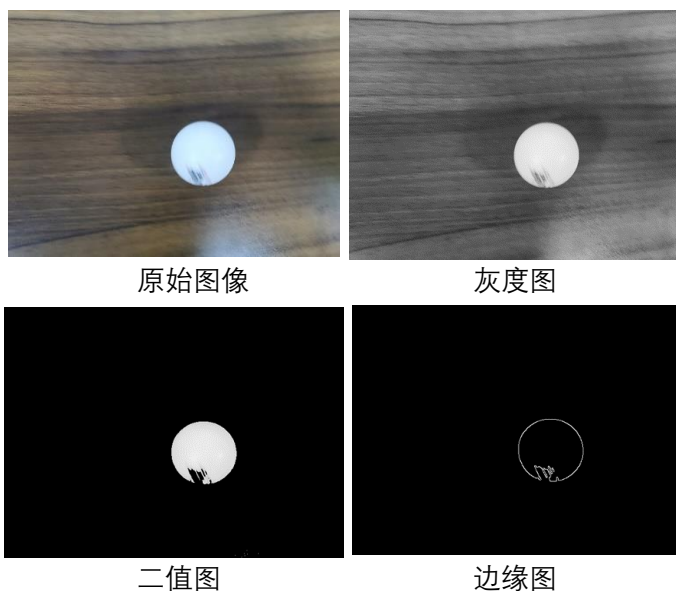


## 1.识别算法

### 1.1 图像处理

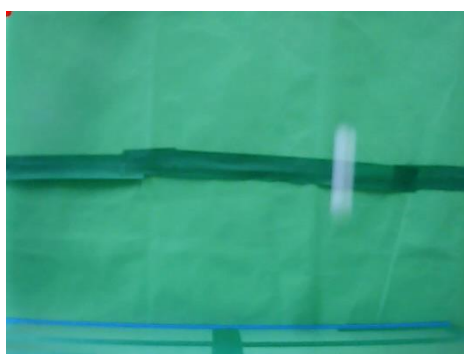
要对一幅照片中小球的球心位置进行定位需对照片进行处理, 根据目标颜色获得边缘图, 即具有目标颜色区域的边缘会被框出。在本项目中小球为白色, 所以图片中白色区域的边缘会被识别, 获得小球的轮廓后, 根据轮廓求出球心的位置坐标。图像坐标系中原点位于相片左上角, X 正方向指向右, Y 正方向指向下, 坐标单位为像素值。即每一个像素点所在的列数为 X 坐标值, 所在的行数为 Y 坐标值。

本程序主要基于 OpenCV 计算机视觉库, OpenCV 提供大量 API 接口函数, 可以非常方便地实现图像处理。首先将 RGB 图像转换为灰度图, 此时每一个像素的像素值在 0 到 255 之间。然后添加一个掩膜, 将像素值大于 200 的像素设置为 0, 小于 200 的像素设置为 255, 再与原图进行与运算。此时获得的图像只有目标颜色区域会显示为白色, 其余区域显示为黑色, 得到二值图。使用高斯模糊滤去噪声, 进行边缘检测获得图像中小球的边缘。



### 1.2 计算球心

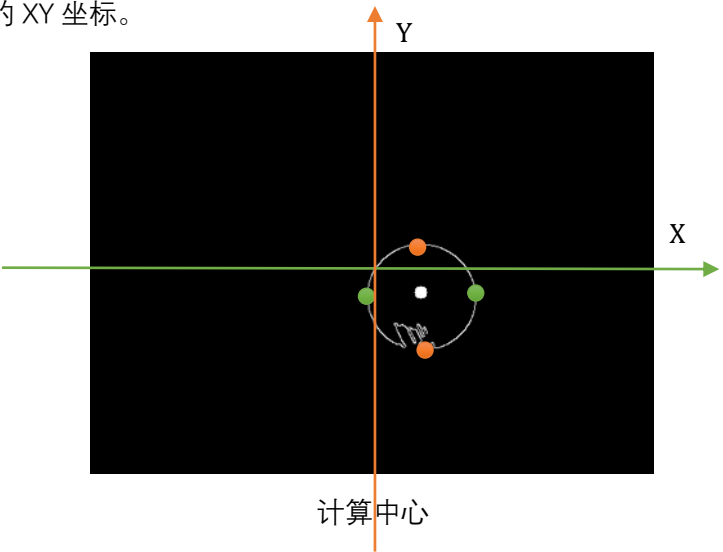
乒乓球在实际运动过程中运动速度非常快, 由于装置中所使用的为普通相机, 每秒帧数为 30fps, 所能拍到的运动中的乒乓球是一团不规则白影。



