Android SDK

IPCameraDemo for basical function，the code only as development reference，some part of function need the developer writing。If has messy code, please setup workspace language-mode as utf-8。

1, The files you need of second develop：

jni: Libs/armeabi/libvstc2_jni.so

Jni interface:vstc2.nativecaller the whole catalogue and files.

Jni back adjust service: BridgeService.java

Do not revise above files, it can be use in your program directly.

2, Video connection flow:

1.Start service:BridgeService;

2.Initialize the server:NativeCaller.PPPPInitial

3. Initialization callback:NativeCaller.Init();

4. Turn on p2p connection:StartPPPP;

5.Turn on video stream after p2p back to online:StartPPPPLivestream

6 Turn off video stream:StopPPPPLivestream

7 Disconnect p2p connection:StopPPPP

8 Release p2p resources: NativeCaller.Free();

Demo program for reference:.

3, Parameter illustruction of video connection progress:

After StartPPPP,returns camera connection process of PPPPMsgNotify(String did, int type, int param) from BridgeService

did:camera sequance number

type：0 menas camera status turns back，1 means relay mode now

param：

```
public static final int PPPP_STATUS_CONNECTING = 0;//connecting
public static final int PPPP_STATUS_INITIALING = 1;//connected, initializing
public static final int PPPP_STATUS_ON_LINE = 2;//online
public static final int PPPP_STATUS_CONNECT_FAILED = 3;// failed to connection
public static final int PPPP_STATUS_DISCONNECT = 4;// connection was turned off
public static final int PPPP_STATUS_INVALID_ID = 5;//invalid UID
public static final int PPPP_STATUS_DEVICE_NOT_ON_LINE = 6;//off line
public static final int PPPP_STATUS_CONNECT_TIMEOUT = 7;//time out to connect
public static final int PPPP_STATUS_WRONGUSER_RIGHTPWD = 8;//wrong password
public static final int PPPP_STATUS_WRONGPWD_RIGHTUSER = 9;// wrong password.
public static final int PPPP_STATUS_WRONGPWD_WRONGUSER = 10;// wrong password
```

StartPPPPLivestream after BridgeService,VideoData (String did, byte[] videobuf, int h264Data, int len, int width, int height, int timestamp, short milistamp, int sessid, int version, int originFrameLen)

did:camera sequance number

videobuf:one frame of video data

h264Data:0 genetal definition video data，1 HD camera data

len:one frame data size

width:one frame data width

height: one frame data height

Another parameters without manipulate temporarily
Please reference to play activity of demo about video manipulate

4, Motion detection:

1,Check the current camera alarm settings
ativeCaller.PPPPGetSystemParams(did, 4);
CallBack_AlarmParams of BridgeService turns back to current camera related alarm parameters, motion_armed (0 means turn off motion detection, 1 means turn on motion detection)
2,Setting alarm message
NativeCaller.PPPPAlarmSetting, motion_armed(0 means turn off motion detection,1 means turn on motion detection) ,other parameters can do the same as above interface.
Please check get_params.cgi of TIP-Camera-CGI.pdf about other parameter details
3,Turn back alarm interface
When the camera setting as open motion detection, camera will show it in the CallBack AlarmNotify of BridgeService after the camera receive motion detection message
public void CallBack_AlarmNotify(String did, int alarmtype){}
did: camera sequence number
alarmtype: 1 motion detection, 2 IO back to alarm

5,local record

1,General camera record.
When h264Data=0 of videodata, you can keep each frame of video data back to the camera directly
2,HD record
HD record request original data, it is different with general camera,HD need NativeCaller.RecordLocal (did,1) when recording
Back to CallBack_H264Data of BridgeService, keep each frame video data

