**基于Lucene的分析与应用**

**测试报告**

Version 1.0

小组成员：

刘宏宇

滕延林

顾泽鹏

杨帆

周晓懿

**版本变更记录**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 变更时间 | 修改人 | 审核人 |
| 1.0 | 2016/05/26 | 顾泽鹏 滕延林 刘宏宇 | 杨帆 周晓懿 |
| 1.01 | 2016/06/20 | 滕延林 | 顾泽鹏 刘宏宇杨帆 周晓懿 |

目录

[1. 编写说明 4](#_Toc452069502)

[1.1标识 4](#_Toc452069503)

[1.2编写目的 4](#_Toc452069504)

[2. 实际测试用例对应表 4](#_Toc452069505)

[3. 测试用例 5](#_Toc452069506)

[3.1系统部署测试 5](#_Toc452069507)

[3.2文件添加测试 5](#_Toc452069508)

[3.3文件删除测试 6](#_Toc452069509)

[3.4文件修改测试 8](#_Toc452069510)

[3.5索引更新测试 9](#_Toc452069511)

[3.6文本检索测试 10](#_Toc452069512)

[3.7图像特征选择测试 11](#_Toc452069513)

[3.8数据预处理测试 12](#_Toc452069514)

[3.9距离选择测试 13](#_Toc452069515)

[3.10网站搭建测试 14](#_Toc452069516)

[3.11Hash排序测试 15](#_Toc452069517)

[3.12图像检索测试 16](#_Toc452069518)

[3.13兼容性测试 17](#_Toc452069519)

# 编写说明

## 1.1标识

文档标题：基于Lucene的分析与应用测试报告

版本号：V1

## 1.2编写目的

根据需求文档中的用例，对于用例进行测试，保证实际产品与需求相对应，使产品满足需求规格。在测试中发现并修改问题，保证产品的正确性。

# 实际测试用例对应表

本测试报告中已经完成的测试用例与计划的测试用例及需求用例的对应关系。本报告完成了部分重要用例的测试，包括系统部署测试文件添加测试、文件删除测试、文件修改测试、文件更新测试、索引更新测试、文本检索测试、图像特征选择测试、数据预处理测试、距离选择测试、网站搭建测试、Hash排序测试、图像检索测试、兼容性测试、性能测试等，后续将会完善所有的测试用例。

|  |  |  |  |
| --- | --- | --- | --- |
| 模块 | 需求用例 | 测试用例 | 完成情况 |
| 基本需求 | 系统部署 | 系统部署测试 | 完成 |
| 文本库管理 | 文件添加测试；  文件删除测试；  文件修改测试；  文件更新测试； | 完成 |
| 索引更新 | 索引更新测试 | 完成 |
| 文本检索 | 文本检索测试 | 完成 |
| 实现需求 | 图像特征选择 | 图像特征选择测试 | 完成 |
| 数据预处理 | 数据预处理测试 | 完成 |
| 距离选择 | 距离选择测试 | 完成 |
| 网站搭建 | 网站搭建测试 | 完成 |
| Hash排序 | Hash排序测试 | 完成 |
| 图像检索 | 图像检索测试 | 完成 |
| 非功能需求 | 兼容性 | 兼容性测试 | 部分完成 |
| 高效性 | 性能测试 | 部分完成 |

# 测试用例

## 3.1系统部署测试

**（1）测试目标**

Lucene可以正常导入。

**（2）测试过程**

将Lucene导入eclipse

**（3）测试结果**

导入正常

**（4）结果分析**

系统可以正常部署

## 3.2文件添加测试

**（1）测试目标**

保证系统可以正常入库，并且入库结果可以影响检索效果，即添加的文件可以被检索到。

**（2）测试过程**

1. 打开页面，选择一张目标图像，进入图像检索页面，进行图像检索
2. 获取并记录检索结果
3. 选择一张与目标图像比较像的图像，将该图像入库
4. 重新检索，比较两次检索的差异

