

# 基于 Python 的三线摆实验综合系统设计\*

姜 响 薛 丽 任一鸣

(湖北科技学院电子与信息工程学院 湖北 咸宁 437100)

(收稿日期:2022-11-07)

**摘 要:**基于 Python 及 OpenCV、Matplotlib 和 Pygame 等库开发了一款在三线摆实验中集周期测量、图像绘制、数据处理及实验仿真等功能于一体的软件. 实验时用摄像头模块记录下盘标志物运动, 在软件内用 OpenCV 识别并解析数据、Matplotlib 绘制实时运动图像、两套独立算法计算周期. 此外自制数据记录板块可辅助学生数据处理, 以 Pygame 搭建的二维实验仿真系统可辅助学生学习实验操作.

**关键词:**Python; OpenCV; 三线摆; 周期测量; 实验仿真

在大学物理实验中, 测量刚体的转动惯量是其中一个重要的实验项目, 而三线摆法又是测量转动惯量的常用方法之一, 在用三线摆法测量转动惯量的实验中, 重要且困难的环节是测量圆盘摆动周期. 传统测量方法有秒表计时法和光电门法. 秒表计时通过记录 50 次摆动用时得出单次周期, 但易受观测者影响且易出现误记现象; 光电门精度提高, 但仍易误记. 以上两法均无法获得三线摆运动图像等信息. 针对传统方法的改进已见诸报端<sup>[1-6]</sup>, 但安装传感器、磁化三线摆等做法会对其运动带来影响; 某些传感器受限工作原理, 所得图像仅呈周期变化, 无法揭示角度或位移与时间的关系; 利用 Phyphox 或 Tracker 等软件则需熟悉其使用技巧以及不同情况下生成数据的特征, 不利推广.

鉴于以上原因, 基于 Python 开发了一款软件. 实验时在三线摆下盘贴上红色标志物, 摄像头模块接入电脑并拍摄标志物运动, 在软件内利用 OpenCV 将视频转化为一系列运动数据, 利用 FFT(快速傅里叶变换) 和均值算法计算周期并以 Matplotlib 实时绘制图像. 此外基于 Tkinter 架构了软件数据表格, 可辅助及引导学生完成数据处理和误差分析; 基于 Pygame 设计了实验仿真系统, 可辅助学生学习实验操作.

## 1 设计方案

### 1.1 图像处理与数据可视化

图像处理捕捉标志物的实现主要通过 OpenCV 的图像二值化及轮廓识别等功能; 计算周期则通过对均值点集作逐差及分析 FFT 频谱两种方法; 数据可视化则以 Matplotlib 交互模式将已得数据实时绘图展示. 此外因 Matplotlib 动态响应过长, 故以多进程运行计算周期与数据可视. 实现步骤如下:

(1) 导入视频并识别标志物. OpenCV 的 VideoCapture 函数导入摄像头, 对视频以 read 方法逐帧获取图像(图 1). 以 resize 函数切割图片, 只保留中间  $\frac{1}{3}$  区域.



图 1 原始图像

以 cvtColor 函数将 rgb 通道改为 hsv, 并以 inRange 函数进行图像二值化, 其参数为颜色空间. 以红色为例, 其 hsv 颜色空间为: 色调 156 ~ 180 或 0 ~ 10, 饱和度 43 ~ 255, 明度 46 ~ 255, 处理后图

\* 湖北科技学院教学研究项目, 项目编号: 2020-XA-014, 2020-XB-002.

作者简介: 姜响(2002-), 男, 2020 级本科生.

通讯作者: 薛丽(1985-), 女, 博士, 副教授, 研究方向为物理学.

