

思路：

1. 初始化：
 - 将摄像头云台的水平位置设置为 90 度，垂直位置稍微向下以观察地面和小球。
2. 检测球体：
 - 使用摄像头实时检测红绿蓝三种颜色的球体或方块。
 - 若在画面中成功检测到球体进行下列操作。
 - 分析摄像头图像，识别球体位置和大小。
 - 获取球体的重心坐标和能够框住球体的最小矩形的宽度和高度。
 - 若摄像头画面中无法探测到球体，则小车进行水平转向寻找球体
3. 判断目标位置：
 - 根据球体的重心坐标，判断其在摄像头视野中的位置。
 - 如果球体位于摄像头中间部分，进入步骤 5。
 - 如果球体偏离摄像头中间部分，进入步骤 4。
4. 调整云台位置：
 - 根据球体在摄像头视野中的位置，控制小车转向，调整的小球在摄像头视野中水平位置至摄像头中心部分。
 - 如果球体大致处于中间部分，将云台水平移动一定角度，使球体位于摄像头中间。
 - 重复调整直到球体位于摄像头中间。
5. 移动小车：
 - 根据球体的大小判断其距离远近。
 - 如果球体较远，小车需要前进一段距离。
 - 如果球体较近，小车需要后退一段距离。
 - 根据实际情况，可能需要多次前进和后退来逐步逼近目标球体。
 - 每行进一段距离后，重复步骤 3 和 4，分远中近，阶段性调整小车的行进方向和摄像头视野。
6. 保持相对位置：
 - 根据摄像头视野中小球的大小，判断小车距离。
 - 当小车移动到合适的距离后，根据球体的重心坐标和大小调整小车的位置。
 - 使小车与球体保持合适的相对位置，这可以通过控制小车的左右移动来实现。
 - 当小车处于合适的相对位置后，根据摄像头的水平此时的水平位置和小球大小，调整小车朝向，使机械臂正向朝向小球。
7. 操作机械臂：
 - 当小车与球体保持合适的相对位置，机械臂朝向正指向小球后，使用脚本指令操作机械臂抓取球体。
 - 根据具体的机械臂设计和控制方式，执行相应的指令来抓取球体或方块。
8. 完成抓取：
 - 机械臂完成抓取动作后，检查抓取是否成功。
 - 如果失败，可以尝试重复上述步骤，重新定位和抓取。

伪代码如下：

```
C > Users > Huang Xianmin > Desktop > 新建 文本文档.txt
1  # 初始化
2  set_turret_horizontal_angle(90) # 将摄像头云台的水平位置设置为90度
3  set_turret_vertical_angle(40)  # 将摄像头云台的垂直位置稍微向下
4
5  # 检测球体
6  ball_detected, ball_center, ball_size = detect_ball() # 检测球体，返回检测结果
7
8  # 判断目标位置
9  if ball_detected:
10     if is_ball_in_center(ball_center):
11         move_to_target() # 进入步骤5，移动小车至目标位置
12     else:
13         adjust_turret_position(ball_center) # 进入步骤4，调整云台位置
14 else:
15     search_for_ball() # 进入步骤2，寻找球体
16
17 # 调整云台位置
18 def adjust_turret_position(ball_center):
19     while not is_ball_in_center(ball_center): #根据水平和垂直位置调整摄像头云台
20         if ball_center.horizontal < TURRET_CENTER_HORIZONTAL:
21             set_turret_horizontal_angle(get_turret_horizontal_angle() + TURRET_ADJUSTMENT_ANGLE)
22         elif ball_center.horizontal > TURRET_CENTER_HORIZONTAL:
23             set_turret_horizontal_angle(get_turret_horizontal_angle() - TURRET_ADJUSTMENT_ANGLE)
24
25         if ball_center.vertical < TURRET_CENTER_VERTICAL:
26             set_turret_vertical_angle(get_turret_vertical_angle() + TURRET_ADJUSTMENT_ANGLE)
27         elif ball_center.vertical > TURRET_CENTER_VERTICAL:
28             set_turret_vertical_angle(get_turret_vertical_angle() - TURRET_ADJUSTMENT_ANGLE)
29
30     ball_detected, ball_center, ball_size = detect_ball() # 重新检测球体位置
31
32 # 移动小车
33 def move_to_target():
34     if ball_size > FAR_THRESHOLD:
35         move_forward(DISTANCE_FAR)
36     elif ball_size < NEAR_THRESHOLD:
37         move_backward(DISTANCE_NEAR)
38
39     # 根据实际情况可能需要多次前进和后退来逼近目标球体
40
41     ball_detected, ball_center, ball_size = detect_ball() # 重新检测球体位置
42
43 # 根据实际情况重复上述步骤至小车到达合适相对位置
44
45 #调整小车朝向使得机械臂指向小球
46 adjust_car_direct(ball_size,turret_vertical_angle,turret_horizontal_angle)
47
48 # 保持相对位置和操作机械臂
49 def maintain_relative_position():
50     while not is_target_reached():
51         adjust_position()
52
53     operate_arm() # 操作机械臂进行抓取
54
55 # 完成抓取
56 if is_grab_success():
57     task_complete()
58 else:
59     retry_or_abort() # 重试或放弃抓取
60
```

仿真软件模拟演示：