**（3）测试结果**



Figure 1文件添加前检索结果



Figure 2文件添加后检索结果

**（4）结果分析**

在Figure 2中，第5张为新添加的图片。可以看到，文件添加前，处于第5张位置的图片被挤到了第6张。文件添加过程正确，并且可以被检索到。

## 3.3文件删除测试

**（1）测试目标**

保证库中的文件可以正常删除，并且删除结果可以影响检索结果，即删除后的文件不会被检索到。

**（2）测试过程**

1. 打开页面，选择一张目标图片，进行检索。
2. 得到并记录检索结果。
3. 在选择一张检索结果的图片，并删除。
4. 重新检索，比较两次结果。

**（3）测试结果**

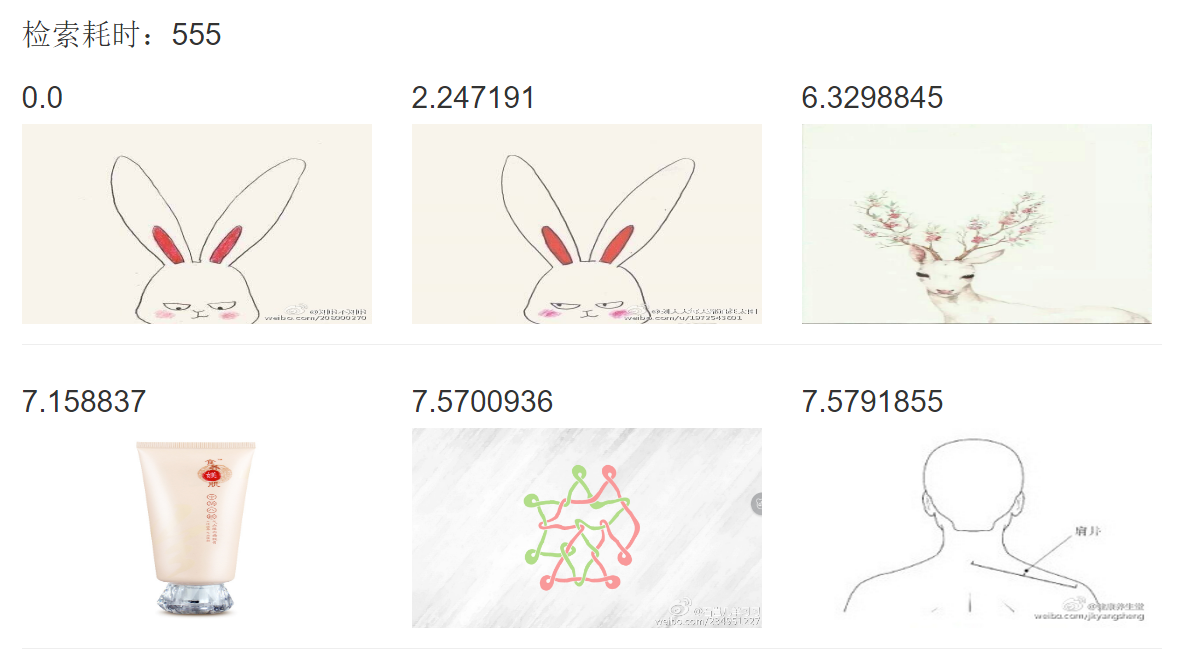


Figure 3文件删除前检索结果

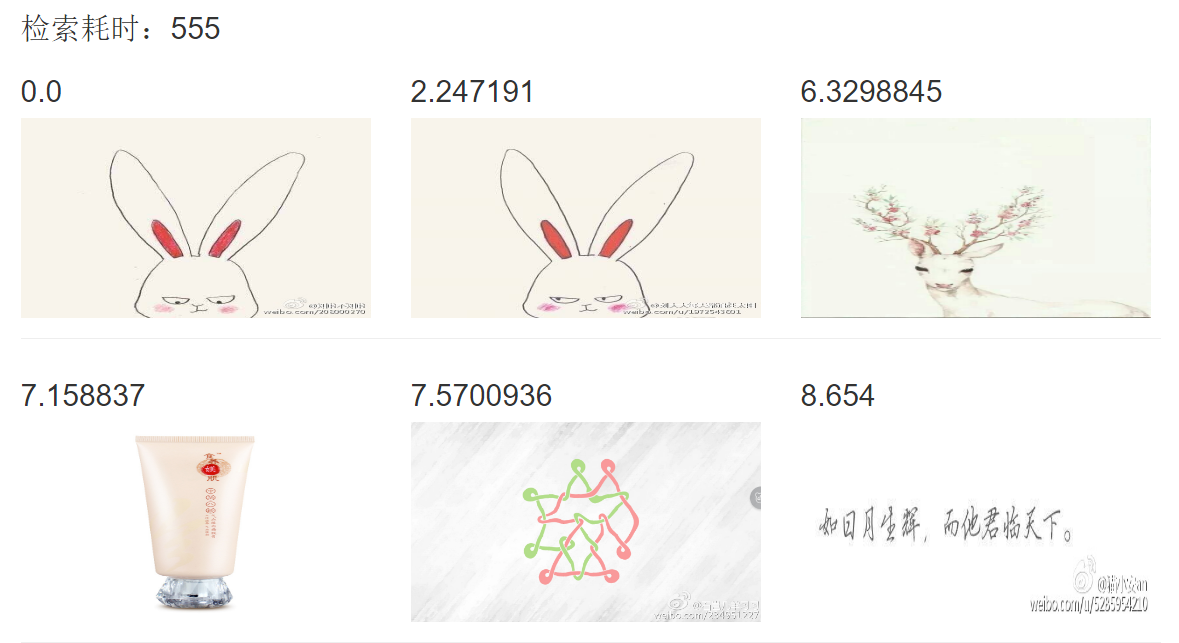


Figure 4文件删除后检索结果

