2023-2024学年度第一学期

华南师范大学

计算机学院

《初级软件设计实作》报告

|  |  |
| --- | --- |
| 题目： | 桌面透视工具 |
| 班级： | 计算机科学与技术 1 班 |
| 学号： | 20203231005 |
| 姓名： | 林镝镝 |
| 指导老师： | 王涛 |
| 仓库地址： | https://github.com/LdoDoeg222/MagnifyingGlass |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **总分** | **指导老师评分** | **备注** |
| **一、报告文档** | **30** |  |  |
| 1内容基本要点 | 20 |  |  |
| 2格式 | 10 |  |  |
| **二、系统验收** | **50** |  |  |
| 1功能 | 30 |  |  |
| 2程序结构、编码规范 | 10 |  |  |
| 3界面 | 10 |  |  |
| **三、作品展示** | **20** |  |  |
| 1功能亮点 | 10 |  |  |
| 2专业表达 | 10 |  |  |
| **总评分** | |  |  |

目录

[1 要求 2](#_Toc155910656)

[1.1 功能需求 2](#_Toc155910657)

[1.2 相关技术 2](#_Toc155910658)

[2 分析 2](#_Toc155910659)

[2.1 系统设计 2](#_Toc155910660)

[2.1.1 技术选型 2](#_Toc155910661)

[2.1.2 结构设计 2](#_Toc155910662)

[2.2 代码实现 3](#_Toc155910663)

[2.2.1 Viewer的设计 3](#_Toc155910664)

[2.2.2 针对Scanner的设计 4](#_Toc155910665)

[3 运行情况 6](#_Toc155910666)

[4 需求变更 6](#_Toc155910667)

[5 总结 7](#_Toc155910668)

[参考文献 7](#_Toc155910669)

# 要求

## 功能需求

实现一个电脑桌面的放大镜。具体功能如下：

1. 在软件启动时，显示一个“取景框”（小）和“展示框”（大），展示框的内容与取景框中的内容是一致的，但展示框内容是放大。 取景框是一个虚线框。
2. 用户可以用鼠标来移动取景框。
3. 当取景框在电脑桌面上移动时，软件马上把该放大镜取景框所在位置的内容给放大显示到展示框中。

## 相关技术

1. 图像展现
2. 截屏功能的实现
3. 鼠标消息处理

# 分析

## 系统设计

### 技术选型

结合个人的技术能力，可以选择的技术语言主要为：C++、Python、Java，不过基于代码便捷的考虑，我选用了Python作为本项目的技术语言。

经过一些资料的查阅，发现Python的tkinter绘图工具非常适合本次项目，因此，项目的技术选型为Python + tkinter。

### 结构设计

可以使用面向对象设计法，对所需工具进行设计——分为两个类，Viewer展示框类，Scanner取景框类。

相关技术部分也相当于提示，告诉我们要往三个方面入手。

首先，若要满足取景框Scanner的要求，必须能够做到：

1. 无菜单栏标题栏透明窗口
2. 绘制虚线（图像展现）
3. 可任意点击取景框的任意部位，进行拖动（鼠标消息）

而展示框Viewer则需要满足：

1. 实时显示取景框所在位置“下面”的内容（截屏功能+图像展现）
2. 对取景框内容进行放大处理（图像展现）

因此整体流程如下：

1. Main语句实例化Viewer，Viewer实例化Scanner
2. Viewer获取Scanner所在位置下的图像
3. Viewer将图像放大
4. Viewer将放大后图像绘制在窗体内

## 代码实现

Viewer与Scanner的长宽需要呈现等比例关系，因此它们需要用相同的基础长宽baseW、baseH，具体的长宽则可以通过设置倍率变量rate来控制。



同时，main语句实例化Viewer对象，并调用mainloop使其进入主循环，而Viewer则在其构造函数实例化Scanner

文本

低可信度描述已自动生成

文本, 信件

描述已自动生成

### Viewer的设计

Viewer在实例化Scanner后，需要实现的功能包括：

1. 获取Scanner所在位置（通过tkinter对象的成员方法winfo\_x、winfo\_y方法）
2. 获取Scanner下方的图像（通过ImageGrab.grab方法来实现）
3. 对从Scanner所在位置中获取的图像，进行等比例拉伸放大（通过 ImageGrab 对象的resize方法实现）
4. 将放大后的图像实时绘制在窗口中（通过canvas对象的create\_image方法实现）
5. 为实现实时绘制，上述三个步骤需以循环的方式被不断重复调用（通过tkinter的after方法实现）

文本

描述已自动生成

### 针对Scanner的设计

Scanner被实例化后，只需要实现鼠标拖动功能、半透明功能、虚线框、无菜单功能即可。

半透明是基础属性，只需通过调用attributes方法调整数值即可；

虚线框需要绘制，可通过canvas对象的create\_line方法并且指定dash参数，并且指定顶点即可；

无菜单则只需调用overrideredirect方法。

图片包含 日历

描述已自动生成

针对鼠标事件的逻辑，首先必须先理清楚鼠标事件会被如何处理——鼠标从点击窗体到释放，会有什么参数受到影响。

经过整理后，了解到如下参数：

1. 在 start\_dragging中的event.x和event.y：按压起点相对窗口原点的坐标
2. 在do\_dragging中的event.x和event.y：触发事件时鼠标相对窗口原点的坐标（也即下一次do\_dragging事件触发前，鼠标相对窗口原点的坐标）
3. self.winfo\_x() 和self.winfo\_y()：Scanner相对于整个屏幕的坐标

而tkinter对象中，geometry方法可以通过设置绝对坐标，来指定窗口位置，因此，首先需要将winfo坐标参数作为基底，并将do\_dragging的event坐标参数和start\_dragging的event坐标参数作差，得到delta\_x和delta\_y后与winfo坐标参数求和，再通过geometry方法设置位置，从而实现鼠标点击窗体任意位置实现拖动的效果。

文本

描述已自动生成

# 运行情况

文本

描述已自动生成

# 需求变更

在原型审查后，系统被提出了新的需求：

需要让取景框位于左半屏时以原比例显示，而在右半屏时采用放大显示。

针对这一问题，只需在grabber.resize方法上进行修改即可，如图

文本

中度可信度描述已自动生成

运行结果如图：

图形用户界面, 应用程序

描述已自动生成

图形用户界面, 应用程序

描述已自动生成

# 总结

通过本次实训，我成功地实现了一个简单的桌面透视工具。但仍有可以改进的地方——使得半透明效果彻底优化，又或者提供更改窗口大小支持，在取景框窗体内提供支持自定义倍率的接口等。总之，经过本次实训，我基本掌握了设计初级软件的能力，也希望这份能力未来能在我的计算机之旅中提供助力。

# 参考文献

无