|  |  |  |
| --- | --- | --- |
| 2012/4/23 | 张斌 | 创建 |
|  |  |  |

# 1.概述

影像管理系统是DVR项目的一个子系统，其主要实现对出租车采集的影像资料管理、分析、存储检索等功能。

海量的影像数据的采集和存储都是离散的存储在影像系统中，由内部网络提供IO和数据访问能力。

系统关系图

访问影像系统需要一套标准的接口来实现，接口是基于网络服务形式的，并以插件的方式提供管理和访问功能，插件接口定义见后文。

# 2.系统组成

Tcp/ip

主机环境

作业软件

影像系统

前台控制程序

插件接口

IPC

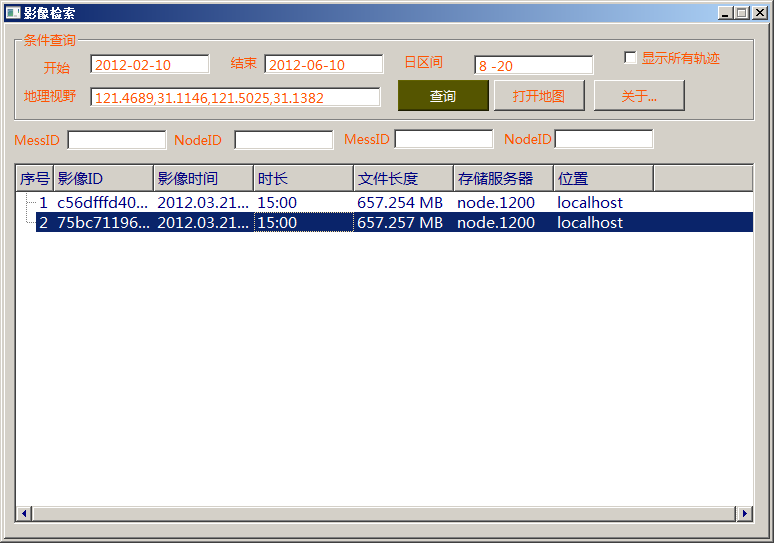
系统结构示意图

影像系统由若干后台服务进程和前台控制软件组成

1. 后台服务进程： 实现数据存储、检索、调度管理等功能
2. 前台控制软件： 提供作业人员访问、检索影像的操作功能。控制软件以独立进程方式运行，其好处在于降低与客户软件的耦合，保证系统功能独立、运行的稳定性。
3. 插件接口： 驻留在作业软件进程中的功能接口，提供函数级的调用，并将其调用转化为与前台控制软件和影像后台服务的通信服务。



影像回放控制



影像检索管理



地图模拟客户端

# 3.数据结构

## 1.GeoPoint\_t

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| x | Float |  |
| Y | Float |  |
| 地理坐标点信息 | | |

## 2.GeoRect\_t

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| x | Float |  |
| y | Float |  |
| Width | Float |  |
| Height | Float |  |
| 地理区域信息 | | |

## 3.GpsPoint\_t

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| Lon | Float |  |
| lat | Float |  |
| time | Uint32 | Unix timestamp 1970~ |
| speed | float |  |
| angle | float |  |
| gps坐标信息点 | | |

## 4.GpsLine\_t

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| first | GpsPoint\_t | 轨迹线段开始点 |
| second | GpsPoint\_t | 轨迹线段开始点 |
| Gps行驶路段轨迹，由开始点和结束点构成 | | |

## 5.GpsPath\_t

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| id | Uint32 | 轨迹唯一编号 |
| lines | GpsLine\_t \* | 线段集合 |
| size | Uint32 | 线段数量 |
| 影像行驶轨迹，由若干个预处理的线段构成 | | |

# 4.接口定义

## 1.更改可视地理区域

函数: void GeoSightChanged ( GpsRect\_t rect)

功能：作业软件改变视野区域后，需通知影像控制端，以便进行对当前视野区域的影像轨迹检索

参数: rect – 可视地理区域；或者需要检索的地理区域

返回:

## 2.影像轨迹选择

函数: void ImagePathClicked ( uint32 id, uint32 timetick)

功能： 地图上选择影像轨迹之后通知影像控制端进行影像回放,影像将直接跳跃到指定的时间点回放。

参数:

id – 轨迹的编号

timetick – 点击轨迹上最接近的gps时间( unix timestamp 1970~)