**（4）结果分析**

Figure 3中第6张的图片为删除的文件。可以看到删除后，该图片不会被检索到。原来处于第7位置的图片，提到了第6位。文件删除功能正常。

## 3.4文件修改测试

**（1）测试目标**

测试库中的文件是否能被修改，并且修改结果可以在检索界面看到。

**（2）测试过程**

1. 在检索界面输入关键字，进行文本检索。
2. 修改文件内容。
3. 再次检索，比较两次检索差异。

**（3）测试结果**

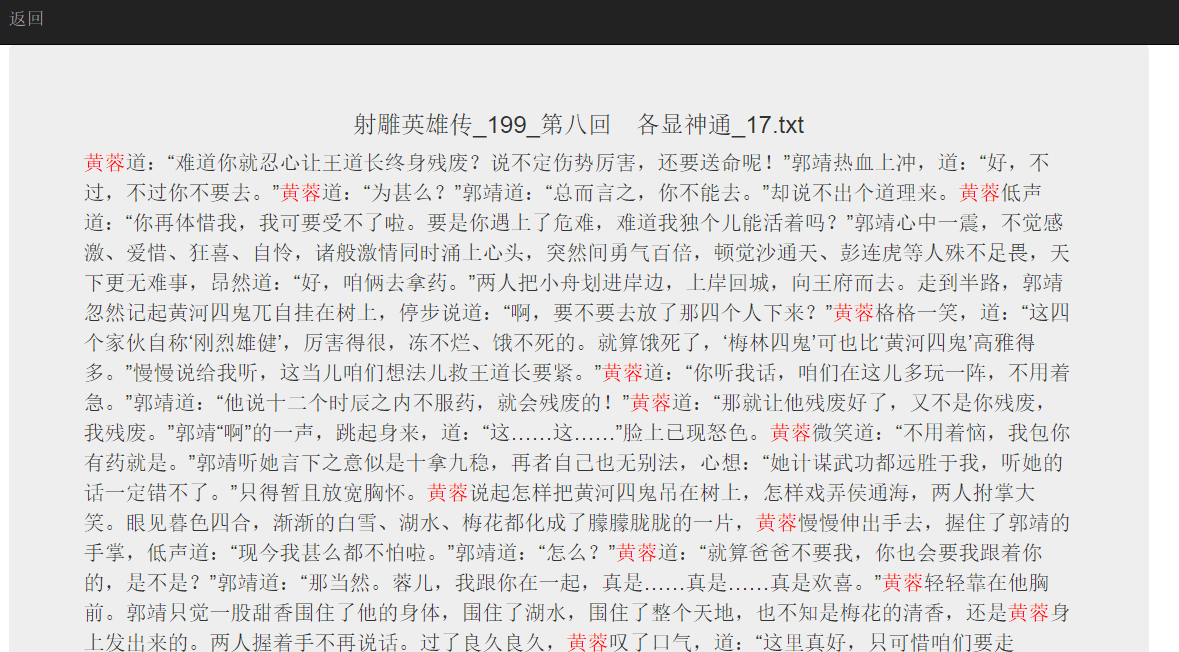


Figure5文件修改前检索结果

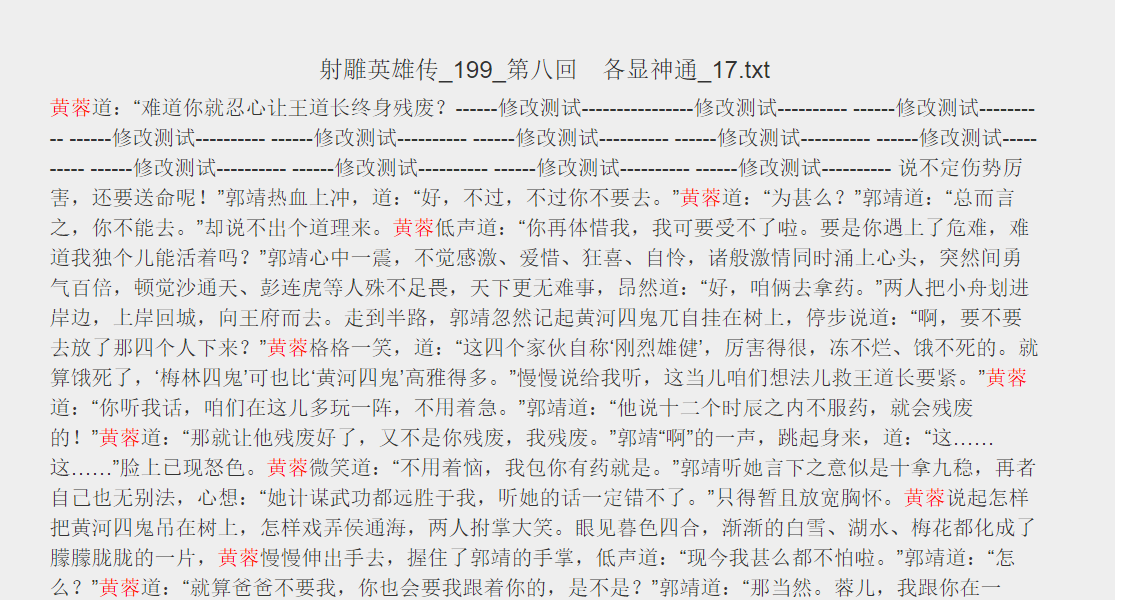


Figure 6文件修改后检索结果

**（4）结果分析**

可以看到，两次检索得到的章节一样，说明两次检索到的是同一个文件。只不过内容有了微小修改。说明文件修改功能正常。

