件技术的成熟，使得摄像头得到了越来越广泛的应用。

#### 1.1.1 视频运动目标检测的研究现状

视频序列中运动目标的检测与跟踪是计算机视觉和图像编码研究领域的一个重要课题，在机器人导航、智能监视系统、交通检测、医学图像处理以及视频图像压缩和传输等领域都有广泛的应用。运动目标检测就是判断视频序列中是否存在运动目标，并确定运动目标的位置。运动目标的提取主要包括运动检测以及目标提取两个步骤，其中运动检测处于整个视觉监视系统的最底层，是各种后续高级处理如目标分类，行为理解等的基础。

在近年来，随着技术的快速发展，多领域的研究都取得了显著的进展。特别是在数据处理、目标检测以及环境科学等领域，新的理论和方法不断涌现，为解决实际问题提供了有力的支持。

首先，在数据处理和导航系统的应用方面，付梦印等人深入探讨了 Kalman 滤波理论及其在导航系统中的应用<sup>[1]</sup>。Kalman 滤波作为一种高效的递归滤波器，通过预测和更新两个步骤，能够在存在不确定性的动态系统中估计出系统状态。这一理论在导航系统中的应用，极大地提高了系统的准确性和可靠性。

在目标检测领域，邓宇的研究为我们提供了复杂背景下运动目标检测技术的深入见解<sup>[2]</sup>。随着计算机视觉和图像处理技术的不断发展，目标检测技术在安防、交通监控等领域的应用越来越广泛。如何在复杂的背景下准确、快速地检测出目标，成为该领域的重要研究方向。

在环境科学领域，张爱茜等人关注了氯代芳香族化合物对羊角月牙藻的毒性及 QSAR 分析<sup>[3]</sup>。这一研究不仅揭示了氯代芳香族化合物对水生生物的潜在危害，

还通过 QSAR（定量结构-活性关系）分析，为预测和评估这类化合物的环境风险提供了有力的工具。

此外，文献 [4] 提出了自适应背景混合模型 (Adaptive Background Mixture Models)，为实时跟踪提供了新的解决方案。这一模型通过对背景图像进行建模和更新，能够有效地区分前景目标和背景，为视频监控、人机交互等领域的研究提供了新的思路。

综上所述，从 Kalman 滤波理论在导航系统中的应用，到复杂背景下的目标检测技术，再到环境科学中的毒性评估和 QSAR 分析，以及实时跟踪中的自适应背景混合模型，这些研究成果不仅丰富了相关领域的理论和方法，也为解决实际问题提供了有力的支持。随着技术的不断进步，相信这些领域的研究将会取得更加显著的成果。

### 1.1.2 运动目标检测技术

运动目标检测技术研究如何完成研究对象（图像序列）中感兴趣的目标区域的“准确定位”问题。

#### 1.1.2.1 光流法

三种传统的运动目标检测算法之一。当物体运动时，在图像上对应物体的亮度模式也在运动，从而称光流是图像亮度模式的视在运动。光流法检测采用了目标随时间变化的光流特性。

#### 1.1.2.2 帧间差分法

三种传统的运动目标检测算法之一。帧间差分式检测相邻两帧图像之间变化的最简单、最直接的方法，它是直接比较了两帧图像对应像素点的灰度值的不同，然后通过阈值来提取序列图像中的运动区域。二值图像中为“0”的像素对应在前两帧图像之间没有发生（由于运动而产生的）变化的地方，为“1”的像素对应两帧图像间发生变化的地方，这常是由目标运动而产生的。

计算得到

$$Q_s = \varphi(h - z)^{3/2} F^{4/5} \beta \gamma \delta, \quad (1.1)$$

其中， $E_s$  与  $E_c$  分别表示基体和涂层的平均弹性模量； $\alpha_s$  与  $\alpha_c$  分别表示基体和涂层的热膨胀系数； $\Delta T$  表示喷涂前后温差； $D$  与  $d$  分别表示基体和涂层的厚度。

将所查各值代入公式可得：

$$\begin{aligned}
 C_0 &= \sqrt{2 \frac{k}{k-1} R T_0^* \left(1 - \frac{1}{\prod_T^m}\right)} \\
 &= \sqrt{2 \times \frac{1.4}{1.4-1} \times 287 \times 285 \times \left(1 - \frac{1}{3^{0.286}}\right)} \\
 &= 392.9 \text{ m/s.}
 \end{aligned} \tag{1.2}$$

