Android5.0以上截图录屏

主要使用到的类：

MediaProjection:用来捕捉屏幕，可以用来截取屏幕和录制屏幕

MediaProjectionManager：一个系统级服务，用来管理检索MediaProjection

Surface:通过屏幕的conpositor来管理原始缓冲区句柄，通常是用户通过image buffer（如sufaceTexture,MediaRecoder，Allocation）来创建，也可以交给一个控件来绘画界面（如OpenGl,Meidalplayer,CamerDevice）

主要步骤：

1. 使用context.getSystemService(MEDIA\_PROJECTION\_SERVICE)来获取一个MediaProjectionManager的实例
2. 调用MediaProjectionManager.createScreenCaptureIntent()来询问用户获取捕捉屏幕的权限，同时覆写onActivityResult()方法来获取结果
3. 使用MediaProjectionManager.getMediaProjection(int resultCode,Intent resultData)来获取MediaProject的实例,
4. 通过MediaProjection.createVirtrualDisplay(String name,int width,int height,int dpi,int flags,Surface surface,Callback callback,Handler handler)创建VirtualDisplay实例。上述方法中的surface是真正用来截屏或者录屏的

主要方法介绍

1. MediaProjection.createVirtualDisplay(String name,

Int width,

Int height,

Int dpi,

Int flags,

Surface suface,

VirtualDisplay.Callback callback,

Handler handler)

创建一个虚拟显示屏

参数：

Name 虚拟显示屏的名字，不能为空

Width 虚拟屏幕的宽度，必须大于0

Height 虚拟屏幕的高度，必须大于0

Dpi 虚拟屏幕的dpi密度，必须大于1

Falgs 一个虚拟屏幕标志，可以通过DisplayManger查看虚拟屏幕标志集合

Callback 当屏幕发生改变，为空或者0 的时候调用回调函数，

Handler 那个handler中的回调函数应被调用，如果为null则主线程的回调函数会被调用

MediaProjection.registerCallback(MediaProject.Callback callback,

Handler handler)

注册一个监听器，用来接收MediaProjection变化状态

参数：

Callback:将要被调用的回调

Handler:应该调用那个回调程序，

VirtualDisplay.CallBack 一个接收VirtualDisplay的状态变化的信息接口