## 3.5索引更新测试

**（1）测试目标**

测试文件系统更新后，索引十分能够正常更新，保证检索的正确性。

**（2）测试过程**

1. 查看索引目录。
2. 更新文件系统。
3. 再次查看索引目录，比较差异。

**（3）测试结果**

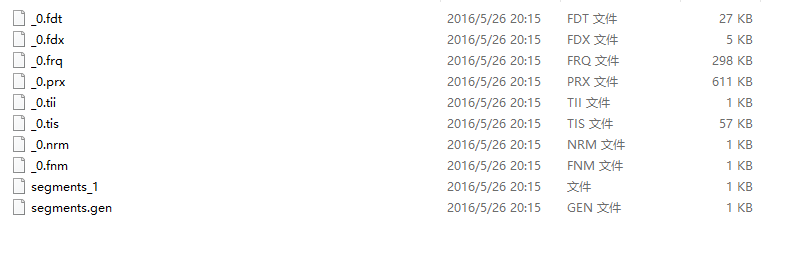


Figure 7索引更新前

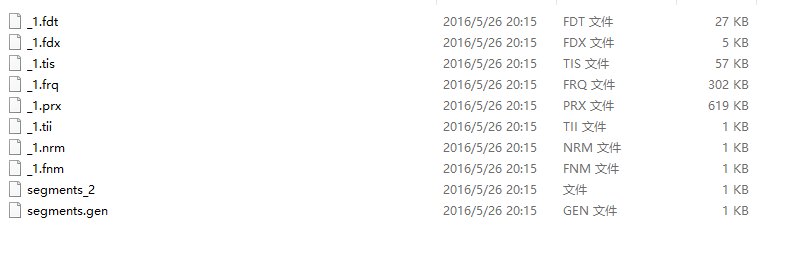


Figure 8索引更新后

**（4）结果分析**

可以看到，文件系统更新后，索引有了变化。另外，文件的添加、删除、修改的测试结果，可以从侧门验证索引更新功能正常。

## 3.6文本检索测试

（1）测试目标

保证文本可以正常检索

（2）测试过程

1. 打开页面<http://buaatyl.cn/lucenceWeb/textSearch.jsp>
2. 输入关键字，进行检索
3. 查看检索结果十分符合预期