像如图2所示,图像二值化后还需适当腐蚀噪点。



图2 二值化处理后的图像

以 `findContours` 函数识别图像全部轮廓并将其中心面积最大者视为标志物。求解该轮廓点集期望即得中心点坐标。记录计算机实时时间与中心点坐标组合保存。在图像中重新绘制轮廓、标出中心点并写入时间,并以 `imshow` 函数显示(图3左上角)。

(2) 计算周期。创建子进程并通过切片刷新共享内存向其传递数据。子进程中将位移均值归零后以两种算法计算周期。

均值算法:设置“宽度”变量。遍历数据,判断位移是否低于“宽度”,若是则以列表保存其对应时间。对列表中相近部分取平均获得近似均值点集,作逐

差得周期。

FFT:调用 `numpy.fft.fft` 函数,参数为位移数据,采样周期为两点时间差,返回值为频谱数据。取数据实部并遍历寻出其最大或最小值点,其频率即为振动频率。

(3) 可视化数据。在子进程中以 `Matplotlib.pyplot` 类的 `ion` 函数启动交互, `subplot` 函数分配图像区间, `plot` 函数绘制  $x-t$ 、 $v-t$ 、 $x-v$  和 FFT 频谱图像, `text` 函数显示周期和最小分度值(图3)。

(4) 结果呈现。实测标志物面积并通过一个输入界面传入软件。软件将实测面积与图像中面积取比值并开方即得毫米单位与像素点单位的比率,进一步可得一系列拥有实际单位的数据。不启动交互绘制最终实验图像,并以 `csv` 格式导出全部数据。

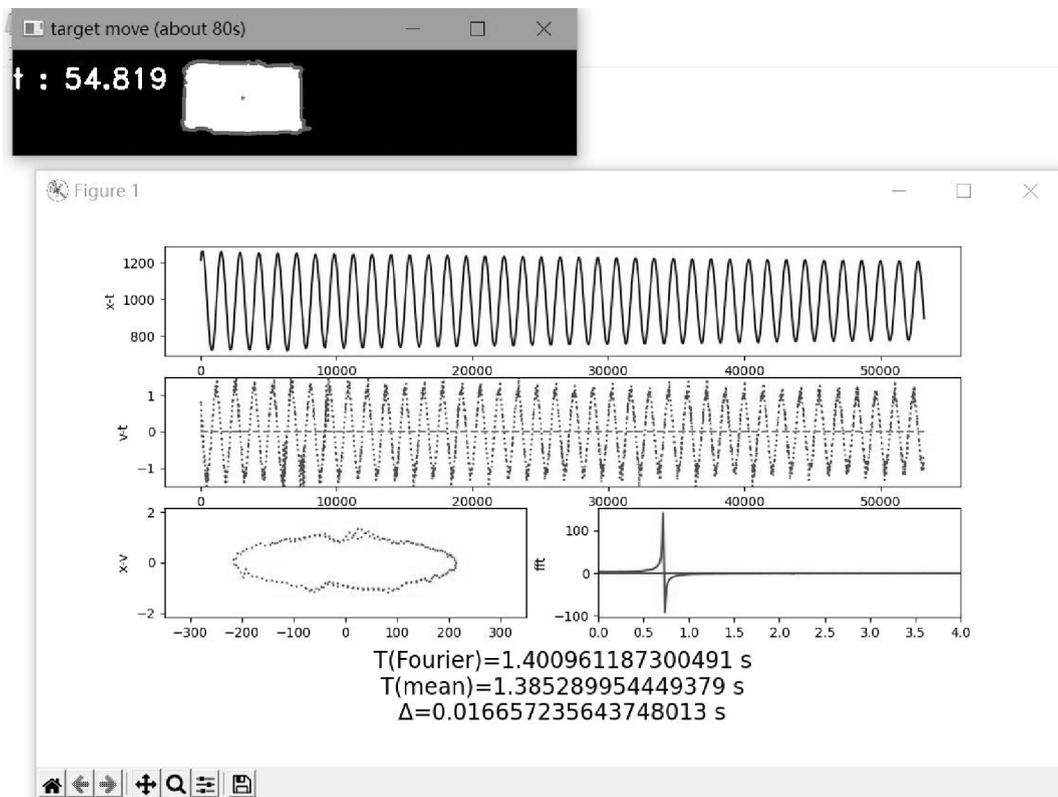


图3 处理过程实拍

## 1.2 操作及仿真界面

数据表格界面需实现输入、提示、数据拟合等功能,以稍底层的 Tkinter 架构,其它操作界面以 `easygui` 架构。实验仿真界面则以游戏库 `Pygame` 开

发。具体实现操作及仿真界面如下:

### 1.2.1 数据表格等操作界面

对于数据表格,创建两个子进程并实例化窗口,在嵌套循环中生成并渲染 `Text` 控件并以 `StringVar`

变量跟踪其输入,组成输入部分;以 label 控件显示表格旁文字;以 botton 控件生成按钮并设置 command 参数的关联函数,“refresh”按钮关联函

数中将输入数据保存在 pkl 文件中以实现数据保存,“tips”按键关联函数中创建顶层窗口并显示图片以实现提示功能.表格和提示界面如图 4 所示.