处理过程如图1.1所示。

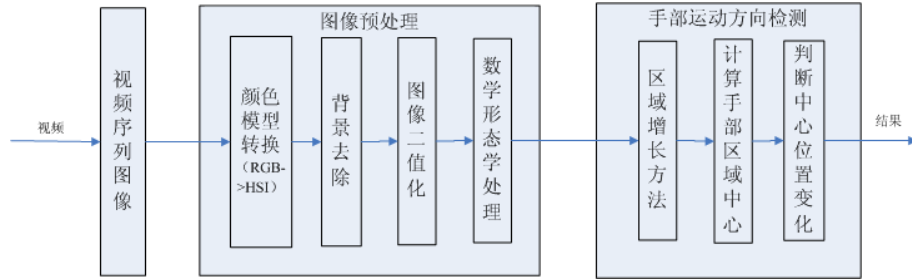


图 1.1 手部运动方向检测结构图

由图1.1可以知道，当得到一个变量的概率密度函数 pdf 时，熵就可以用来度量其状态的连贯性，同时，熵也是能量的一种表示。

## 1.2 本章小结

视频序列中运动目标的检测与跟踪是计算机视觉和图像编码研究领域的一个重要课题，在机器人导航、智能监视系统、交通检测、医学图像处理以及视频图像压缩和传输等领域都有广泛的应用。运动目标检测就是判断视频序列中是否存在运动目标，并确定运动目标的位置。

## 第2章 基础知识

基于视频序列的运动目标检测与跟踪涉及到很多研究领域，如数字图像处理、计算机视觉、信息融合、模式识别与人工智能等。

### 2.1 视频图像预处理

#### 2.1.1 常用颜色模型

颜色模型的用语是在某些标准下用通常可接受的方式简化彩色规范。本质上颜色模型是坐标系统和子空间的规范。位于系统中的每种颜色都由单个点来表示。

(1) RGB 彩色模型：

在 RGB 模型中，每种颜色出现在红、绿、蓝的原色光谱分量中，这个模型基于笛卡尔坐标系。

图2.1所示的立方体。图中 R、G、B 位于 3 个角上。在该模型中，灰度等级沿着主对角线从原点的黑色到点 (1, 1, 1) 的白色分布。

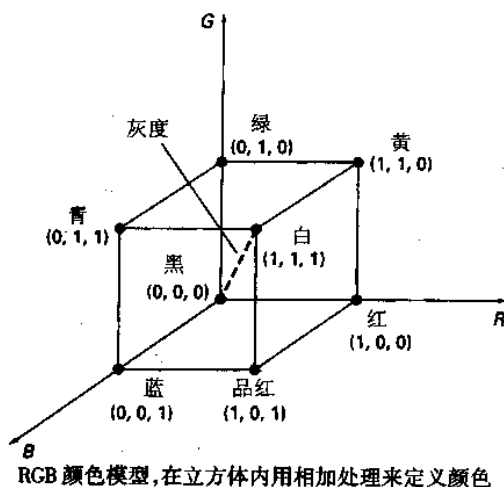


图 2.1 RGB 彩色立方体示意图

(2) 灰色模型：

本质上颜色模型是坐标系统和子空间的规范。位于系统中的每种颜色都由单个点来表示。单位在每列的书写示例如表2.1所示。

表 2.1 单位在每列的书写示例

| 基体         | 序号 | 粉末类型和<br>预热温度 (°C) | 失效温度 (°C) | $E_c$ 计算值 (GPa) |
|------------|----|--------------------|-----------|-----------------|
| SUS304 不锈钢 | 1  | 粗粉 & 1000          | 180       | 4.21            |
|            | 2  | 粗粉 & 800           | 10        | 4.38            |
|            | 3  | 细粉 & 1000          | 300       | 4.95            |
|            | 4  | 细粉 & 800           | 120       | 5.08            |

表格的分栏情况示例如表2.2所示。

表 2.2 分栏情况示例

| 基体         | 粉末类型 | 预热温度 (°C) | 平均值    |
|------------|------|-----------|--------|
| SUS304 不锈钢 | 粗粉   | 600       | 44.28% |
|            |      | 800       | 42.37% |
|            |      | 1000      | 39.74% |
|            | 细粉   | 600       | 27.95% |
|            |      | 800       | 25.41% |
|            |      | 1000      | 24.77% |
| 碳钢         | 粗粉   | 1000      | 35.65% |
|            | 细粉   | 1000      | 22.95% |

