

# 接口文档

---

## iLoboke | 足球机器人接口说明

文档版本号 (V1.0)



苏州南江乐博机器人有限公司



<b>1. 接口说明</b>	<b>3</b>
<b>2. 接口清单</b>	<b>3</b>
<b>3. 接口详细说明</b>	<b>3</b>
<b>3.1 视觉信息接口</b>	<b>3</b>
3.1.1 基础说明	3
3.1.2 接口详细参数说明	4
3.1.2.1 接口数据结构 FieldVisionMsg（详见 basevision.h）	4
● 清除 push_back 函数造成内存增长	5
3.1.2.2 接口数据结构 Robot（详见 basevision.h）	5
接口数据结构 point2f（详见 basevision.h）	5
3.1.2.3 接口类 RoboCupSSLClient	5
● 打开 UDP 通信	5
● 关闭 UDP 通信	5
● 接收视觉信息	6
● 设置通信多播地址	6
● 设置通信端口	6
3.1.2.4 接口类 VisionMessage	6
● 更新场上视觉数据且将国际标准视觉信息格式转变成自定义视觉信息格式	6
● 把两个相机半场视觉数据组成一帧全场视觉数据	6
● 打印全场视觉信息	7
3.1.2.5 接口类 SSL_WrapperPacket	7
● 检测视觉包信息	7
<b>3.2 无线通信接口</b>	<b>7</b>
3.2.1 基础说明	7
3.2.2 接口详细参数说明	9
3.2.2.1 接口类 CWirelessModule	9
● 构造函数	9
● 打开串口	9
● 设置发射机频点	10
● 获取来自机器人反馈的信息	10
● 向机器人发送消息	10
3.2.2.2 接口结构 ROBOTCOMMAND	11
<b>4. DEMO 说明</b>	<b>11</b>
<b>5. 附：无线通信包结构</b>	<b>11</b>
5.1 启动包（设置发射机）	11
5.2 机器人通讯包格式	12

## 1. 接口说明

本文档主要提供了使用 iLoboke I 足球机器人所需要的视觉信息接口以及无线通信接口。用户可以使用视觉信息接口，获取处理过的视觉识别信息，在此基础上，可以自主开展路径规划、运动算法、机器人调度等任务，最后通过无线通信接口，实现对机器人硬件的控制。

## 2. 接口清单

序号	接口类型	接口方式	备注
1	视觉信息接口	SDK 调用	详见： VisionSDK .zip
2	无线通信接口	SDK 调用	详见： WirelessSDK.zip

## 3. 接口详细说明

### 3. 1 视觉信息接口

#### 3. 1. 1 基础说明

视觉 SDK 主要针对 iLoboke I 足球机器人的视觉数据进行封装，连接视觉机后，通过使用 SDK 提供的接口，可以获取视觉识别信息。

- 头文件：prtoBuf 头文件(打包在 google 文件中)

messages\_robocup\_ssl\_detection.pb.cc

messages\_robocup\_ssl\_detection.pb.h

messages\_robocup\_ssl\_geometry.pb.cc

messages\_robocup\_ssl\_geometry.pb.h

messages\_robocup\_ssl\_wrapper.pb.cc

messages\_robocup\_ssl\_wrapper.pb.h

basevision.h

mixvision.h

netraw.h  
processvision.h  
robocup\_ssl\_client.h  
util.h  
vector.h

- 库文件:

VisionSDK.lib (Release 版)  
VisionSDKD.lib (Debug 版)  
libprotobuf.lib (Release 版)  
libprotobufD.lib (Debug 版)

- 对应的库路径: lib 文件

**注意事项:**

1. 编译环境: Visual Studio Ultimate 2013;
2. 由于 windows.h 头文件与 winsock2.h 冲突; 需要在预编译选项添加宏才能编译通过, 宏如下:

WIN32\_LEAN\_AND\_MEAN  
\_SCL\_SECURE\_NO\_WARNINGS  
\_WINSOCK\_DEPRECATED\_NO\_WARNINGS

3. 宏添加路径: 右键点击项目->属性->c/c++->预编译->预编译定义

### 3. 1. 2 接口详细参数说明

#### 3. 1. 2.1 接口数据结构 FieldVisionMsg (详见 basevision.h)

字段名称	类型	备注
cameraId	int	相机编号
frame	int	图像帧号,从 1 开始,最大值为 4095。超过 4095 后,又从 1 开始
ball	vector<point2f>	球坐标
blueRobot	vector<Robot>	蓝队视觉信息
yellowRobot	vector<Robot>	黄队视觉信息