（3）测试结果

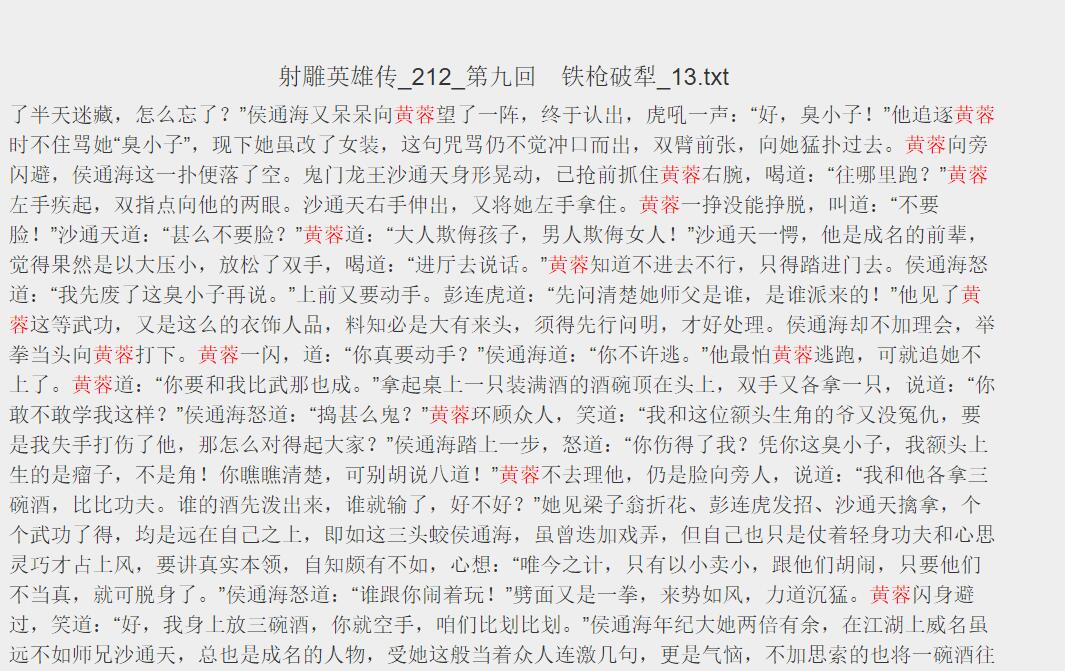


Figure 9文本检索结果

（4）结果分析

可以看到，输入关键字“黄蓉”，得到射雕英雄传的章节，符合预计结果，并且关键字可以高亮显示。说明文本检索功能正常。

## 3.7图像特征选择测试

（1）测试目标

比较不同特征对于图像检索的影响，寻找最合适的特征。

（2）测试过程

1. 对于同样的图片，提取不同的特征。
2. 比较检索结果，统计准确率、

（3）测试结果

Figure 10 不同特征准确率比较

（4）结果分析

从图中结果可以看到，CEDD特征的表现最好，在系统中选择CEDD作为图像检索的特征。

## 3.8数据预处理测试

（1）测试目标

测试数据预处理，使数据具有良好的格式。

（2）测试过程

1. 记录原始数据。
2. 将数据预处理后，比较结果。

（3）测试结果



Figure 11数据预处理前



Figure12数据预处理后

（4）结果分析

从结果可以看出，数据预处理前，文件比较大，不利于检索。数据预处理后将文件切分成小的部分，便于检索。并且以章节名称命名。

## 3.9距离选择测试

（1）测试目标

比较不同距离计算方式对于图像检索的影响，寻找最合适的距离

欧氏距离

余弦距离

Tanimoto距离（广义Jaccard距离）

（2）测试过程

1. 选定一种特征，这里选择CEDD特征。
2. 根据不同距离计算相似度，并比较检索结果，统计准确率

（3）测试结果

Figure 13不同特征准确率比较

（4）结果分析

经过多组的实验表明，欧氏距离的效果最差，但是花费的时间最少。余弦距离的效果优于欧氏距离而差于Tanimoto距离，花费的时间比欧氏距离略多。Tanimoto距离是效果最好的，但是计算Tanimoto距离消耗的时间最多。经过权衡，最终选择Tanimoto距离来衡量点之间的距离

## 3.10网站搭建测试

**（1）测试目标**

保证系统可以在web上正常部署，且可以正常访问。

**（2）测试过程**

1. 打开网页
2. 输入网址<http://buaatyl.cn/lucenceWeb/imageSearch.jsp>以及<http://buaatyl.cn/lucenceWeb/textSearch.jsp>
3. 查看网页结果

**（3）测试结果**



Figure 14系统部署测试

**（4）结果分析**

页面显示正常，网站搭建成功。但仅说明系统可以运行，并不能说明系统逻辑正确。

## 3.11Hash排序测试

（1）测试目标

测试hash函数的正确性以及效果。即在对结果影响很小的前提下，提高效率。

（2）测试过程

1. 在检索界面，选择一张图像使用非hash方式检索，记录检索结果以及时间。
2. 选择同一张图像使非hash方式检索，记录检索结果以及时间。
3. 比较两者的时间差异以及效果。