6,Alarm part

All instructions of alarm part are through transmisssion cgi, the interface is

NativeCaller.TransferMessage(did, cmd, 1);
Cmd is cgi instruction
Such as get the camera type String
cmd=" get_status.cgi?loginuse=admin&loginpas=pwd&user=admin&pwd=pwd" ;pwd as camera' s password, then send instruction and get the interface to back;
Some instructions:
Obtain arm and disarm status
"get_sensorstatus.cgi?loginuse=admin&loginpas=" + pwd+ "&user=admin&pwd="
arm:
"set_sensorstatus.cgi?cmd=0&loginuse=admin&loginpas=" + pwd+ "&user=admin&pwd=" + pwd
disarm:

"set_sensorstatus.cgi?cmd=1&loginuse=admin&loginpas=" + pwd+ "&user=admin&pwd=" + pwd
Get a bound sensor:
"get_sensorlist.cgi?loginuse=admin&loginpas=" + pwd+ "&user=admin&pwd=" + pwd
Get a bound guard:
"get_sensor_preset.cgi?cmd=1&sensorid=0&loginuse=admin&loginpas="+pwd+"&user=admin&pwd=" + pwd
setting guard:
"set_sensor_preset.cgi?sensorid="+ 255 + "&presetid=" + pos +
"&loginuse=admin&loginpas=" + pwd
pos can set 0-5
The ring of open & close the door:
"set_sensorstatus.cgi?cmd=4&doorbell=1&loginuse=admin&loginpas="+pwd+"&user=admin&pwd=" + pwd
"set_sensorstatus.cgi?cmd=4&doorbell=0&loginuse=admin&loginpas="+pwd+"&user=admin&pwd=" + pwd
Edit sensor name
"set_sensorname.cgi?" + "&sensorid="
+ id + "&sensorname=" + name + "&loginuse=admin&loginpas="
+ pwd
Id as index when back to sensor
Name as to set the name of sensor
Open coding match status:
"set_sensorstatus.cgi?cmd=2&loginuse=admin&loginpas=" + pwd+ "&user=admin&pwd=" + pwd
Close coding-match
"set_sensorstatus.cgi?cmd=3&loginuse=admin&loginpas=" + pwd+ "&user=admin&pwd=" + pwd
Delect sensor:
"del_sensor.cgi?" + "&sensorid="+ id + "&loginuse=admin&loginpas=" + pwd+
"&user=admin&pwd=" + pwd
Id as index when back to sensor
Setting alarm preset position:
"set_sensor_preset.cgi?sensorid="+ id + "&presetid=" + pos + "&sensorid=" +

sensorid
+ "&loginuse=admin&loginpas=" + pwd
Id as index when back to sensor
Pos0-5
Sensor type:
SENSOR_TYPE_DOOR = 0x01,//door sensor
SENSOR_TYPE_INFRARED = 0x02,//PIR
SENSOR_TYPE_SMOKE = 0x03,//smoke detector
SENSOR_TYPE_SMELL = 0x04,//gas detector

SENSOR_TYPE_REMOTE = 0x07,//controller
SENSOR_TYPE_CAMERA = 0x0A,//camera
SENSOR_TYPE_CURTAIN = 0x0B//curtain

7,NativeCaller interface parameter illustruction:

Please check TIP-Camera-CGI.pdf for more parameter details

//initializing server
void PPPPInitial(string svr);
//start to search, build multi-radio / radio，default to use radio
Void StartSearch();
// stop searching
void StopSearch();
//initializing video decoder
Void Init();
//release and close decoder, release PPPP resources
voidFree();
//open P2P server，transmission UID 、user name、password、1、""
int StartPPPP(String, String, String, int,String);
//stop p2p service
int StopPPPP(string);
//request video data，transmission uid、streamid、substreamid
int StartPPPPLivestream(string , int, int);
//stop video data，transmission uid
int StopPPPPLivestream(string);
//pan&tilt control，the first parameter uid ，the second parameter control command
int PPPPPTZControl(string, int);
//image sensor parameter setting，three parameters: uid，camera parameter type，camera parameter
int PPPPCameraControl(string, int, int);
//request the voice from Camera，transmission uid
int PPPPStartAudio(string);
int PPPPStopAudio(string);
//send radio data，transmission uid
int PPPPStartTalk(string);
int PPPPStopTalk(string);
int PPPPTalkAudioData(string, byteArray, int);
//detect internet information
int PPPPNetworkDetect();
//initialization callback function
int PPPPSetCallbackContext(object context);
//obtain system parameter，transmission uid 、news type，news type can be defined in P2P_API_Define.h
int PPPPGetSystemParams(string did, int paramType);
//restart
int PPPPRebootDevice(string did);

