1.

PTZ 机械镜头 （机械镜头可能有用到光学镜头）

EPTZ 电子镜头

电子镜头对图像缩放会影响像素，而光学镜头不会

mtk芯片 (联发科芯片)

淡入淡出，一张图片变淡，一张图片变深

4就是for （stBox4Head-头的框？）

2就是to

vl\_cam\_resultQ 文件下有很多关键操作

YLakaze （AKAZE局部匹配）

VDO\_IVE\_IMAGE\_S 结构体变量代表的就是一张图像，结构体里有一些指针用来指向图像的地址还有一些图像宽度之类东西。

CC\_BUFFER\_ELEM\_S结构体里有三个VDO\_IVE\_IMAGE\_S 结构体变量指针也就是有三张图像，还有图像的一些时间（u64TimeStamp）、序号之类的东西

Alg表示这个任务工程

IWB 大的平板电脑

Rect 矩形框

zoom缩放大小

vt 语音追踪

<YLprincipalPtsRefine.h> 感觉是镜头设置，误差调整的一些参数

<YLprincipalPtsRefine4Iwb.h>

VDO\_TRACK\_PARAM 视频追踪参数的一个结构体

VDO\_CAMERA\_ATTR\_S 感觉是设置摄像头的各种参数（电子镜头、机械镜头）

重区：两帧重复

静区：很多帧重复

独区没有重复

CC\_BBOXES\_S结构体感觉是一些框设置大小的东西

<utils/Log.h> 安卓日志打印的东西

3840,2160是4K

1280,720是720P

854,480是480P

char acParamPernFile[] = "/phone/config/a.yaml"; //躯干模型参数

char acParamHeadFile[] = "/phone/config/b.yaml";//人头模型参数

char acParamPrecFile[] = "/phone/config/c.yaml";//也是关于人体模型的参数

char acModlePernFile[] = "/phone/config/a.dlc"; //躯干模型

char acModleHeadFile[] = "/phone/config/b.dlc";//人头模型

char acModlePrecFile[] = "/phone/config/c.dlc";//也是关于人体模型

\_win不用了

nce：算法模型的一系列推理平台，通过神经网络啥的，丢进去一张图片就会返回一个框

2.

Windows DLL编程（DLL动态链接技术）

\_\_declspec(dllexport)导出到dll

\_\_declspec(dllimport)从dll导入

fscanf 函数原型为 int fscanf(FILE \* stream, const char \* format, [argument...]); 其功能为根据数据格式(format)，从输入流(stream)中读入数据，存储到argument中。

%hd是一个短整数

%ld是一个长整数

%x的意思是以十六进制数形式输出整数

# FILE \*fp = fopen("/proc/version", "r");获取Ubuntu操作系统的MAC地址等硬件信息

#define QUOTE(name) #name #name的作用就是把#后面的看成[字符串](https://so.csdn.net/so/search?q=%E5%AD%97%E7%AC%A6%E4%B8%B2&spm=1001.2101.3001.7020)

[\_\_DATE\_\_](https://links.jianshu.com/go?to=https%3A%2F%2Fwww.codersrc.com%2Farchives%2F9308.html) 当前日期，一个以 “MMM DD YYYY” 格式表示的字符串常量。

\_\_TIME\_\_ 当前时间，一个以 “HH:MM:SS” 格式表示的字符串常量。

[\_\_FILE\_\_](https://links.jianshu.com/go?to=https%3A%2F%2Fwww.codersrc.com%2Farchives%2F9281.html) 这会包含当前文件名，一个[字符串](https://links.jianshu.com/go?to=https%3A%2F%2Fwww.codersrc.com%2Farchives%2F7815.html)常量。

[\_\_LINE\_\_](https://links.jianshu.com/go?to=https%3A%2F%2Fwww.codersrc.com%2Farchives%2F9270.html) 这会包含当前行号，一个[十进制](https://links.jianshu.com/go?to=https%3A%2F%2Fwww.codersrc.com%2Farchives%2F8488.html)常量。

memcpy是内存拷贝，什么数据都可以

强转变量时，定义变量要考虑对应

比如：int \*retval;