（3）测试结果

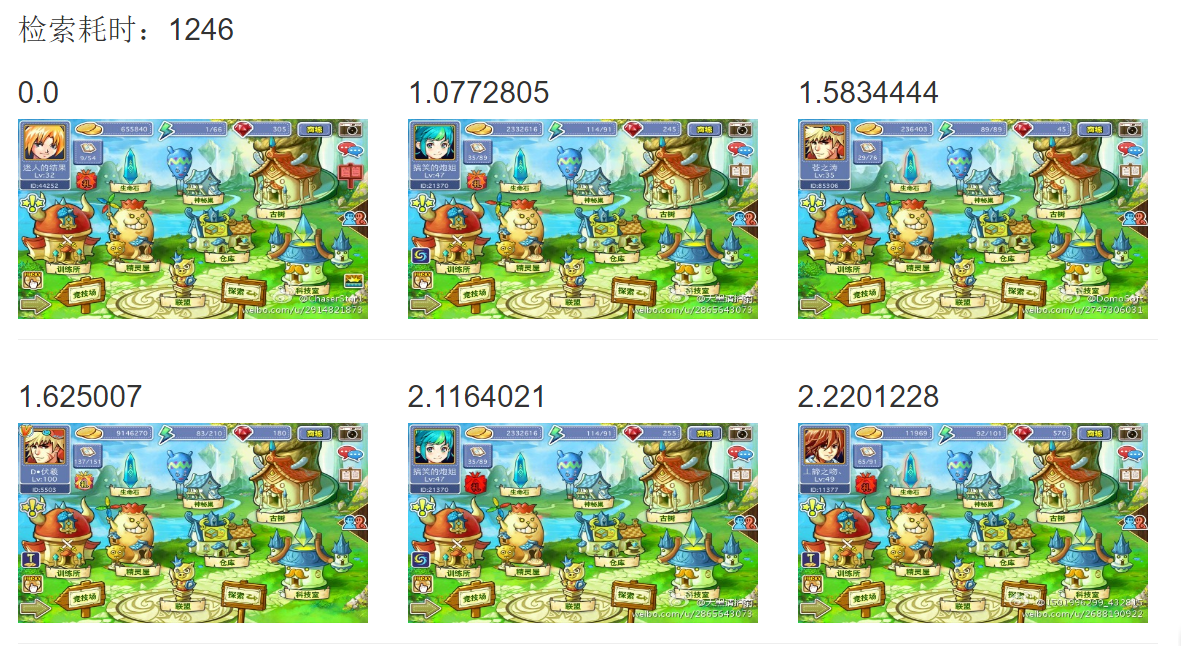


Figure 15未使用hash检索结果

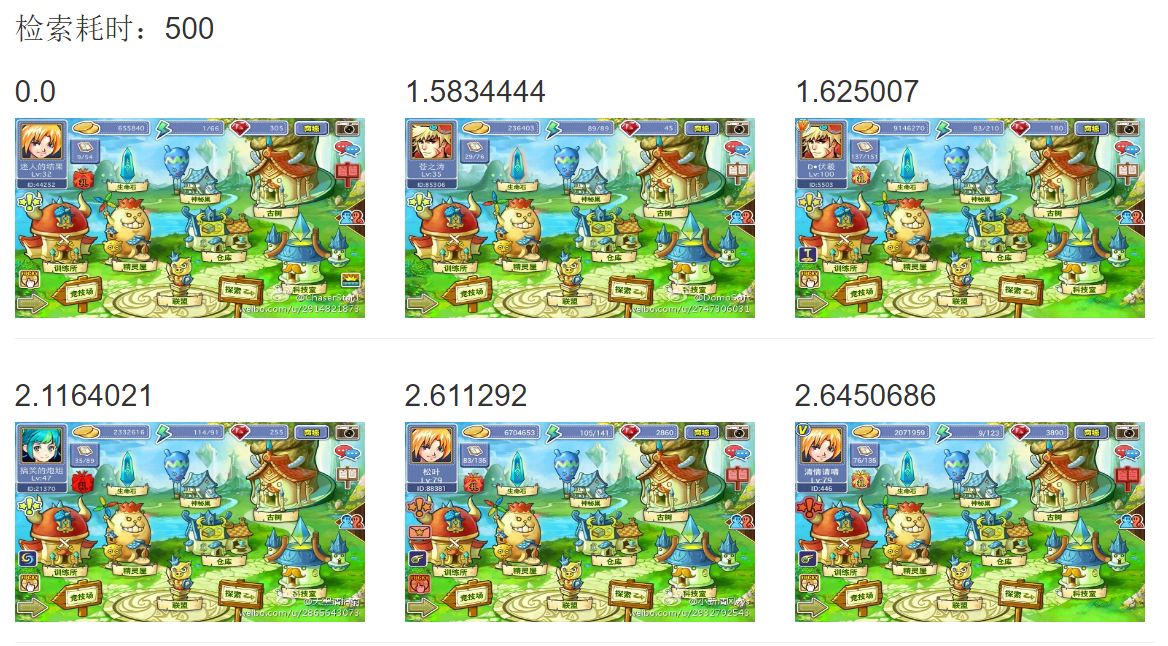


Figure 16使用hash检索效果

（4）结果分析

使用hash后对于检索效果影响不大但时间从1246ms减小到500ms，效率提高一倍以上。说明hash函数正确，并且对于效率改善效果明显。

## 3.12图像检索测试

（1）测试目标

测试图像检索是否正常，给出结果十分符合预期。

（2）测试过程

1. 打开页面，选择一张目标图片进行检索。
2. 得到检索结果。
3. 与目标图片比较。

（3）测试结果

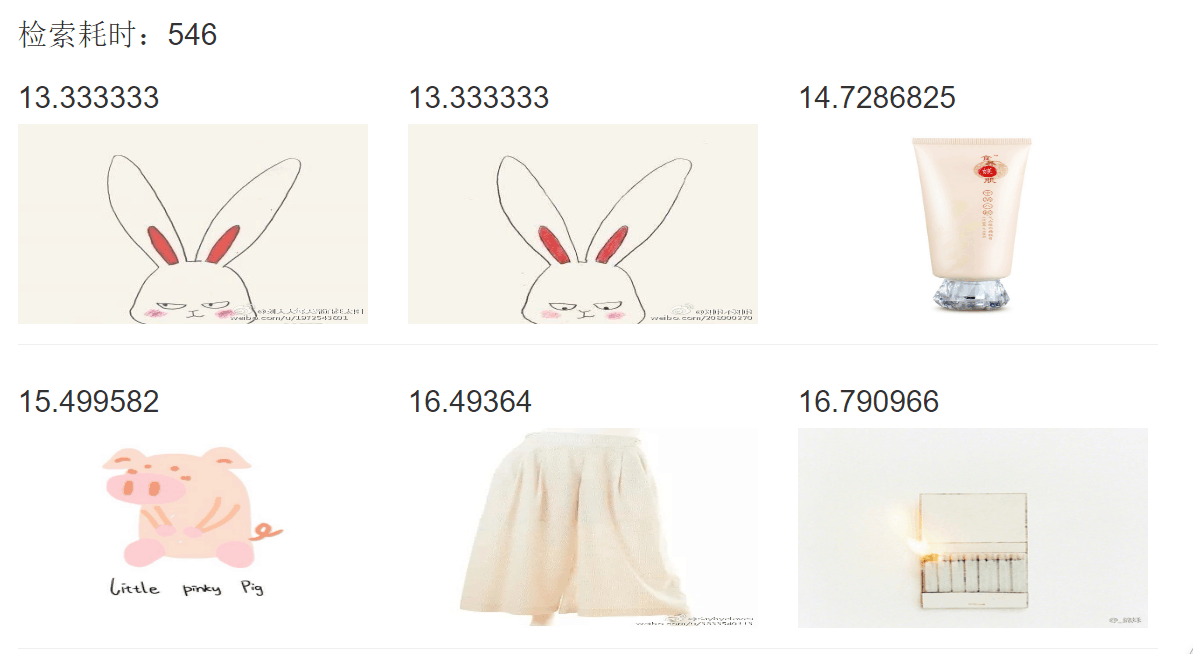


Figure 17 图像检索效果

（4）结果分析

图片检索能够正常进行，并且结果比较符合人们需求。图片检索功能正常。