相机拍摄图像

所以需要求出形状不规则物体的中心, 并且为了使程序运行的时间尽可能短, 达到实

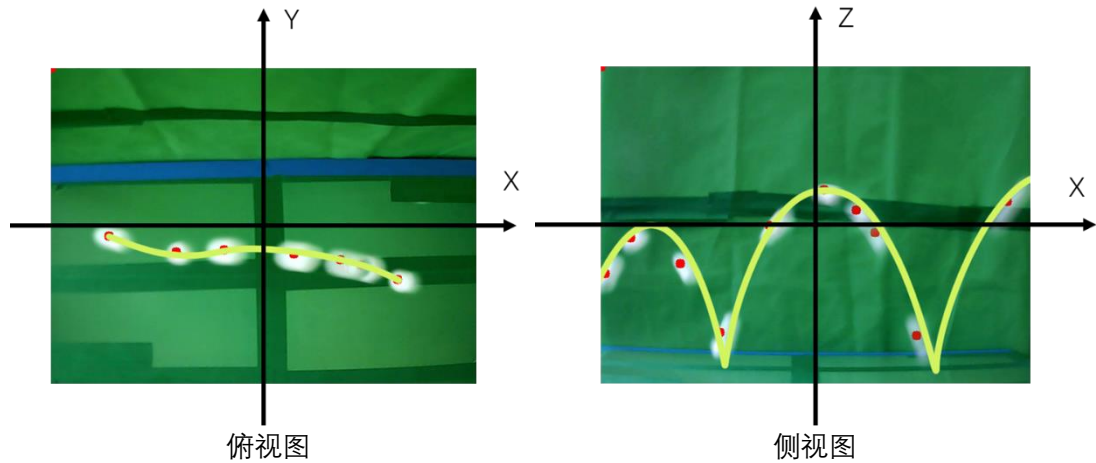
时检测球心的目的。采用的方法是从边缘图中分别获得 XY 坐标为最大和最小的点，求均值作为中心点的 XY 坐标。



## 2. 轨迹预测

### 2.1 观测轨迹

在获得乒乓球的位置后需要提前预判乒乓球的落点位置，才能使机器人及时接住乒乓球。我们假设乒乓球以一 XY 平面内的速度抛出，忽略空气阻力，乒乓球运动过程中不发生旋转。经过实际观测，发现俯视图中乒乓球的轨迹是一条曲线，不符合我们的设想：近似直线。侧视图中，乒乓球的轨迹近似于抛物线。

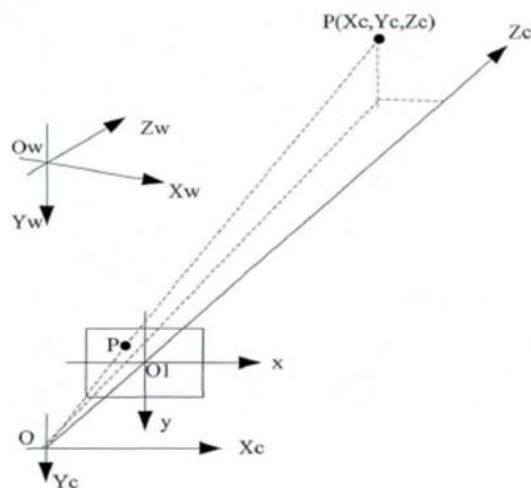


根据相机的成像原理，空间中某一点在空间坐标系中的坐标值与在图像坐标系中的坐标值成正比，比例系数与深度有关。乒乓球运动过程中与俯拍的摄像头之间的距离不断发生变化，比例系数变化，图像中检测到的点并不能真实反应乒乓球在水平投影面中的位置。

$$x = f \frac{X_c}{Z_c}$$

$$y = f \frac{Y_c}{Z_c}$$

(x,y)图像坐标系坐标， $(X_c, Y_c, Z_c)$ 空间坐标坐标， $f$ 相机焦距

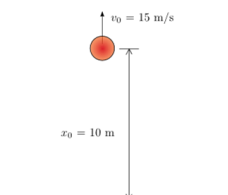


相机成像原理

## 2.2 预测模型

由于俯视图中获得乒乓球位置信息并不准确，并且为了简化预测过程，我们仅通过侧视图预测接球点。每次从乒乓球台边缘的中点抛出乒乓球，假设无 Y 方向的初始速度，即可以确定接球点的 Y 坐标。另已知接球点的 X 坐标，即乒乓球台另一侧边缘，需要预测到达目标 X 坐标时的 Z 坐标及时间。