// Restore factory setting
int PPPPRestorFactory(string did);
//setting devicce wifi parameter, set_wifi.cgi
int PPPPWifiSetting(string did, int enable, string ssid, int channel, int mode,int authtype, int encryp, int keyformat,
int defkey, string key1, string key2, string key3, string key4, int key1_bits, int key2_bits, int key3_bits, int key4_bits, string wpa_psk);
//setting device internet parameter, set_network.cgi
int PPPPNetworkSetting(string did, string ipaddr, string netmask, string gateway, string dns1, string dns2, int dhcp, int port, int rtsport);
//setting device user parameter, include visitor, operator, administrator parameter， set_users.cgi
int PPPPUserSetting(string did, string user1, string pwd1, string user2, string pwd2, string user3, string pwd3);
//setting camera date, time and parameter, please refer to set_datetime.cgi
int PPPPDatetimeSetting(string did, int now, int tz, int ntp_enable, string ntp_svr);
//setting camera DDNS 、parameter please refer to set_ddns.cgi
int PPPPDDNSSetting(string did, int service, string user, string pwd, string host, string proxy_svr, int ddns_mode, int proxy_port);
//email settings， set_mail.cgi
int PPPPMailSetting(string did, string svr, int port, string user,
string pwd, int ssl, string sender, string receiver1, string receiver2, string receiver3, string receiver4);
//Ftp settings， set_ftp.cgi
int PPPPFtpSetting(string did, string svr_ftp, string user, string pwd, string dir, int port, int mode, int upload_interval);
//camera pan&tile parameter setup， set_misc.cgi
int PPPPPTZSetting(string did, int led_mod, int ptz_center_onstart, int ptz_run_times, int ptz_patrol_rate, int ptz_patrul_up_rate,
int ptz_patrol_down_rate, int ptz_patrol_left_rate, int ptz_patrol_right_rate, int disable_preset);
//setting camera alarm selections， set_alarm.cgi
int PPPPAlarmSetting(string did, int motion_armed, int motion_sensitivity, int input_armed, int ioin_level, int iolinkage, int ioout_level,int alarmpresetsit, int mail, int snapshot, int record, int upload_interval, int schedule_enable, int schedule_sun_0, int schedule_sun_1, int schedule_sun_2, int schedule_mon_0, int schedule_mon_1, int schedule_mon_2, int schedule_tue_0, int schedule_tue_1, int schedule_tue_2, int schedule_wed_0, int schedule_wed_1, int schedule_wed_2, int schedule_thu_0, int schedule_thu_1, int schedule_thu_2, int schedule_fri_0, int schedule_fri_1, int schedule_fri_2,
int schedule_sat_0, int schedule_sat_1, int schedule_sat_2);
//setting record， set_recordsch.cgi
int PPPPSDRecordSetting(string did,
int record_cover_enable, int record_timer, int record_size, int record_time_enable, int record_schedule_sun_0, int record_schedule_sun_1,int record_schedule_sun_2,
int record_schedule_mon_0, int record_schedule_mon_1, int record_schedule_mon_2, int record_schedule_tue_0, int record_schedule_tue_1, int record_schedule_tue_2,

