

中科院声学所离线系统场景分割服务端

开发测试环境说明

版 本 历 史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 起止日期 | 备注 |
| V1.0 | 王超锋 |  | 2021-11-17 | 创建文件 |
|  |  |  |  |  |

目录

[1.编译环境 3](#_Toc742)

[1.1硬件环境 3](#_Toc2699)

[1.2软件环境 3](#_Toc8978)

[1.3编译方式 4](#_Toc16331)

[1.4输出结果 4](#_Toc6304)

[2.测试环境部署 5](#_Toc6991)

[2.1 系统部署 5](#_Toc23849)

[2.2 运行环境 6](#_Toc9622)

[3.系统测试 8](#_Toc8016)

[3.1测试工具 8](#_Toc592)

[3.2测试脚本 8](#_Toc19765)

[3.3测试数据 8](#_Toc9095)

[3.4测试方法 8](#_Toc893)

[3.4.1 C++方式 9](#_Toc25352)

[3.4.2 Python方式 10](#_Toc3477)

[3.5测试结果 10](#_Toc7186)

# 1.编译环境

1

**1.1硬件环境**

|  |  |  |
| --- | --- | --- |
| 分类 | 最低配置 | 推荐配置 |
| CPU | 4C | 16 |
| 内存 | 8G | 32G |
| 磁盘 | 64G | 128G |

## 1.2软件环境

|  |  |  |
| --- | --- | --- |
| 分类 | 名称 | 版本 |
| 操作系统 | Centos | 7.x |
| 数据库 | Redis | 3.2.0 |
| Mysql | 5.7.30 |
| 能力接口 | Tomcat | 9.0.35 |
| GCC | GCC | 4.8.5 |
| G++ | G++ | 4.8.5 |
| GUN | make | 3.82 |
| 总控端源代码 | OfflineSystemClientSrc | / |

## 1.3编译方式

编译

cd ClusterServerSrc/

make

## 1.4输出结果

编译输出

Cluster\_Scene\_server\_test 场景分割服务端工程文件

# 2.测试环境部署

2

## 2.1 系统部署

1、解压部署包

2、配置数据库信息到db\_user\_pwd

3、配置环境变量bash\_global

4、安装应用和依赖包：./SystemStartup.sh install

5、申请替换秘钥

6、启动系统：./SystemStartup.sh

7、检查系统启动是否正常

## 2.2 运行环境

硬件环境

|  |  |  |
| --- | --- | --- |
| 分类 | 最低配置 | 推荐配置 |
| CPU(支持avx、sse4指令集) | 4C | 64C |
| 内存 | 16 | 64G |
| 磁盘 | 128G | 1T |

软件环境

|  |  |  |
| --- | --- | --- |
| 分类 | 名称 | 版本 |
| 操作系统 | Centos | 7.x |
| 数据库 | Redis | 3.2.0 |
| Mysql | 5.7.30 |
| 能力接口 | Tomcat | 9.0.35 |
| GCC | GCC | 4.8.5 |
| G++ | G++ | 4.8.5 |
| GUN | make | 3.82 |
| JDK | jdk1.8.0\_171 | 1.8.0\_171 |
| Python | Python | 2.7 |

# 3.系统测试

3

## 3.1测试工具

c++测试工具：demo\_3.1.tar.gz

## 3.2测试脚本

Python测试脚本：test\_thinkit\_offlineasr\_list\_json.py

## 3.3测试数据

8k16bitwav,单声道mp3语音，双声道mp3语音。

## 3.4测试方法

### 3.4.1 C++方式

1、配置test.sh文件中

#发送线程数

ThreadNum=20

#待处理语音路径

voicedir=/home/thinkit/test\_tools/demo/voice

#URL

Url=http://127.0.0.1:30100/offlineasr

Delvoice=false

channel\_flag=true

#声道数配置

channels=0

stereo\_on=3

type=1

#获取结果文件类型

postfix=.xml

1. 待测试语音放入voice目录
2. 执行./test.sh

### 3.4.2 Python方式

1. 配置test\_thinkit\_offlineasr\_list\_json.py

SERVER\_IP = "127.0.0.1"

SERVER\_PORT = 30100

SERVER\_URI = '/offlineasr'

kFiledir = '/home/thinkit/python\_demo/voice'

kchannels = 0

kstereo\_on = 3

kThreadNum = 1 # 线程数

kType = 1 # 使用第几个集群

kpostfix = 'sent.txt' #获取结果类型

kOnePacketLen = 16000 # 每包大小，最好设置为1600

gGrstimeOut = 3600 #超时时间

1. 待测试语音放入voice目录
2. 执行python test\_thinkit\_offlineasr\_list\_json.py

## 3.5测试结果

使用c++和python测试工具测试不同语音，均能正常转写，转写结果正常。测试期间，系统正常运行，没有出现宕机、崩溃现象。