侧视图观察到乒乓球的轨迹类似于抛物线。预测模型基于弹球模型，弹球模型是混合动态系统的经典示例。混合动态系统既有连续动态特性又有离散转换特性，其动态特性会发生变化，而且状态会发生跃变。在一定高度斜抛出一个小球，每当小球与底面碰撞时速度会发生突变，反向并衰减。碰撞后速度与碰撞前速度成正比，比例为恢复系数  $k$ 。

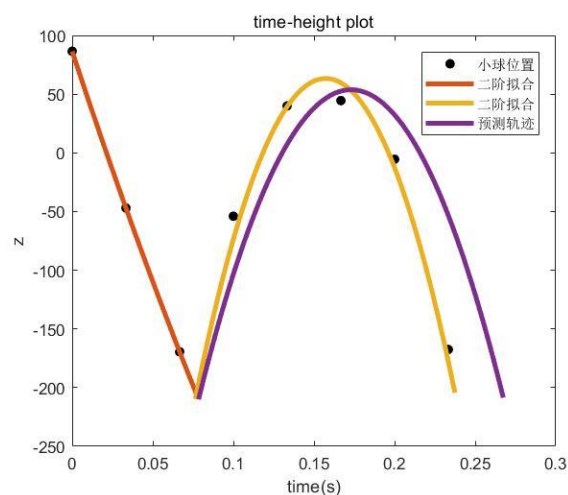
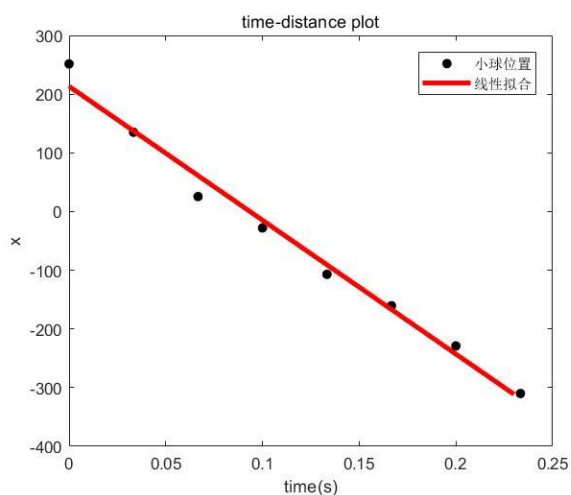


$$\begin{aligned} \text{水平方向: } x &= v_x t \\ \text{竖直方向: } z &= v_z t - \frac{1}{2} g t^2 + z_0 \\ z = 0, v_z^- &= k v_z^+ \end{aligned}$$

恢复系数:  $k$

弹球模型

根据预测模型，从侧视图中获得采样点，通过水平方向的运动计算出速度  $v_x$ ；选取采样点中第一个跳跃周期的样本点，二阶多项式拟合，求得该周期 Z 方向的初始速度  $v_{z(i)}$  和该周期所用时间  $t_i$ ，可以写出下一个周期 Z 方向的运动方程：



$$z = v_{z(i+1)}(t - t_i) - \frac{1}{2}a(t - t_i)^2$$

依次迭代出随后每一个周期 Z 方向的运动方程, 求出目标点所在的周期内的运动方程, 和运动到目标点所需要的时间, 即可求出目标点处的 Z 坐标。

### 2.3 测试效果

依据预测模型编写测试代码, 通过前三帧图作为第一个跳跃周期的样本点二阶拟合, 计算出第一跳跃周期的初始速度和时长, 通过迭代求出预测接球点的坐标。以及一阶线性拟合, 计算出水平方向的运动速度。由于前三帧可能会有图像无法检测乒乓球球心, 所以预测程序无法保证每次运行成功。但在乒乓球初始速度较小的情况下, 基本可以实现预测。