表的通栏情况和全表统一单位的情况如表2.3所示。

表 2.3 插入表格的通栏示例 (单位: 台)

| 地点<br>时间 | 电风扇 | 冰箱  | 洗衣机 |
|----------|-----|-----|-----|
| 10 月     | 100 | 200 | 300 |
| 11 月     | 200 |     |     |
| 12 月     | 200 | 100 | 400 |
| 合计       | 500 | 500 | 900 |

若表格一页内放不下, 可以使用跨页表格。跨页表格的情况如表2.4所示。

表 2.4 CMS\_VIDEO 数据表 (跨页表格)

| 字段标识       | 字段含义 | 数据类型         | 是否主键 | 是否外键 |
|------------|------|--------------|------|------|
| ID         | ID   | INTEGER      | 是    | 否    |
| VIDEO_NAME | 视频名称 | VARCHAR2(20) | 否    | 否    |

表 2.4 （续表）

| 字段标识        | 字段含义   | 数据类型         | 是否主键 | 是否外键 |
|-------------|--------|--------------|------|------|
| VIDEO_TYPE  | 视频类型   | VARCHAR2(20) | 否    | 是    |
| VIDEO_PATH  | 视频路径   | VARCHAR2(20) | 否    | 否    |
| UPLOADER_ID | 上传人 ID | INTERGER     | 否    | 是    |
| UPLOAD_DATE | 上传日期   | DATE         | 否    | 否    |
| ISPASS      | 是否审批   | INTERGER     | 否    | 否    |

