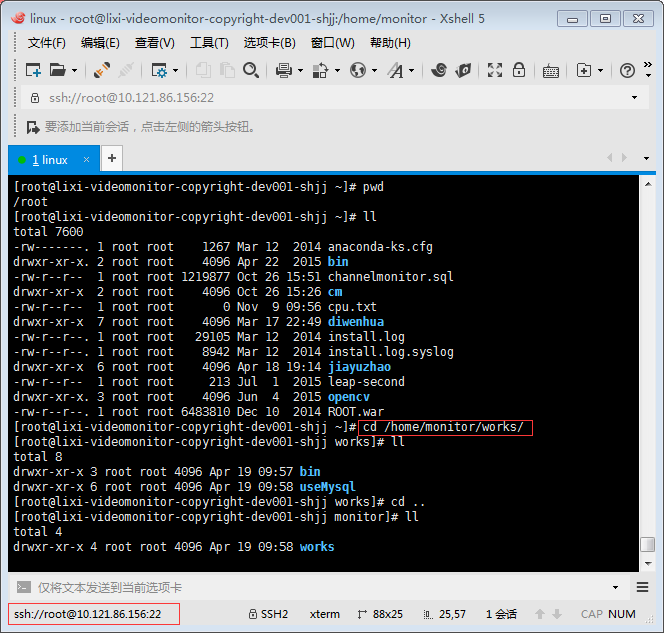
**站外版权监控系统学习总结**

1. **关于测试环境：**

主机：10.121.86.156

**1.1 工作目录：/home/monitor/works**

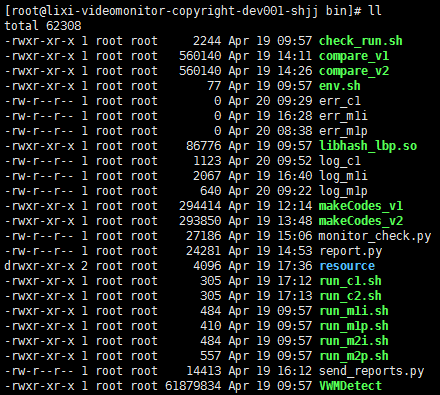


工作目录下文件的说明：

运行的应用程序在/home/monitor/works/bin里；

C++源码在/home/monitor/works/useMysql里；

* + 1. **/home/monitor/works/bin中各文件说明：**



check\_run.sh：自动运行时检测makeCodes和compare是否进行；

compare\_v1：比对表iqiyidata和pathdata中used\_flag == 1的；

compare\_v2：比对表iqiyidata和pathdata中used\_flag == 2的；

env.sh：环境配置；

err\_c1：compare\_v1错误日志；

err\_m1i：makeCodes\_v1处理表iqiyidata错误日志；

err\_m1p：makeCodes\_v1处理表pathdata错误日志；

libhash\_lbp.so：makeCodes、compare、report.py调用的动态库；

log\_c1：compare\_v1日志；

log\_m1i：makeCodes\_v1处理表iqiyidata日志；

log\_m1p：makeCodes\_v1处理表pathdata日志；

makeCodes\_v1：算特征值，表iqiyidata和pathdata中code\_flag == 0的；

makeCodes\_v2：算特征值， 表iqiyidata和pathdata中code\_flag == 0 && used\_flag == 2的；

monitor\_check.py：关键字搜索（表keyword\_search、keyword\_search\_log、keyword\_search\_report）；