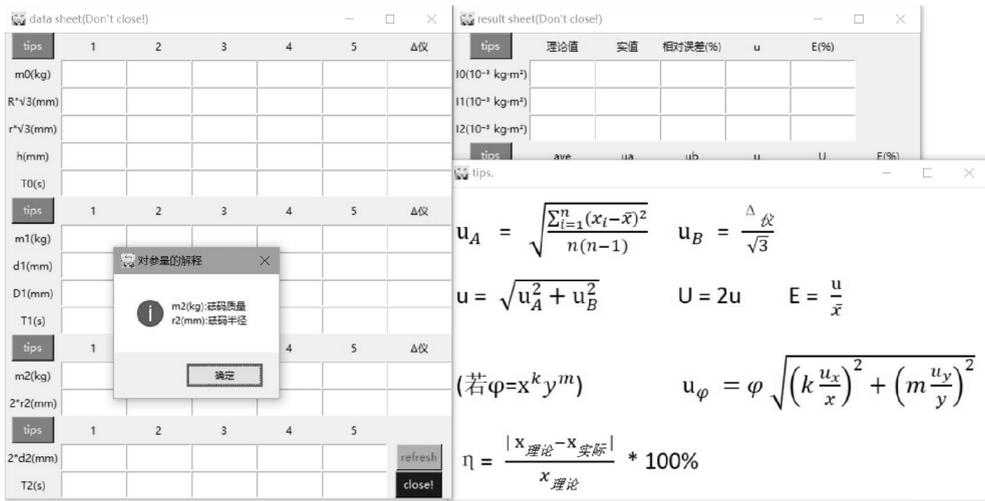


图 4 数据表格和部分提示

实验时,在 data 表格中填入测量数据,点击“tips”获取提示并在 result 表格中填写计算结果.保存数据后,系统以 in 语句检测数据中是否存在空字符串,若无则将字符串转为浮点数并验算数据处理及误差分析结果,并以信息窗口逐个呈现.此外考虑到“验证平行轴定理”环节有表达式<sup>[7]</sup>

并架构.以箭头按钮和“游标卡尺”为例.对于箭头按钮,创建 Arbutton 类对象,初始参数包括窗口及箭头指向,箭头本体以 lines 方法绘制;创建 set 方法在窗口中渲染箭头,参数为箭尖坐标;创建 isdown 和 isup 方法,通过检测鼠标事件判断箭头按下或弹起.对于“游标卡尺”(图 6),用绘图软件绘制主尺和分尺,仿真时导入图片并将两尺上下边缘贴合渲染,主尺按既定“待测物”参数向左移动即可模拟读数.

$$d^2 = -\frac{I_0 - 2I_1}{2m_1} + \frac{(m_0 + 2m_1)gRr}{8\pi^2 d_0 m_1} T^2 \quad (1)$$

实验需以线性拟合验证  $T^2 - d^2$  关系及截距,实现方案:调用 numpy.polyfit 函数,参数为横纵坐标及拟合方程最高次幂,函数返回拟合方程及判据系数.以 plot 函数绘制拟合曲线及散点.可通过判据系数及截距与理论值相对误差验证平行轴定理.

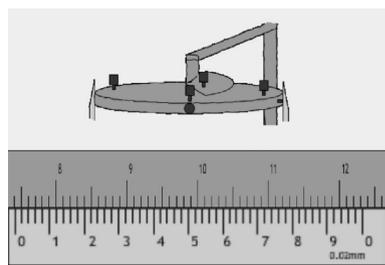


图 6 游标卡尺仿真实验

其它操作界面都以调用 easygui 库中函数来实现.以系统主界面为例,调用 choicebox 函数,其 choices 参数为各功能名称的字符串,即得图 5 般界面.