## 2.2 本章小结

本章主要介绍了表格的显示。

## 第3章 视频图像预处理

### 3.1 引言

本章是视频图像的预处理阶段，首先，获取视频图像；然后对视频图像序列中的每帧图像进行图像预处理。如图3.1所示。

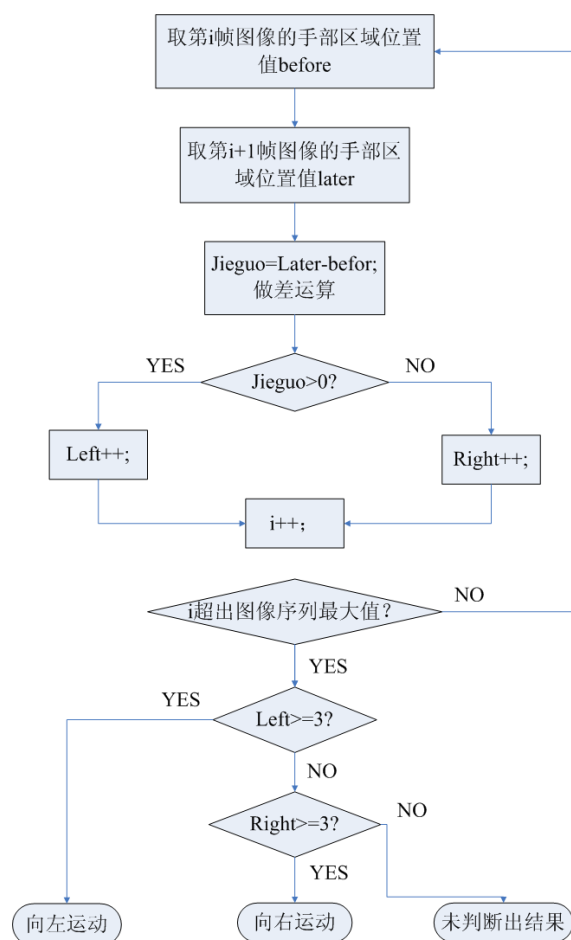


图 3.1 手势运动方向检测流程图

由图3.1可知，视频图像的预处理阶段，首先，获取视频图像；然后对视频图像序列中的每帧图像进行图像预处理。

### 3.2 图像的多种显示方式

分图的情况如图3.2所示。

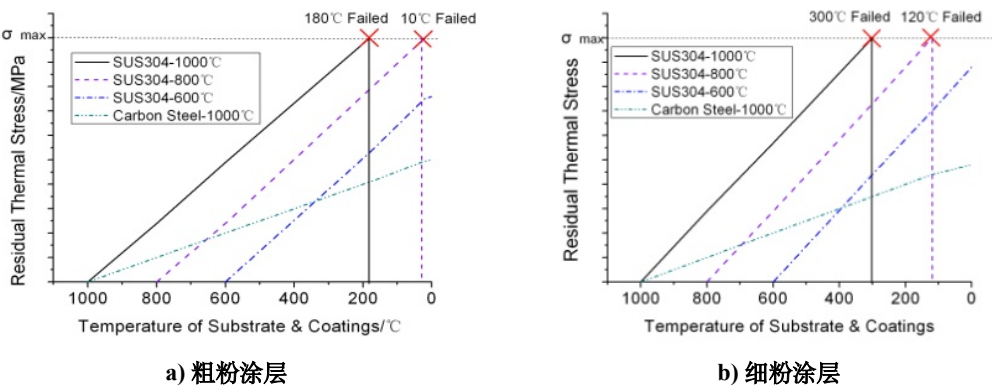


图 3.2 涂层在冷却过程中残余热应力的变化情况

在图中说明比较多的情况下，采取如图3.3的格式。

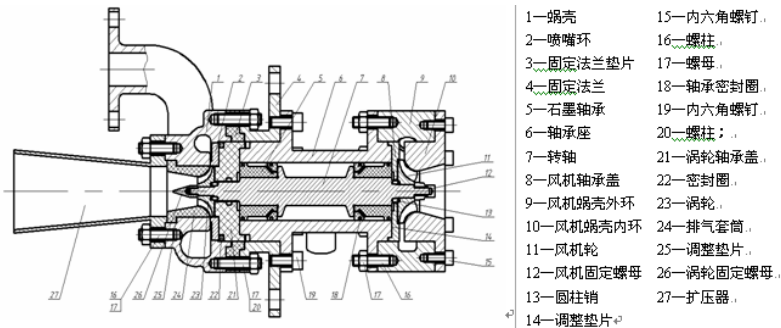


图 3.3 透平膨胀机的组成结构

### 3.3 本章小结

本章主要介绍了图片的格式。

## 第 4 章 系统性能分析

伪代码的使用示例如下所示。

---

### 算法 1: 不相交分解 (disjoint decomposition)

---

在此处添加不带编号的内容（若无，则将该行注释即可）。

**输入:** A bitmap  $Im$  of size  $w \times l$

**输出:** A partition of the bitmap

```

1 special treatment of the first line;
2 for  $i \leftarrow 2$  to  $l$  do
3   special treatment of the first element of line  $i$ ;
4   for  $j \leftarrow 2$  to  $w$  do
5      $left \leftarrow \text{FindCompress}(Im[i, j - 1]);$ 
6      $up \leftarrow \text{FindCompress}(Im[i - 1, ]);$ 
7      $this \leftarrow \text{FindCompress}(Im[i, j]);$ 
8     if  $left$  compatible with  $this$  then // 此处添加注释:  $O(left, this) == 1$ 
9       if  $left < this$  then  $\text{Union}(left, this);$ 
10      else  $\text{Union}(this, left);$ 
11    end
12    if  $up$  compatible with  $this$  then //  $O(up, this) == 1$ 
13      if  $up < this$  then  $\text{Union}(up, this);$  // 再次用 Union 函数
14      // this is put under up to keep tree as flat as
15      // possible
16      else  $\text{Union}(this, up);$ 
17      // this linked to up
18    end
19  end
20 foreach element  $e$  of the line  $i$  do  $\text{FindCompress}(p);$ 
21 end

```

---

引用时的格式为：算法1实现了…。

## 结论

在 Visual c++6.0 开发环境下，借助于 OpenCV 开放平台，设计并实现了基于低端摄像头视频手势运动检测系统。

## 参 考 文 献

- [1] 付梦印, 邓志红, 张继伟. Kalman 滤波理论及其在导航系统中的应用[M]. 北京: 科学出版社, 2003.
- [2] 邓宇. 复杂背景下的运动目标检测技术研究[D]. 贵州大学, 2007.
- [3] 张爱茜, 陈日清, 魏东斌, 等. 氯代芳香族化合物对羊角月牙藻的毒性及 QSAR 分析[J]. 中国环境科学, 2000, 20(02): 102-105.
- [4] STAUFFER C, GRIMSON W E L. Adaptive background mixture models for real-time tracking [C]. Proceedings. 1999 IEEE computer society conference on computer vision and pattern recognition (Cat. No PR00149): Vol. 2. IEEE, 1999: 246-252.