### ● 清除 `push_back` 函数造成内存增长

方法名	clear		
输入参数	参数名	参数类型	参数说明
返回参数		void	

### 3. 1. 2.2 接口数据结构 Robot（详见 `basevision.h`）

字段名称	类型	备注
id	int	车号，（0~11）
orientation	int	机器人朝向角（与 X 轴正方向夹角）
robot	point2f	机器人全局坐标

### 接口数据结构 point2f（详见 `basevision.h`）

字段名称	类型	备注
x	float	X 轴坐标
y	float	Y 轴坐标

### 3. 1. 2.3 接口类 RoboCupSSLClient

#### ● 打开 UDP 通信

方法名	open		
输入参数	参数名	参数类型	参数说明
	blocking	bool	如果为 false，表示不阻塞的方式进行 udp 通信，true 表示阻塞方式通信，默认 false
返回参数		int	返回值： 2：成功 其他：失败

#### ● 关闭 UDP 通信

方法名	close		
输入参数	参数名	参数类型	参数说明
返回参数			

### ● 接收视觉信息

方法名	receive		
	参数名	参数类型	参数说明
输入参数	packet	SSL_WrapperPacket	google buffer 消息格式,
返回参数		bool	如果接受到信息, 返回 true, 否则返回 false

### ● 设置通信多播地址

方法名	setAddress		
	参数名	参数类型	参数说明
输入参数	net_ref_address	String	多播地址
返回参数		void	

### ● 设置通信端口

方法名	setPort		
	参数名	参数类型	参数说明
输入参数	Port	Int	端口
返回参数		void	

## 3. 1. 2.4 接口类 VisionMessage

### ● 更新场上视觉数据且将国际标准视觉信息格式转变成自定义视觉信息格式

方法名	updateMsg		
	参数名	参数类型	参数说明
输入参数	detecte	SSL_DetectionFrame	国际视觉标准格式 protobuf
	currentFieldMsg	FieldVisionMsg	自定义视觉信息格式
返回参数		void	

### ● 把两个相机半场视觉数据组成一帧全场视觉数据

方法名	combinateFrame		
	参数名	参数类型	参数说明
输入参数	currentFieldMsg	FieldVisionMsg	当前场地未组包的视觉数据

返回参数		bool	返回值: True:成功 False:失败
------	--	------	-----------------------------

- 打印全场视觉信息

方法名	printfMsg		
	参数名	参数类型	参数说明
输入参数	currentFieldMsg	FieldVisionMsg	当前场上视觉数据
返回参数		void	

### 3. 1. 2.5 接口类 SSL\_WrapperPacket

- 检测视觉包信息

方法名	has_detection		
	参数名	参数类型	参数说明
输入参数			
返回参数		bool	返回值: True:检测到 False:未检测到

### 3. 2 无线通信接口

#### 3. 2. 1 基础说明

无线通讯 SDK 主要要用于上层程序通过无线发射机对小 iLoboke I 足球机器人进行控制。

本 SDK 提供了一个用于接口通讯的类 CWirelessModule，其中构造函数的第一个参数为串口号，其他三个参数使用默认值，如果使用其他值可能造成无法正常通讯。

setFreq()函数用于设置频点。对于刚加电的无线发射机，必须先调用此函数，然后才能调用下面的函数进行通讯。对于已经进行加电并完成频点设置的无线发射机，则可以不再调用此函数。此函数只需要调用一次即可。

doWirelessModule()函数用于上层程序向对应机器人发送运动消息；

RequestSpeedInfo()函数用于提取来自机器人反馈的信息。

- 头文件：WirelessModule.h

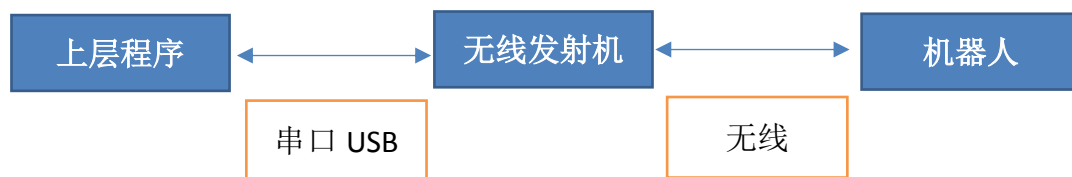
- 库文件：

WirelessComm.lib（Release 版）

WirelessCommD.lib（Debug 版）

需要在编译自己的程序的时候链接上面的库文件之一。

- 上层程序与机器人之间的通讯方式如下：



无线发射机与上层主机之间的连接通过 USB 接口完成，一般的 Windows 操作系统都能直接识别并自动安装驱动程序，如果不能自动识别，请安装 lib 目录下的“小型足球发射机驱动.rar”，解压后运行 PreInstall.exe。