report.py：过滤比对有错的（雾状图片），surf算法；

resource/：python用到的库；

run\_c1.sh：运行compare\_v1，输出到err\_c1和log\_c1；

run\_c2.sh：运行compare\_v2，输出到err\_c2和log\_c2；

run\_m1i.sh：运行makeCodes\_v1处理表iqiyidata，输出到err\_m1i和log\_m1i；

run\_m1p.sh：运行makeCodes\_v1处理表pathdata， 输出到err\_m1p和log\_m1p；

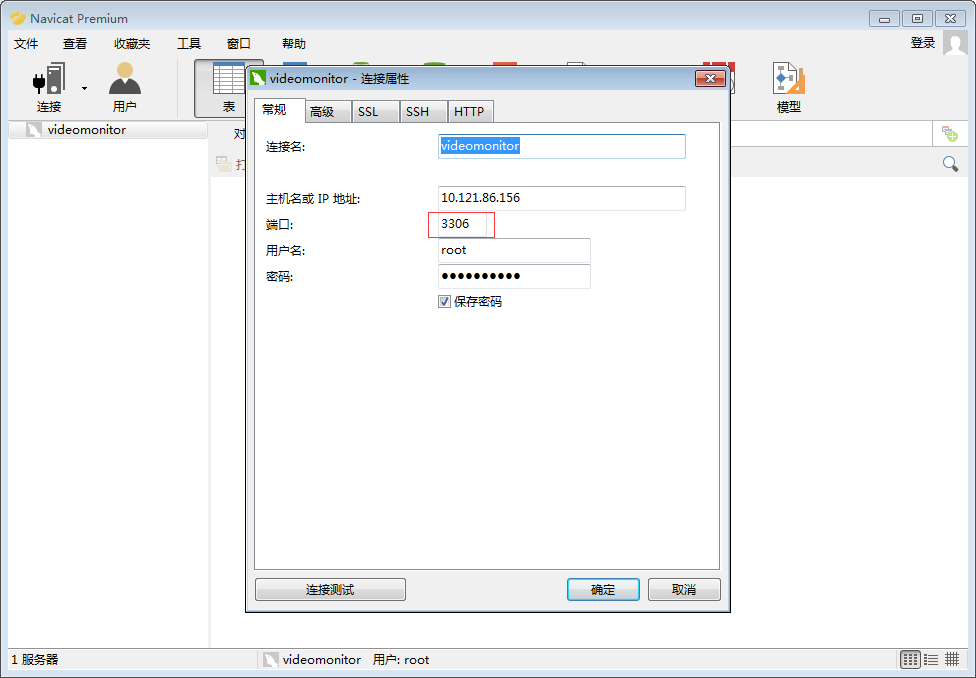
run\_m2i.sh：运行makeCodes\_v2处理表iqiyidata，输出到err\_m2i和log\_m2i；

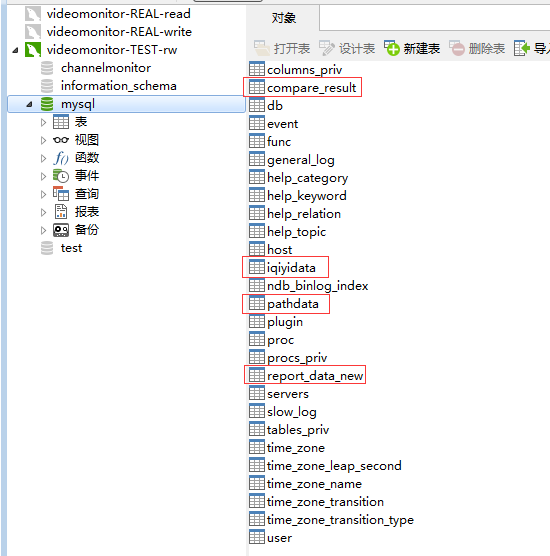
run\_m2p.sh：运行makeCodes\_v2处理表pathdata， 输出到err\_m2p和log\_m2p；