## 致 谢

衷心的感谢数学专业各位老师，在大学学习期间，给予了我极大地鼓励和帮助，在学习上给予了我严谨、耐心的指导，在生活上给与了我亲切、热情的关怀。老师们渊博的学识、谦逊、谨慎的治学作风，一丝不苟、尽职尽责的工作态度以及正直的为人之道，都将是我终身受益，并激励我始终刻苦努力。在此，我向各位老师表示崇高的敬意和衷心的感谢！

## 附录 1 企业信息表

表 A.1 企业信息表

| 字段名称                 | 中文描述              | 类型       | 长度 |
|----------------------|-------------------|----------|----|
| ID                   | ID                | NUMBER   | 15 |
| COMPANY_ID           | 公司 ID             | VARCHAR2 | 60 |
| LOGISTER_AGENT       | 委托代理人             | VARCHAR2 | 60 |
| SHORT_NAME           | 物流商简称             | VARCHAR2 | 60 |
| BUSINESS_FIELD       | 行业类别              | VARCHAR2 | 10 |
| WAY_VEHICLE          | 公路运输              | VARCHAR2 | 10 |
| WAY_TRAIN            | 铁路运输              | VARCHAR2 | 10 |
| WAY_SHIP             | 船舶运输              | VARCHAR2 | 10 |
| WAY_PIPELINE         | 管道运输              | VARCHAR2 | 10 |
| WAY_CONTAINER        | 集装箱运输             | VARCHAR2 | 10 |
| WAY_OTHERS           | 其他运输方式            | VARCHAR2 | 60 |
| FAX                  | 传真                | DATE     |    |
| SETUP_DATE           | 成立日期              | VARCHAR2 | 60 |
| BUSINESS_LICENSECODE | 营业执照号码            | DATE     | —  |
| BUSINESS_LICENSEDATE | 营业执照有效期           | VARCHAR2 | 60 |
| GAS_LICENSECODE      | 许可证号码             | DATE     | —  |
| GAS_LICENSEDATE      | 许可证有效期            | VARCHAR2 | 60 |
| HAZARD_LICENSECODE   | 化学危险品经营<br>许可证号码  | DATE     | —  |
| HAZARD_LICENSEDATE   | 化学危险品经营<br>许可证有效期 | VARCHAR2 | 60 |
| STATE_TAXACCOUNT     | 国税税号              | VARCHAR2 | 60 |
| CREATE_USERID        | 创建人               | NUMBER   | 15 |

## 附录 2 相关定理证明

## 附录 3 程序代码

```
int main(){
    int i;
    printf("hello latex!\n");
}
```

```
    return 0;
}
```

计算  $n$  的阶乘 (C++):

```
#include<iostream>

long long factorial(int n){
    if (n <= 1) {
        return 1;
    }
    return n * factorial(n - 1);
}

// 测试代码
int main(){
    int n;
    std::cout <<"请输入一个整数: ";
    std::cin >> n;
    if (n < 0) {
        std::cout <<"请输入一个非负整数。" << std::endl;
        return 1;
    }
    std::cout << n <<"的阶乘是: " << factorial(n) << std::endl;
    return 0;
}
```

计算  $n$  的阶乘 (Java):

```
public class FactorialCalculator {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("请输入一个非负整数: ");
        int n = scanner.nextInt();

        if (n < 0) {
            System.out.println("输入错误! 请输入一个非负整数。");
        } else {
            long factorial = calculateFactorial(n);
            System.out.println(n + "的阶乘是: " + factorial);
        }
    }
}
```

```

        scanner.close();
    }

    public static long calculateFactorial(int n) {
        long result = 1;
        for (int i = 1; i <= n; i++) {
            result *= i;
        }
        return result;
    }
}

```

计算  $n$  的阶乘 (Python):

```

def factorial(n):
    if n == 0 or n == 1:
        return 1
    else:
        return n * factorial(n-1)

# 测试代码
n = int(input("请输入一个非负整数: "))
if n < 0:
    print("输入错误! 请输入一个非负整数。")
else:
    result = factorial(n)
    print(f"{n}的阶乘是: {result}")

```

计算  $n$  的阶乘 (MATLAB):

```

function result = factorial(n)
    if n < 0
        error('输入错误! 请输入一个非负整数。');
    elseif n == 0 || n == 1
        result = 1;
    else
        result = 1;
        for i = 2:n
            result = result * i;
        end
    end
end

```

```

        end
    end

% 测试代码
n = input('请输入一个非负整数: ');
try
    result = factorial(n);
    fprintf('%d的阶乘是: %d\n', n, result);
catch ME
    if strcmp(ME.identifier, 'MATLAB:narginchk:
        notEnoughInputs')
        fprintf('没有输入任何数字。 \n');
    else
        fprintf('发生错误: %s\n', ME.message);
    end
end
end

```