## 3.13兼容性测试

（1）测试目标

测试系统能否在不同平台下运行。

（2）测试过程

1. 首先在电脑上打开，测试在Windows下效果。
2. 然后再手机上打开，测试在Android下效果

（3）测试结果



Figure 18兼容性测试

（4）结果分析

系统可以在不同平台下正常运行。兼容性良好。

## 3.14效率测试

（1）测试目标

分别测试hash前和hash后的检索效率。

（2）测试过程

1. 准备20张图片。
2. 在检索界面，对20张图像分别使用非hash方式检索，记录检索结果以及时间，并计算平均时间。
3. 对20张图像使用hash方式检索，记录检索结果以及时间，并计算平均检索时间。
4. 对两种检索方式进行比较。

（3）测试结果

|  |  |  |
| --- | --- | --- |
| **图片编号** | **Hash(ms)** | **非hash(ms)** |
| 1 | 499 | 1023 |
| 2 | 654 | 1197 |
| 3 | 532 | 998 |
| 4 | 432 | 892 |
| 5 | 754 | 1233 |
| 6 | 456 | 900 |
| 7 | 804 | 1429 |
| 8 | 302 | 500 |
| 9 | 331 | 600 |
| 10 | 485 | 993 |
| 11 | 528 | 1211 |
| 12 | 520 | 1212 |
| 13 | 559 | 1032 |
| 14 | 732 | 1523 |
| 15 | 231 | 594 |
| 16 | 563 | 1252 |
| 17 | 505 | 1002 |
| 18 | 523 | 1130 |
| 19 | 612 | 1231 |
| 20 | 225 | 400 |
| 平均 | 512.35 | 1017.6 |