返回:

## 3.加载影像轨迹到地图

函数【回调】: void ImagePathShow( GpsPath\_t path)

功能：影像控制端刷选出轨迹之后，发送到作业地图上显示

参数: path – 影像轨迹，由若干line组成，这些line已经被预先处理，便于实现时间跳跃

返回:

## 4.地图影像轨迹选择

函数【回调】: void ImagePathSelected ( uint32 id)

功能：影像控制端选择影像记录，通知作业软件在图上高亮选择影像轨迹

参数: id – 轨迹编号

返回:

## 5.清除图上影像轨迹

函数【回调】: void ClearImagePath ( uint32[] ids, uint32 size)

功能：

参数:

返回:

## 6.影像回放定位

函数【回调】: void ShowSymbol ( GpsPoint\_t gps)

功能：影像控制端在回放视频时，计算出当前车辆轨迹点，发送到地图要求作业软件显示当前作业车辆的位置信息

参数: gps

返回:

# 5.其他

功能Demo演示：<http://sw2us.com/sites/default/files/flvshow/dvr.demo.html>

下载视频：<http://sw2us.com/sites/default/files/dvr.demo2.avi>

Python消息定义

#地理显示区域改变 , mapapp -> image\_client

#地图改变发送通知到影像客户

class MsgPlugin\_GeoSightChanged\_1(MsgPlugin\_Base):

def \_\_init\_\_(self,rc):

MsgPlugin\_Base.\_\_init\_\_(self,'plugin\_geosightchanged\_1')

self.attrs['rect']={} # {x,y,w,h}

#MsgIpc\_MapMoved = MsgIpc\_GeoSightChanged

# mapapp -> imageclient

# 地图上选择路段，通知影像播放程序，播放程序将从路段开始位置播放影像

class MsgPlugin\_MapRoadSelected\_1(MsgPlugin\_Base):

def \_\_init\_\_(self):

MsgPlugin\_Base.\_\_init\_\_(self,'plugin\_maproadselected\_1')

self.attrs['road']={} # {first:{mess,roadid},second:{mess,roadid}}

class MsgPlugin\_ImagePathSelected\_1(MsgPlugin\_Base):

def \_\_init\_\_(self):

MsgPlugin\_Base.\_\_init\_\_(self,'plugin\_imagepathselected\_1')

self.attrs['id']=None # 轨迹编号

self.attrs['timestamp'] = 0 #图上选择路段最接近的gps时间

class MsgPlugin\_ClearImagePath\_2(MsgPlugin\_Base):

def \_\_init\_\_(self):

MsgPlugin\_Base.\_\_init\_\_(self,'plugin\_clearimagepath\_2')

self.attrs['ids']=[]

#发送到地图上的轨迹线段用一个id标识 ,ids为空则清除所有的轨迹线段

#

# imageclient -> mapapp

# 选择影像记录在图上显示出影像的轨迹

# 一旦查询影像记录返回，将在地图上显示所有的行驶轨迹

# 传递的这些连续坐标，以两点一条线段方式load进地图显示，

# 这样每条线段都可以相应用户的点击选择操作

# 这些坐标点的时间颗粒可以控制在间隔5-10秒内

class MsgPlugin\_ImagePathShowOnMap\_2(MsgPlugin\_Base):

def \_\_init\_\_(self):

MsgPlugin\_Base.\_\_init\_\_(self,'plugin\_showimagepathonmap\_2')

self.attrs['id']='' # {id,points:[{lon,lat,tick,speed,angle},...]}

self.attrs['lines']=[] #[{first:{lon,lat,speed,angle,time},second:{..}},..]

#选择影像记录，在图上高亮显示轨迹

class MsgPlugin\_ImagePathSelected\_2(MsgPlugin\_Base):

def \_\_init\_\_(self):

MsgPlugin\_Base.\_\_init\_\_(self,'plugin\_imagepathselected\_2')

self.attrs['id']=''

#图上显示车辆图标

class MsgPlugin\_ShowSymbol\_2(MsgPlugin\_Base):

def \_\_init\_\_(self):

MsgPlugin\_Base.\_\_init\_\_(self,'plugin\_showsymbol\_2')

self.attrs['show'] = True #

self.attrs['pos']={} # {lon,lat,time,speed,angle}