send\_reports.py：验证pathdata中url可用，检测相似度，发送邮件到邮箱；

VWMDetect：未用，检测音频水印；

**1.2 数据库：**





与视频监控系统有关的表为：iqiyidata、pathdata、compare\_result、report\_data\_new；

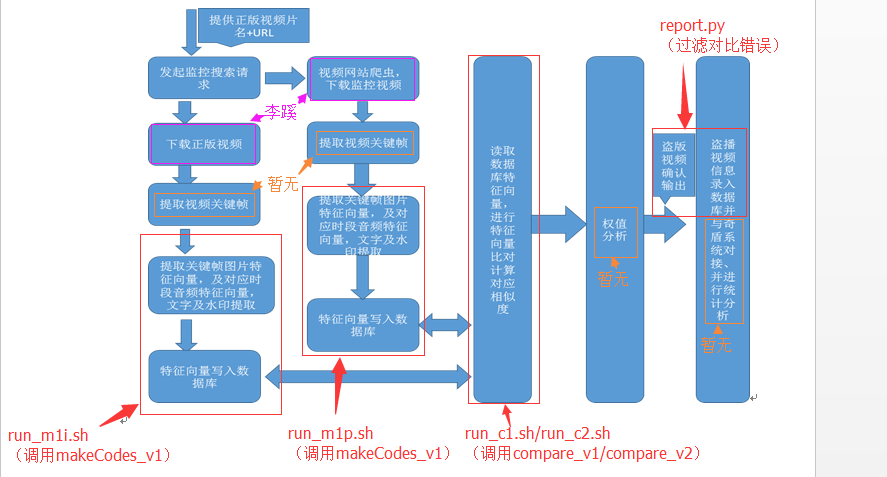
iqiyidata：存放各iqiyi的视频帧数据在计算特征值（makeCodes）之后的报告（.key）路径；

pathdata：存放各站外的视频帧数据在计算特征值（makeCodes）之后的报告（.key）路径；

compare\_result：存放iqiyi视频帧与站外视频帧比对结果中，判为相似结果报告（.csv）的路径；

report\_data\_new：存放compare\_result中，进行过滤比对错误（雾状图片）的结果；

1. **关于流程：**
   1. **系统中各模块与工作目录中文件关系：**



注：音频特征和文字及水印提取均暂无；

* 1. **运行一次系统的流程：**

iqiyi视频Hash和lbp特征值提取；

通过运行run\_m1i.sh（调makeCodes\_v1）/run\_m2i.sh（调makeCodes\_v2）；

处理数据是表iqiyidata中记录的image\_dir（/media/0mage/iqiyi/）中视频帧；

输出是视频帧的特征值记录文件（.key），保存在/media/0mage/key/iqiyi/，并在表iqiyidata中记录路径key\_path；

->

站外视频Hash和lbp特征值提取；

通过运行run\_m1p.sh（调makeCodes\_v1）/run\_m2p.sh（调makeCodes\_v2）；

处理数据是表pathdata中记录的image\_dir（/media/2mage/…/）中视频帧；

输出是视频帧的特征值记录文件（.key），保存在/media/2mage/key/…/，并在表pathdata中记录路径key\_path；

->

iqiyi和站外视频比对（比对使用hash、lbp和SSIM）出结果；

通过运行run\_c1.sh（调compare\_v1）/run\_c2.sh（调compare\_v2）；

处理数据是表iqiyi和表pathdata中记录的特征值记录文件（.key），通过表iqiyi和表pathdata记录的key\_path（/media/0mage/iqiyi/或/media/2mage/…/）找到；