(void \*\*)&retval //这边就是一个二级指针转二级指针

全局变量和静态全局变量的区别：

非静态全局变量的作用域是整个源程序，

静态全局变量的作用域只在定义该变量的源文件内有效，在同一源程序的其他源文件不能使用它。

ADB（全称：Android Debug Bridge），ADB是android sdk里的一个工具, 用这个工具可以直接操作管理android模拟器或者真实的android设备。Android的初衷是用adb这样的一个工具来协助开发人员在开发android应用的过程中更快更好的调试apk，因此adb具有安装卸载apk、拷贝推送文件、查看设备硬件信息、查看应用程序占用资源、在设备执行shell命令等功能。

sprintf函数的格式：int sprintf( char \*[buffer](https://so.csdn.net/so/search?q=buffer&spm=1001.2101.3001.7020), const char \*format [, argument,...] );

除了前两个参数固定外，可选参数可以是任意个。buffer是[字符数组](https://so.csdn.net/so/search?q=%E5%AD%97%E7%AC%A6%E6%95%B0%E7%BB%84&spm=1001.2101.3001.7020)名；format是格式化字符串

sprintf函数的功能与printf函数的功能基本一样，只是它把结果输出到指定的[字符串](https://so.csdn.net/so/search?q=%E5%AD%97%E7%AC%A6%E4%B8%B2&spm=1001.2101.3001.7020)中了

fprintf函数的格式：int fprintf( FILE \*stream, const char \*format, [ argument ]…)

将结果输出到对应的流文件中。

实时线程有静态优先级的概念，里面拥有1-99个静态优先级，数字越到优先级越高

非实时线程没有静态优先级的概念

线程中支持三种调度策略：

抢占式调度策略，在同一静态优先级的情况下，抢占调度策略的线程一旦运行到便会一直抢占CPU资源，而其他同一 一直等到这个抢占式调度策略的线程退出才能被运行到（非实时线程会有一小部分资源分配到） SCHED\_FIFO

轮询式调度策略，在同一静态优先级的情况下，大家一起合理瓜分时间片，不会一直抢占CPU资源（非实时线程会有 分配到） SCHED\_\_RR

其他普通式调度策略，只能作用于非实时线程，由系统自动分配时间片，并且根据运行状态自动分配动态优先级 SCHED\_OTHER

static\_cast< 想要的类型> (原数据)， 是C++中的关键字。相当于C语言的强制类型转换

Wiki是一种在网络上开放且可供多人协同创作的[超文本系统](https://baike.baidu.com/item/%E8%B6%85%E6%96%87%E6%9C%AC%E7%B3%BB%E7%BB%9F/4145999)

int prctl(int option, unsigned long arg2, unsigned long arg3, unsigned long arg4, unsigned long arg5);

prctl(PR\_SET\_NAME, “process\_name”)

第一个参数是操作类型，指定PR\_SET\_NAME，即设置进程名  
第二个参数是进程名字符串，长度至多16字节

IPC（Inter-[Process](https://baike.baidu.com/item/Process/1170280)[Communication](https://baike.baidu.com/item/%20Communication/20394231)，[进程间通信](https://baike.baidu.com/item/%E8%BF%9B%E7%A8%8B%E9%97%B4%E9%80%9A%E4%BF%A1/1235923)

纯虚函数是在声明虚函数时被“初始化”为0的函数。声明纯虚函数的一般形式是

virtual 函数类型 函数名 (参数表列) =0;

注意: ①纯虚函数没有函数体；②最后面的“=0”并不表示函数返回值为0，它只起形式上的作用，告诉编译系统“这是纯虚函数”; ③这是一个声明语句，最后应有分号。

纯虚函数只有函数的名字而不具备函数的功能，不能被调用。它只是通知编译系统: “在这里声明一个虚函数，留待派生类中定义”。在派生类中对此函数提供定义后，它才能具备函数的功能，可被调用。