int record_schedule_wed_0, int record_schedule_wed_1, int record_schedule_wed_2, int record_schedule_thu_0, int record_schedule_thu_1, int record_schedule_thu_2,
int record_schedule_fri_0, int record_schedule_fri_1, int record_schedule_fri_2, int record_schedule_sat_0, int record_schedule_sat_1, int record_schedule_sat_2);
//obtain record file list，get_record_file.cgi
int PPPPGetSDCardRecordFileList(string did, int startTime, int endTime);
//start to play back，livestream.cgi
int StartPlayBack(string did, string filename, int offset,int picTag);
int StopPlayBack(string did);
//yuv420 转 rgb565
intYUV4202RGB565(byteArray yuv, byteArray rgb, int width, int height);
//decoding h264 frame
int DecodeH264Frame(byteArray h264, int bIFrame, byteArray yuv, int length, intArray size);
int SetBill(string did, string svr, int port, string user, string pwd, int reboottime);
//set_syswifi.cgi
int SetAPParam(string did, string ssid, string pwd);
//upgrade new firmware
int UpgradeFirmware(string did, string svr, string fpath,int ftype);
//xq -06-18
int FormatSD(string did);
//pass CGI
int TransferMessage(string did, string msg, int len);
1，HZ:
50hz:NativeCaller.PPPPCameraControl(did, 3, 0);
60hz:NativeCaller.PPPPCameraControl(did, 3, 1);
2,digitalzoom：
amplify：NativeCaller.*PPPPCameraControl*(did, 18, 1);
stop： NativeCaller.*PPPPCameraControl*(did, 18, 0);
Lessen：NativeCaller.PPPPCameraControl(strDID, 17, 1);
stop：NativeCaller.PPPPCameraControl(strDID, 17, 0);
3, wifi setup：
how to set :
NativeCaller.PPPPWifiSetting
Please refer to cgi file and know more about set_wifi.cgi parameter illustration；
Steps：
① Request current camera wifi setting parameter
NativeCaller.PPPPGetSystemParams(did, 4);

About CallBack_WifiParams back to parameter interface,please refer to wireless parameter of get_params.cgi of cgi ；
② Obtain camera wifi signal NativeCaller.PPPPGetSystemParams(did, 20);
Then back to CallBack_WifiScanResult parameter illustration in service：
* **@param** did camera uid
* **@param** ssid wifi ssid

* **@param** mac wifi mac
* **@param** security
0->means WEP-NONE
1->means WEP
2->WPA-PSK TKIP
3->WPA-PSK AES
4->WPA2-PSK TKIP
5->WPA2-PSK AES
* **@param** dbm0 signal intensity
* **@param** dbm1
* **@param** mode working mode，0->infra 1->adhoc
* **@param** channel channel
* **@param** bEnd 1 finished to search

③ select corresponding wifi to setup;

4.sd card setup:

The way to set: NativeCaller.PPPPSDRecordSetting

Please refer to set_recordsch.cgi of cgi file about the parameter

The record time is divide one day into three periods. Such as schedule_sun_0, schedule_sun_1, schedule_sun_2, each period divide into 32 mini periods, 15 minutes per period, record as 1, un-record as 0

Obtain current SD card status: NativeCaller.PPPPGetSystemParams(did, 22), back to service interface way, callBackRecordSchParams parameter illustration please refer to get_record.cgi of cgi file.

5.Alarm settings:

How to set: NativeCaller.PPPPAlarmSetting

Please refer to set_alarm.cgi of cgi file about parameter

Obtain current camera alarm settings:

NativeCaller.PPPPGetSystemParams(did, 4); back to sevice way: please refer to get_params.cgi of cgi files about alarm parameter;

When PPPPAlarmSetting motion_armed=1 means open motion detection，when trigger motion detection, BridgeService CallBack_AlarmNotify(String did, int alarmtype) has return back，did means the camera uid already trigger motion detection，alarmtype=1 means motion detection alarming, alarmtype=2 means IO linkage alarm