界面架构以“气泡水准仪”和“摄像头调整”为例.“气泡水准仪”中,绘制并渲染 3 个大小颜色不同的圆充当“底座”“中心环”和“气泡”,组成虚拟气泡水准仪,其仿真实验界面如图 7 所示.实验时操控一系列按钮,“气泡”根据按钮事件进行移动;“摄像头调整”(图 7)则以 rectangle 函数绘制长方形来模拟标志物并捕捉键盘 w、a、s、d 响应来模拟调整摄像头操作.



图 5 主界面

1.2.2 仿真实验界面

考虑 Pygame 底层性强,一些控件需提前绘制

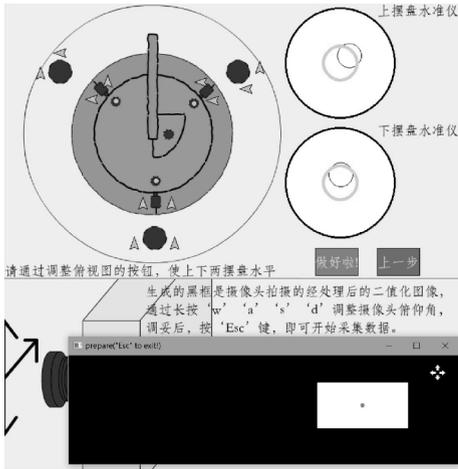


图7 气泡水准仪与摄像头调整实验界面

## 2 实验

电子天平称量实验中下盘质量  $m_0$  以及圆环的质量  $m$ ; 游标卡尺测量上盘半径  $r$ 、下盘半径  $R$  及圆环内径  $d$  和外径  $D$ . 记录完毕后用气泡水准器调整三线摆上、下圆盘水平, 最后卷尺测量 3 根摆绳绳长  $h$ . 摄像头与电脑连接, 在主界面“计算摆动周期”中点击“摄像头实时测量数据”, 进入准备界面.

红色标志物贴于三线摆下圆盘侧面, 摄像头正对标志物并且拉开一定距离, 如图 8 所示.



图8 摄像头对准红色标志物

摆动三线摆, 注意摆角尽可能小并尽量避免横摆. 调整摄像头使之清晰呈现出标志物运动, 按“Esc”键软件开始记录, 生成动态图像同时计算摆动周期等数据. 待 80 s 后自动停止记录, 保存数据并重复多次实验. 之后放上待测物, 继续重复多次实验.

## 3 实验结果及分析

为保证结果可对比, 实验时以软件和秒表同时测量摆动周期. 表 1、表 2 为测量数据.

表 1 其他物理量的测量结果

	下盘质量 $m_0/\text{kg}$	待测圆环质量 $m/\text{kg}$	上盘半径 $r/\text{mm}$	下盘半径 $R/\text{mm}$	摆绳绳长 $h/\text{mm}$	圆环内径 $d/\text{mm}$	圆环外径 $D/\text{mm}$
1	0.465	0.208	37.18	71.43	476.0	111.78	119.40
2	\	\	37.51	70.86	475.0	111.80	119.36
3	\	\	37.70	70.86	476.0	111.90	119.38
$\Delta_{\text{仪}}$	0.001	0.001	0.02	0.02	0.5	0.02	0.02

表 2 3 种不同方法测得的周期、求得的转动惯量及其相对误差

方法	无待测物摆动周期 $T_0/\text{s}$			圆盘转动 惯量 $I_0/$ ( $10^{-3} \text{kg} \cdot \text{m}^2$ )	$I_0$ 相对误差 $\eta_{I_0}/\%$	有待测物摆动周期 $T/\text{s}$			总转动 惯量 $I/$ ( $10^{-3} \text{kg} \cdot \text{m}^2$ )	$I$ 相对 误差 $\eta_I/\%$
人工	1.390 7	1.390 6	1.393 8	1.253 4	6.78	1.428 8	1.426 0	1.436 6	1.916 6	2.52
均值	1.391 1	1.386 9	1.386 1	1.246 8	6.23	1.438 7	1.437 3	1.438 0	1.936 9	3.60
FFT	1.378 8	1.379 6	1.379 2	1.231 3	4.91	1.429 3	1.428 9	1.428 2	1.912 1	2.28

由上表, FFT 所得转动惯量数据的相对误差较小. 比较发现以 FFT 算法获得的周期数据与其他组相比偏小, 且空载圆盘转动惯量的相对误差明显较大.