在 C++ 中，每一个对象都能通过 this 指针来访问自己的地址，为 this 的目的总是指向“这个”对象，所以 this 是一个常量指针

this只能在成员函数中使用。

全局函数，静态函数都不能使用this。

实际上，成员函数默认第一个参数为T\*const register this。（隐藏的）

静态成员变量，它是所有对象所共享的，类内只是声明，类外实现定义。静态成员变量不属于对象，是属于类级别的。 当我们再次计算对象大小的时候，静态成员变量不纳入对象内存的。

注释一段代码

#if 0

#else 后面这段代码有效

#endif

#if 1 后面这段代码有效

#else

#endif

restrict关键字：用来限定指针变量，被该关键字限定的指针变量所指向的内存操作，必须由本指针完成

union联合体

联合的成员是共用一块内存空间的，这样一个联合变量的大小，至少是最大成员的大小

睡眠和挂起是两种行为，阻塞则是一种状态。

操作系统中睡眠、阻塞、挂起的区别形象解释：  
        首先这些术语都是对于线程来说的。对线程的控制就好比你控制了一个雇工为你干活。你对雇工的控制是通过编程来实现的。  
        挂起线程的意思就是你对主动对雇工说：“你睡觉去吧，用着你的时候我主动去叫你，然后接着干活”。  
        使线程睡眠的意思就是你主动对雇工说：“你睡觉去吧，某时某刻过来报到，然后接着干活”。

        线程**阻塞**的意思就是，你突然发现，你的雇工不知道在什么时候没经过你允许，自己睡觉呢，但是你不能怪雇工，肯定你这个雇主没注意，本来你让雇工扫地，结果扫帚被偷了或被邻居家借去了，你又没让雇工继续干别的活，他就只好睡觉了。至于扫帚回来后，雇工会不会知道，会不会继续干活，你不用担心，雇工一旦发现扫帚回来了，他就会自己去干活的。因为雇工受过良好的培训。这个培训机构就是操作系统

在计算联合体大小之前我们必须知道两个知识点：  
1.联合的大小至少是最大成员的大小  
2.当最大成员大小不是最大对齐数的整数倍的时候，就要对齐到最大对齐数的整数倍。

YUV 格式通常有两大类:打包(packed)格式和平面(planar)格式。前者将 YUV 分量存放在同一个数组中,

通常是几个相邻的像素组成一个宏像素(macro-pixel);而后者使用三个数组分开存放 YUV 三个分量,就像是一个三维平面一样。

一、YUV420、YUV420P和YUV420SP区别

YUV420： 即打包格式的YUV420

YUV420P: 即YUV420 Planar，Y\U\V数据是分开存放的

YUV420SP:即YUV420 semi planar, 这个格式的数据量跟YUV420 Planar的一样，但是U、V是交叉存放的

U8C1 就是RGB的一个通道，灰度图

U8C3 就是RGB的三个通道

FIXME：说明此标识处有错误，需要修正

strcpy和memcpy的不同

1）复制的内容不同。strcpy只能复制字符串，而memcpy可以复制任意类型的内容。strcpy只用于字符串复制，并且还会复制字符串的结束符。memcpy对于复制的内容没有限制，用途更广。

2）复制的方法不同。strcpy不需要指定长度，遇到结束符’\0’才会结束，所以容易溢出。memcpy则是根据第三个参数决定复制的长度

3）用途不同。通常在复制字符串时用strcpy，在复制其他类型数据时一般用memcpy。

fread函数用于从文件流中读取数据，其函数原型为：

size\_t fread(void\* buffer, size\_t size, size\_t count, FILE\*stream);

【参数设置】

1) buffer为接收数据的地址，对于fread来书是要读出数据的地址，即数据保存的地址

2) size是要读出内容的单字节数。

3) count是要进行读出size字节的数据项的个数。

4) stream为目标文件指针。

<cutils/properties.h>有什么作用

Pts是什么意思 （<YLprincipalPtsRefine.h>）

CC是什么意思 （CC\_BBOXES\_S，CC\_BBOXES\_S）

MKIT是什么意思

Android Studio推荐使用CMakeLists.txt，对于已经使用Android.mk写的编译文件，转换为CMakeLists.txt即可。

