OpenCV：

OpenCV是一个用于图像处理、分析、机器视觉方面的开源函数库。该库采用C及C++语言编写，可以在windows, Linux, mac OSX系统上面运行。该库的所有代码都经过优化，计算效率很高，因为，它更专注于设计成为一种用于实时系统的开源库。OpenCV采用C语言进行优化，而且，在多核机器上面，其运行速度会更快。它的一个目标是提供友好的机器视觉接口函数，从而使得复杂的机器视觉产品可以加速面世。在该作品中，OpenCV主要应用于摄像头对魔方颜色的捕捉和识别。

SurfaceView：

SurfaceView继承了View，其内嵌着专门用于绘制的Surface(绘图表面)，有着自己独立的绘图表明、缓冲区，也就是这个控件不将自己的UI数据填入到它的宿主窗口的绘图表面的图形缓冲区里面去，因此SurfaceView相当于在Window上挖个洞，用来进行绘制。所以在SurfaceView里的Surface和绘制的View是不在UI线程的View hierachy中的。创建Surface时会在SuraceFlinger内部建立对应的Layer，同时在WMS又有自己对应的WIndowState与宿主窗口分离，因此可以不在UI线程进行渲染，也就不会阻塞主线程的事件响应。可以控制刷新频率，同时利用双缓存机制，也就是为了绘制时不会闪烁，会把要绘制的放到内存，然后再一次型绘制出来。在该作品中，SurfaceView主要用于实现拍照功能。

ListView列表组件：

Listview应该是最为常见的控件。对于大多数规则排列的界面，几乎都可以用ListView进行编写。对于单一界面来说，ListView既是最难的控件，又是使用最为频繁的控件。ListView 通常用于展示大量的数据，比如读取数据库中的数据，查看列表。ListView优势也较为明显，比如显示大量数据时节约内存，自带ScrollView 的功能可以实现界面滚动等。在该作品中，ListView主要用于显示已配对的蓝牙设备，帮助连接蓝牙。

Bluetooth蓝牙：

在该作品中，使用 Android Bluetooth API 来完成蓝牙连接的四项主要任务：设置蓝牙、查找局部区域内的配对设备或可用设备、连接设备，以及在设备之间传输数据。通过蓝牙通信，实现上位机与下位机的联系。

魔方算法优化：

图像识别部分：

重点：

①步骤：

1.手机摄像头读入图像。

2.图像预处理：高斯模糊，直方图均衡化，RGB2HSV，HSV2RGB。

3.将图像划分成九个区域，并获取坐标

4.获取每个坐标下的HSV、RGB值

5.得到9\*3的数组，并生成魔方状态步骤

②算法

1.阈值法

阈值法是一种应用十分广泛的方法，它计算量小，鲁棒性强，然而不可避免的会出现精度方面的缺陷，因此需要opencv辅助提高精度

2. KNN分类算法

KNN分类算法的分类依据为该群体的最大相似特征是什么，比如一个物体具有红色这一特征，而另外一个群体的全部物体也具有红色有一特征，那么该物体就属于这个群体。KNN方法与极限原理的相关性极强，然而只有极少量的相邻样。在对Kociemba算法了解后，我们只需将魔方颜色分为与面数相同的类别而不需要判断其具体颜色。对于三阶魔方，只有54个样本，且各类颜色间差异显著，所以在KNN算法下较易达到颜色识别的目的。

难点：

①根据光的折射原理，在不同光照强度下，同一颜色在拍摄后会有不同的

RGB值，而据此得出的数据在定阈值下无法准确将颜色分类。只有当光照强度在一定范围内时，阈值法才能输出准确的结果，这样才能凸显其运算迅速的优点。

由此可见阈值法的局限性很大。

②在拍摄过程中除了拍摄技巧、位置等的外部影响，摄像头的像素及存储照片的质量都是颜色RGB值的重要因素。这两点也会导致识别结果与实际情况有偏差，影响精度。

③蓝牙信号不稳定，下位机向上位机发送指令可能会有延迟，没有拍照就直接识别，影响最终结果。

在作品中将核心算法部分全部交由Kociemba算法，通过优化算法达到减少步数的要求