三线摆圆盘转动的同时还不可避免存在横向摆

动<sup>[1]</sup>. 某次实验中 [ $x-t$  图像见图 9(a)] 两种频率相近的振动相互叠加, 出现了明显的“拍”现象. 而在由傅里叶变换所生成的频谱图像中, 两种不同频率的波峰(或波谷)很容易区分. 如图 9(b) 所示, 出现了 1 和 2 两个频率不同的波谷.

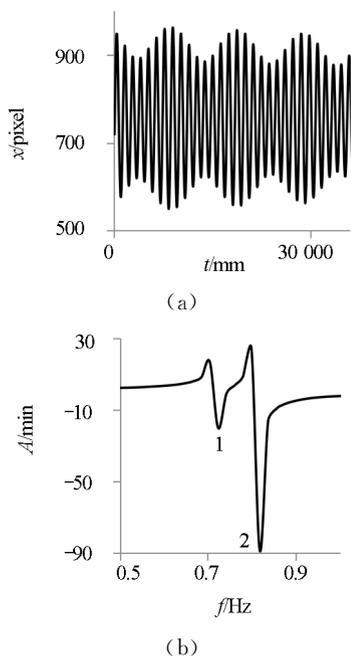


图9 某次实验所得系列图像

一定条件下,三线摆圆盘转动会与横摆产生耦合<sup>[8]</sup>,影响周期测量,可解释空载圆盘转动惯量相对误差为何相比较大。

#### 4 结束语

通过 OpenCV 采集三线摆运动数据,其改进对三线摆运动几乎无影响,且测量结果相比而言误差较小,更具使用价值;通过 Matplotlib 实现了三线摆

运动的图像化,观察到传统实验难以观察到的现象,有助于培养学生从图像和表格中发现问题的能力,以自制数据表格进行数据处理,方便整理且实现了实验操作全程无纸化.基于 Pygame 搭建的二维实验仿真系统有助学生学习实验操作。

#### 参考文献

- [1] 刘昶. 三线扭摆系统测量刚体转动惯量试验方法改进[J]. 汽车科技, 2011(3): 67-71.
- [2] 高寒, 张黔, 谢媛, 等. Tracker 软件在三线摆测量刚体转动惯量实验中的应用[J]. 科教导刊(中旬刊), 2017(2): 61-63.
- [3] 殷子棋, 丁益民, 史振宇, 等. 利用智能手机磁场传感器测量刚体的转动惯量[J]. 大学物理实验, 2018, 31(3): 72-74.
- [4] 吴星星, 丁益民, 秦凤玲. 利用智能手机旋转传感器测量转动惯量[J]. 大学物理实验, 2020, 33(2): 79-81.
- [5] 石明吉, 李波波, 杨雪冰, 等. 基于霍尔效应的三线摆周期测量装置的研制[J]. 物理与工程, 2018, 28(6): 95-99.
- [6] 石明吉, 刘斌, 李波波, 等. 基于电磁感应的三线摆周期测量装置的研制[J]. 物理与工程, 2019, 29(4): 88-92.
- [7] 杨述武. 普通物理实验(1) 力学、热学部分[M]. 北京: 高等教育出版社, 2015: 71-74.
- [8] 刘丹, 侯之超. 三线摆方程简化及其共振问题研究[J]. 振动与冲击, 2007(8): 137-139.

## Design on the Three-Line Pendulum Experimental Integrated System Based on Python

JIANG Xiang XUE Li REN Yiming

(School of Electrical and Information Engineering, Hubei University of Science and Technology,  
Xiangyang, Hubei 437100)

**Abstract:** Based on Python and OpenCV, Matplotlib and Pygame and other libraries, a software integrates period measurement, graph drawing, data processing and experimental simulation in three-line pendulum experiments has been developed. During the experiment, the camera module was used to record the movement of marker of three-line pendulum, and OpenCV was used to identify and parse the data, Matplotlib to draw real-time motion graphs, and two independent algorithms to calculate the period. In addition, the self-made data recording section can assist students in data processing, and the two-dimensional experimental simulation system built by Pygame can assist students in learning experimental operations.

**Key words:** Python; OpenCV; the three-line pendulum; period measurement; experimental simulation