输出是对比认为相似的结果记录文件（.csv），保存在/media/reports/，并在表compare\_result中记录路径report\_path；

->

过滤对比错误的结果（需要用到数据库中，记录站外视频源的表，eg：测试中用到表tudou和tudou\_long）；

通过运行report.py；

处理数据是表compare\_result中记录的结果报告文件（.csv），通过表compare\_result的report\_path找到；

输出是对比为相似的结果记录到表report\_data\_new中；

->

验证url可用，验证表report\_data\_new中sim\_N取值，发送相似报告到邮箱；

通过运行send\_report.py；

处理数据是表report\_data\_new中记录；

* 1. **三个标志flag对流程影响：**

code\_flag：0 - 未生成特征值文件

（手动改0时，删除表iqiyidata/pathdata中key\_path位置的.key）

1 - 已生成特征值文件

（文件路径：表iqiyidata/pathdata中key\_path）

used\_flag：1 - 用compare\_v1比对（针对表pathdata中新数据）

2 - 用compare\_v2比对（针对表pathdata中全部数据）

new\_flag： 0 - 不参与比对

1 - 参与比对

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 运行程序 | | iqiyidata | pathdata | compare\_result | report\_data\_new |
| run\_m1i |  | 只处理  code\_flag==0 |  |  |  |
| 后 | 处理的  code\_flag==1 |  |  |  |
| run\_m1p |  |  | 只处理  code\_flag==0 |  |  |
| 后 |  | 处理的  code\_flag==1 |  |  |
| run\_c1 |  | 只处理  code\_flag==1  &&used\_flag==1 | 只处理  code\_flag==1  &&used\_flag==1 |  |  |
| 后 |  | 处理的  used\_flag==2 |  |  |
| run\_c2 |  | 只处理  code\_flag==1  &&used\_flag==2 | 只处理  code\_flag==1  &&used\_flag==2 |  |  |
| 后 |  | 处理的  new\_flag==2 | 添加记录，  status==1 |  |
| report.py |  |  |  | 有记录，  status==1 |  |
| 后 |  |  | 处理的  status==2 | 添加记录，  status==1 |
| send\_report.py |  |  |  |  | 有记录，  status==1 |
| 后 |  |  |  | 处理的  status==2 |

1. **重要的程序说明：**

**makeCodes\_v1.cpp：**

实现视频帧的hash和lbp特征向量提取（makeCodes\_v2暂不用）；

main()创建5个线程，1个读取视频帧，4个计算特征向量；

thread\_getImageList()读取视频帧，thread\_makeCode()计算特征向量；

thread\_getImageList()读取视频帧条件是iqiyidata或pathdata表中code\_flag==0，used\_flag==1；

thread\_makeCode()核心是pHashCode()算hash值到int hCode[2]，以及getLbpHist()算lbp到float hist[59]；

程序结束时生成特征值文件.key，并更新iqiyidata或pathdata中code\_flag=1；

**Compare\_v1.cpp：**

实现iqiyidata和pathdata中记录的视频帧特征向量的比对；

main()创建5个线程，1个读取视频帧，4个比对视频帧特征向量；

thread\_getImageList()读取视频帧，thread\_compare ()比对视频帧特征向量；

thread\_getImageList()读取视频帧条件是iqiyidata和pathdata表中code\_flag==1，used\_flag==1，new\_flag==1；

thread\_compare ()比对依据是hash特征值相似度、lbp特征值相似度和SSIM相似度；流程如下：

1.计算hanming，hash特征值异或（^）；

2.hanming距离 < 5时，compareHist()计算lbp相似度；

3.lbp相似度 > LBP\_SIM时，计算SSIM相似度deepCompare()，需要原视频帧；

当SSIM相似度 > MSSIM\_SIM，生成记录到.csv；

程序结束时，若认为视频相似，生成特征值文件.csv，表compare\_result添加记录，并更新pathdata的used\_flag=2和new\_flag=1；

**Compare\_v2.cpp：**

流程与Compare\_v1.cpp一致，区别是：

thread\_getImageList()读取视频帧条件是iqiyidata和pathdata表中code\_flag==1，used\_flag==2，new\_flag==1；

程序结束时，更新pathdata的used\_flag=2和new\_flag=0；

**report.py：**

过滤compare\_result中的雾状图片；

surf\_matches()寻找雾状图片（SURF算法）；

程序只处理compare\_result中status==1的记录，程序结束时，更新compare\_result中status=2，在表report\_data\_new中添加记录；

**send\_report.py：**

表report\_data\_new中记录检验sim\_N >= 3，以及检验url可用否，若url可用，发送邮件；

程序结束时，更新report\_data\_new中status=2；