无线发射机连接到主机后，从主机设备管理器的“端口（COM 和 LPT）”中可以看到新增加的 COM 口，例如“Silicon Labs CP210x USB to UART Bridge(COM3)”，其中 COM 后的数字即为串口号。如果未发现，请重新安装驱动程序，或联系技术支持人员。

上述的串口号在我们开发中需要用到。

- 频点设置：

为了让一个球队能通过一个发射机控制本队所有机器人，而不同的球队之间不造成干扰，所以每个球队都将有一个自己的用于通讯控制的无线频点，频点需要在无线发射机上进行配置，同时机器人上有对应的拨码盘来设置自己通讯的无线频点。

无线发射机上的频点只能设置一次，后续再重新设置将无效。如果确实需要更换频点，请断开无线发射机电源，重新加电，然后再设置新频点。

### 注意事项：

1. 无线发射机的频点和需要控制的机器人的频点必须一致。



2. 无线发射机状态可以通过两个指示灯进行初步判断：刚启动的时候为交替闪烁；如果同步闪烁则表示未成功设置频点；常亮则表示频点设置成功；在发送或接收数据时外侧指示灯闪烁。
3. 如果频点设置不能成功，请考虑在频点设置函数调用后进行一段时间的sleep，然后再进行与机器人之间的通讯数据发送。
4. doWirelessModule()、RequestSpeedInfo()两个函数对于机器人的车号的使用有区别，请注意。
5. 函数具体使用请参见 demo 中的例子。
6. 通讯中使用的数据结构请参见头文件中的相应说明。
7. 编译环境：Visual Studio Ultimate 2013。

### 3. 2. 2 接口详细参数说明

#### 3. 2. 2.1 接口类 CWirelessModule

##### ● 构造函数

方法名	CWirelessModule		
输入参数	参数名	参数类型	参数说明
	COM_NUM	int	无线发射机连接的串口号，默认值为 1，需要根据实际值进行设置
	COM_RATE	unsigned int	串口通信波特率，采用默认值 115200.
	PARTY	int	串口通信校验位，采用默认值 0.
返回参数	PVERSION	int	串口通信数据报文版本号，采用默认值 2016
		void	

##### ● 打开串口

方法名	openComm		
输入参数	参数名	参数类型	参数说明
	Comm_num	int	串口号

返回参数		int	
------	--	-----	--

## ● 设置发射机频点

方法名	setFreq		
输入参数	参数名	参数类型	参数说明
	freq	int	频点设置（取值范围：0~15）
返回参数		void	0:成功 101:打开失败 200-900:内部错误

## ● 获取来自机器人反馈的信息

方法名	RequestSpeedInfo		
参数 1	参数名	参数类型	参数说明
	n	int	小车的实际车号（取值范围 1~12）
参数 2	info	ROBOTINFO *	返回的信息内容，后附结构说明
返回参数		int	返回值。 -1：车号不正确， -2：内存申请失败， >=0 并且 <=999：成功返回，0~999 之间循环

## ● 向机器人发送消息

方法名	doWirelessModule		
参数 1	参数名	参数类型	参数说明
	index	int	小车的实际车号 -1（注意：需要减 1，也就是 1 号车需要传入 0）（取值范围：0~11）
参数 2	robotCommand	ROBOTCOMMAND&	命令信息内容
返回参数		void	

### 3. 2. 2.2 接口结构 ROBOTCOMMAND

字段名称	类型	备注
shoot	Unsigned int	踢球力度(0~127)
cb	Unsigned char	是否吸球(true:是,false:否)
x	double	机器人前后方向速度(-511~511) (大于 0 时向前, 小于 0 向后), 单位:1cm/s
y	double	机器人左右方向速度(-511~511) (大于 0 向左, 小于 0 向右), 单位:1cm/s
rotate	double	机器人轴向转动速度(-511~511) (大于 0 时顺时针, 小于 0 时逆 时针),单位:1/40rad/s

## 4. Demo 说明

1. VisionSDK .zip 和 WirelessSDK.zip 中分别提供了使用接口的 Demo
2. Demo.zip 提供了一个完整使用两个 SDK 的案例

实现任务：使场上的黄队 1 号车从 X 坐标绝对值大于 100mm 的位置到达 X 坐标绝对值小于或等于 100mm 的位置，其中发射机串口端口号为 3，车和发射机频点为 1

## 5. 附：无线通信包结构

在自行开发了通讯模块的基础上，可以按照以下提供的包结构，给机器人发送通讯包。

### 5.1 启动包（设置发射机）

```
ff b0 01 02 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 31
```

```
//start
```



ff b0 04 05 06 19 00 00 00 00 00 00 00 00 00 00 07 00 00 00 00 00 00 00 c2

//config data 以 9 号频点为例

注:

1. 先发第一个包，收到回包后发第二个包，回包无需处理
2. 第二个包中的“第六字节低四位”表示频点

## 5.2 机器人通讯包格式



位	7	6	5	4	3	2	1	0
字节 (功能)								
0 (Header)	1	1	1	1	1	1	1	1
1 Robots ID <sup>[1]</sup>	0				Robot12	Robot11	Robot10	Robot9
2 Robots ID <sup>[1]</sup>	Robot8	Robot7	Robot6	Robot5	Robot4	Robot3	Robot2	Robot1
3 (misc) <sup>[2]</sup>	0 控球正 <sup>μ</sup> 1 控球反	平射(0) <sup>μ</sup> 挑射(1)	控球档 (0-3)		上传模式 <sup>[3]</sup>	保留, 0	0	1
4 速度控制 <sup>[4]</sup>	保留, 0	保留, 0	Vx 符号位	Vx 高 5 位				
5 速度控制	x 向速度绝对值 vx 低 4 位, 单位: 1cm/s.				Vy 符号位	Vy 高 3 位		
6 速度控制	y 向速度绝对值 vy 低 6 位, 单位: 1cm/s.						w 符号位	w 高 1 位
7 速度控制	沿轴向转速绝对值 w 低 8 位, 单位: 1/40 rad/s.							
8 射门力度 <sup>[5]</sup>	0	R1_pow6	R1_pow5	R1_pow4	R1_pow3	R1_pow2	R1_pow1	R1_pow0
9 (misc)	0 控球正 <sup>μ</sup> 1 控球反	平射(0) <sup>μ</sup> 挑射(1)	控球档 (0-3)		上传模式	ρ	ρ	ρ
10 速度控制	保留, 0	保留, 0	Vx 符号位	Vx 高 5 位				
11 速度控制	x 向速度绝对值 vx 低 4 位, 单位: 1cm/s.				Vy 符号位	Vy 高 3 位		
12 速度控制	y 向速度绝对值 vy 低 6 位, 单位: 1cm/s.						w 符号位	w 高 1 位
13 速度控制	沿轴向转速绝对值 w 低 8 位, 单位: 1/40 rad/s.							
14 射门力度	0	R1_pow6	R1_pow5	R1_pow4	R1_pow3	R1_pow2	R1_pow1	R1_pow0
15 (misc)	0 控球正 <sup>μ</sup> 1 控球反	平射(0) <sup>μ</sup> 挑射(1)	控球档 (0-3)		上传模式	ρ	ρ	ρ
16 速度控制	保留, 0	保留, 0	Vx 符号位	Vx 高 5 位				
17 速度控制	x 向速度绝对值 vx 低 4 位, 单位: 1cm/s.				Vy 符号位	Vy 高 3 位		
18 速度控制	y 向速度绝对值 vy 低 6 位, 单位: 1cm/s.						w 符号位	w 高 1 位
19 速度控制	沿轴向转速绝对值 w 低 8 位, 单位: 1/40 rad/s.							
20 射门力度	0	R1_pow6	R1_pow5	R1_pow4	R1_pow3	R1_pow2	R1_pow1	R1_pow0
21	0x07							
22	0							
23	保留, 0							
24	0	0	0	0	0	0	0	0

样例:

- 【使 1 号车以 100cm/s 的向前速度和 1rad/s 顺时针轴向转速运动】

```

[0]      FF
[1]      00
[2]      01
[3]      01
[4]      06
[5]      40
[6]      00
[7]      28

```



[8]	00
[9]	00
[10]	00
[11]	00
[12]	00
[13]	00
[14]	00
[15]	00
[16]	00
[17]	00
[18]	00
[19]	00
[20]	00
[21]	07
[22]	00
[23]	00
[24]	00

2. 【使 1 号车以 100cm/s 的向后速度和 100cm/s 的向右速度运动】

[0]	FF
[1]	00
[2]	01
[3]	01
[4]	26
[5]	41
[6]	90
[7]	28
[8]	00
[9]	00
[10]	00
[11]	00
[12]	00
[13]	00
[14]	00
[15]	00
[16]	00
[17]	00
[18]	00
[19]	00
[20]	00
[21]	07
[22]	00
[23]	00
[24]	00



## 3. 【最大力度踢球】

[0]	FF
[1]	00
[2]	01
[3]	01
[4]	00
[5]	00
[6]	00
[7]	00
[8]	7F
[9]	00
[10]	00
[11]	00
[12]	00
[13]	00
[14]	00
[15]	00
[16]	00
[17]	00
[18]	00
[19]	00
[20]	00
[21]	07
[22]	00
[23]	00
[24]	00