云台摄像机控制系统的方案和可行性分析

1 需要解决的问题

教会的云台摄像机的动作控制现在由一个测试软件实现，其操作非常不直观，且控制困难。直观地控制摄像机动作的市售方案不是没有，但是昂贵以及提供了众多教会并不需要的功能。于是希望自己建立一套云台摄像机的控制系统，达到直观，容易控制等目的。如果可能，还希易于维护，并且可以移植到其他用途。

1.1 需要实现的功能

教会作为非专业摄影用户，常常用到的摄像机功能很有限。主要就是摄像头动作（电机和焦距）的控制，其他一切参数全自动。具体功能包括：

* 平滑地控制方位和仰角电机的转速
* 平滑地控制焦距
* 在使用者不操作的时候并不运动
* 预存储位置，并可以快速切换
* 平滑（较缓慢）切换存储的位置，尽力防止抖动

1.2 涉及的技术

教会使用的云台摄像机的制造商是SONY，其控制protocol是公开的，称为VISCA protocol。

硬件基础是串口通信。摄像机上是圆串口，计算机上是扁串口。电压应该是15的那种。

2 方案，可行性和优缺点

2.1 独立的控制设备

* 使用arduino单片机，作为控制设备的处理器。
* 使用arduino的游戏手柄mount shield，作为输入设备。
  + 其中joystick用作控制摄像头的pan-tilt，按键控制zoom，按键存储和读取预存位置
* 使用Nokia 5110 LCD作为反馈信息和控制器状态的显示器
* 使用RS232 module作为和摄像机的接口
* 可能需要使用3D打印的外壳保持整洁

优点：系统独立，不依赖于电脑

缺点：硬件系统比较零散，软件难以调整修改和升级，控制的难度较大，手感难以保证

总结：除了独立性，没有什么优点可言

2.2 使用Xbox游戏手柄作为控制器

游戏手柄在控制用在控制摄像头是非常理想的，因为中间有两个模拟输入的joystick可以输入连续的控制量以达到平滑的运动，其上还有很多按键，方便存取预设和模式切换（如果后期需要扩展功能的话）。

2.2.1游戏手柄的操作方式：

* 左joystick，模拟控制pan-tilt
* 右joystick，模拟控制zoom
* 十字键，微调pan-tilt
* 模拟肩键，微调zoom
* ABXY，读取存储的位置
* 按住开关肩键按ABXY，存储位置

2.2.2 实现方式

2.2.2.1 现有的实验控制端+AHK脚本

现有的实验控制端有一个用命令直接控制摄像头的功能，所以可以使用AHK脚本，将手柄的动作映射到控制命令的输入。涉及到的脚本功能有：

* 控制计时器
* 读取手柄数据
* 计算hex格式的命令
* 输入文本信息
* 暂停和开始脚本的运行

优点：开发容易，因为不涉及串口通信硬件以及程序UI的设计

缺点：依赖现有软件，打开和关闭脚本过程不直观，存取现有位置的功能不容易实现

2.2.2.2 开发一个独立app

这个app需要涉及的技术包括但不止于：

* 开发ui
* 计算命令以字节的形式
* 串口通讯
* 读取手柄数据

可能的平台，应该都在Visual Studio里：

* VB.net
* C#.net

优点：操作简单，系统独立，容易推广，位置信息可以在UI上标注而不需要背键盘。可以在项目的后期与其他方案合并于同一个app。

缺点：虽然可以平滑的操作tilt-pan-zoom，但是仍然在直观程度上有改进的空间。

难度：对开发APP的过程并不熟悉

需要在机器上进行一定的调试，所以并不是非常的方便。

2.3使用鼠标操作

使用鼠标拖拽会使得镜头移动非常不平滑，所以在手柄方案存在的前提下，拖拽的方案没有存在的意义。然而鼠标作为最常见的输入设备没有之一，是一个十分值得考虑的。在拖拽方案之外，还有另外一个鼠标方案：选择拍摄区域。

系统在初始化时使用摄像头将所在区域拍摄一个全景相片。用户用鼠标在相片中框出需要拍摄的区域，由系统计算出对应的tilt-pan-zoom，然后由用户选择摄像头的移动速度，快速或缓慢地移动到对应区域。

优点：操作直观，

难度：全景图形的生成和位置的准确对应