预测效果图如下, 图中的黑色点为预测接球点, 当预测数据不足或无法计算出预测值时黑色点会位于图像坐标系原点处。红色点为乒乓球运动到目标 X 点时检测到的真实位置。



预测效果

我们进行了多次实验并对实验结果进行记录：

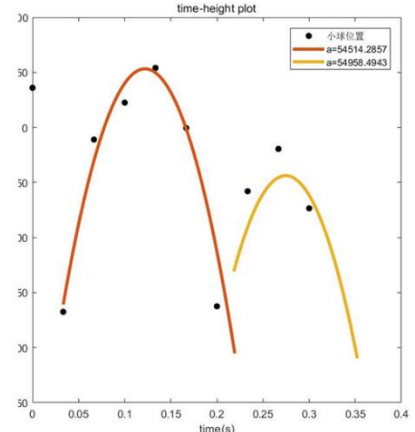
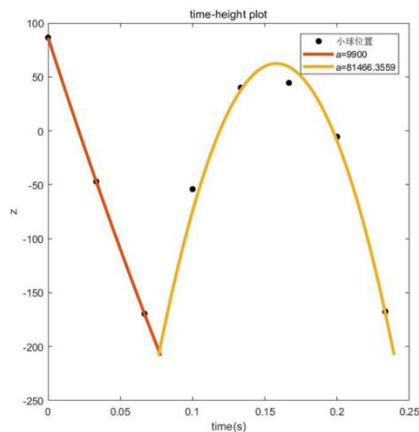
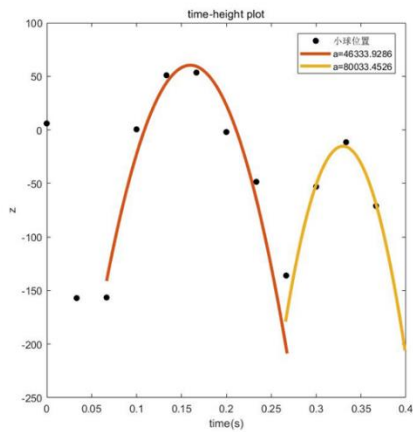
序号	预测值	测量值	误差(cm)
1	-116.55	-9	15.1
2	-168.82	3	24.2
3	135.12	3	18.6
4	-49.72	-51	0.2
5	83.26	-33	16.3
6	-218.65	-78	6.3
7	-192.45	-58.5	18.8
8	-198.27	-153.5	19.8

测试数据

预测值为预测到小球到  $x=-320$  时, 图像坐标系中  $z$  的值。测量值为实际小球运动到该位置时的高度。通过坐标系转换将图像坐标系中的  $z$  坐标转换为空间坐标系中的  $Z_c$  坐标 (单位  $cm$ ), 计算预测和测量值之差即为误差。

从以上数据可以看到, 误差最小为  $0.2cm$ , 可以认为检测准确。但最大  $24.2cm$ , 相当于乒乓球台宽度的一半, 误差较大。平均误差值为  $14.9cm$ , 与满足接打乒乓球的设想有一定差距。

分析导致预测程序误差过大的原因：



1. 模型中我们认为恢复系数  $k$  和  $Z$  方向加速度  $a$  为一个常数，并通过拟合前一个跳跃周期的方法求出加速度  $a$  值。分析几组记录的侧视图数据，分别用二阶多项式拟合求出相邻两周期的加速度  $a$ ，发现同一组数据中不同周期  $a$  值会发生较大变化，并不是所认为的稳定在某一个常数值。恢复系数则是通过根据记录数据手动调整参数，同样无法稳定在某一个常数值附近。
2. 在弹球模型中，我们忽略了许多外力，如空气阻力，空气浮力，和乒乓球旋转造成的马格努斯力。而  $Z$  方向加速度的变化如此巨大说明空气浮力对乒乓球的运动产生较大影响。
3. 我们假设乒乓球  $Y$  方向的初始速度为零，实际上乒乓球在  $Y$  方向具有位移，所以乒乓球在侧面相机中的深度值不断变化，导致求出的空间坐标系坐标有误差，从而预测接球点也有一定误差。
4. 乒乓球台长约 1.2m，相机的每秒帧数为 30fps，乒乓球完全跳过球台只能被记录 10 帧左右的图，即 10 个采样点。而一个周期可记录 3-5 帧，所以用于拟合的样本点太少，导致求出的初始速度和加速度并不准确。