

Media Guard App V3.1

X:\MediaGuardAppV31 | GIT : MediaGuardAppV31

GIT 操作設置

忽略上存的文件夾：

Build/
Output/
Out/
.vs/

开发总结

2024-11-20

旧版 windows 运作正常, 现在目标是 Version 3.1 版本 修改未 linux 稳定版本.

现在缺余下工作:

1.openssl linux和windows 编译版本

2.curl 版本需要linux编译 主要用于 libcurlhelper.cpp 需要加载https证书的时候要用到openssl 和 curl (curl 是一个http协议指令)

3.ffmpeg 问题:

ffmpeg编译的时候,需要选择CUDA,如果激器没有就不选.

HardAndSoftDecode_Ref.md 参考软解和硬解码.

代码结构

LowLevelAnalysis_MediaGuard.pdf 核心文件三个

1. RtspStreamHandle.cpp 解码单元 對當前鏡頭單元解碼 MP4 FLV HLS JPEG RTMP等等 对解码单元进行管理
2. StreamManager.cpp 創建鏡頭【解码单元】的日期文件夾等等 和當前鏡頭單元相關的 整体控制
例如 StreamManager的構造函數

```
switch (nStreamType)
{
case kStreamTypeUsb:
    m_pHandle = std::make_shared<UsbStreamHandle>();
    break;
case kStreamTypeAudio:
    m_pHandle = std::make_shared<AudioStreamHandle>();
    break;
case kStreamTypeRtsp:
    m_pHandle = std::make_shared<RtspStreamHandle>();
    break;
default:
    m_pHandle = std::make_shared<RtspStreamHandle>();
```

```
break;  
}
```

Struct StreamInfo

StreamInfo 是傳入鏡頭對象的參數對象結果

```
(\MediaGuard\StreamDefine.h)  
1. hard code 實例要修改對應參數  
2. 如果要從雲端獲取則要從幾個API綜合獲取對應參數。  
3. API: CameraMpeg::camera_list;  
CameraMpeg::device_by_serial_no;CameraMpeg::setting_n_schedule_by_camera_id  
4. 還有一個是邏輯轉換的api camera_list_trans_to_strean_info， 主要是把雲端邏輯轉換為具體情況的：  
5. 例如：bRtmp = ttrue是開啟RTMP的，但新規則改為 StreamDecodeType  
(StreamInfo.nStreamDecodeType)的類型，HLS/RTMP/進行切換。  
6. 目前網上雲端沒有bRtmp對應的參數，對應的 CamSettingNSchedule。RtmpOutput和雲端對應的沒有UI改動的，只有默認FALSE.所有對RTMP比較混亂的，所以默認就是FALSE
```

通過共享指針實現獲取控制單元。

1. ManagerController.cpp 例如處理控制 保留全局文件夾Picture
2. CameraMpeg.cpp 主要是API 業務邏輯的操作 記錄媒體文件開始結速時間等等的操作

开发与场景功能描述

1. 实现NVR录像 云存储
2. 各种识别业务与警报：
3. 人脸识别/警报
4. 二维码识别/警报
5. 车牌识别/警报
6. 老鼠识别/警报
7. 行为识别/警报

功能

I:图片动态保存最近15分钟的图片，

II:动态把Camera解码单元（RtspStreamHandle.cpp）的单元图片列表转移保存到全局列表（内存）以供识别业务处理。

III:实现MP4/FLV 动态保存

#全局设置

GlobalSetting 来自/conf/Config.json 的全局配置。

1. picRemainMinutes;设置保存的图片在硬盘缓存时长；
2. videoRemainMinutes = 15; //无论超出容量限制都要最少保留15分钟的video，预设必须考虑最少可以存储15分钟的影片

3. HWDeviceType nHDType = kHWDeviceTypeNone; //硬件类型 默认是没有

操作方法备注

- 1. 增加Ctrl+c Ctrl+Break Enter 行为屏蔽 (signal 监视)
- 2. 只能输入exit退出

|
