LIRe 源代码分析 4:建立索引(DocumentBuilder)[以颜色布局为例]

2013年11月02日 16:45:49 阅读数:5222

LIRe源代码分析系列文章列表:

LIRe 源代码分析 1:整体结构

LIRe 源代码分析 2:基本接口(DocumentBuilder)

LIRe 源代码分析 3:基本接口(ImageSearcher)

LIRe 源代码分析 4:建立索引(DocumentBuilder)[以颜色布局为例]

LIRe 源代码分析 5:提取特征向量[以颜色布局为例]

LIRe 源代码分析 6:检索(ImageSearcher)[以颜色布局为例]

LIRe 源代码分析 7:算法类[以颜色布局为例]

前几篇文章介绍了LIRe 的基本接口。现在来看一看它的实现部分,本文先来看一看建立索引((DocumentBuilder))部分。不同的特征向量提取方法的建立索引的类各不相同,它们都位于"net.semanticmetadata.lire.impl"中,如下图所示:

▲ ⊕ net.semanticmetadata.lire.impl

▶ 🔐 BasicDocumentBuilder.class

Danie CEDDDocumentBuilder.class

CEDDImageSearcher.class

▶ the ChainedDocumentBuilder.class

ColorLayoutDocumentBuilder.class

ColorLayoutImageSearcher.class

CorrelogramDocumentBuilder.class

Danie CorrelogramImageSearcher.class

DocumentFactory.class

▶ GenericDocumentBuilder.class

GenericFastDocumentBuilder.class

→ GenericFastImageSearcher.class
 → GenericImageSearcher.class

MSERDocumentBuilder.class

ParallelImageSearcher.class

▶ SimpleDocumentBuilder.class

▶ SimpleImageDuplicates.class

▶ SimpleImageSearcher.class

▶ SimpleImageSearchHits.class

▷ SimpleResult.class

▶ In part of the part of t

VisualWordsImageSearcher.class / 1eixiaohua1020

▶ # net.semanticmetadata.lire.indexing

由图可见,每一种方法对应一个DocumentBuilder和一个ImageSearcher,类的数量非常的多,无法一一分析。在这里仅分析一个比较有代表性的 :颜色布局。

颜色直方图建立索引的类的名称是ColorLayoutDocumentBuilder,该类继承了AbstractDocumentBuilder,它的源代码如下所示:

```
[java] 📳 📑
 2.
      * This file is part of the LIRe project: http://www.semanticmetadata.net/lire
       * LIRe is free software; you can redistribute it and/or modify
3.
      * it under the terms of the GNU General Public License as published by
 4.
 5.
       * the Free Software Foundation; either version 2 of the License, or
6.
      * (at your option) any later version.
7.
      * LIRe is distributed in the hope that it will be useful,
8.
       * but WITHOUT ANY WARRANTY; without even the implied warranty of
9.
      * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
10.
11.
       * GNU General Public License for more details.
12.
13.
       * You should have received a copy of the GNU General Public License
14.
      * along with LIRe; if not, write to the Free Software
15.
       * Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
16.
17.
       * We kindly ask you to refer the following paper in any publication mentioning Lire:
18.
19.
       * Lux Mathias, Savvas A. Chatzichristofis. Lire: Lucene Image Retrieval 欽@
      * An Extensible Java CBIR Library. In proceedings of the 16th ACM International
20.
21.
       * Conference on Multimedia, pp. 1085-1088, Vancouver, Canada, 2008
22.
23.
       * http://doi.acm.org/10.1145/1459359.1459577
24.
       * Copyright statement:
25.
26.
       * (c) 2002-2011 by Mathias Lux (mathias@juggle.at)
27.
28.
      * http://www.semanticmetadata.net/lire
29.
30.
      package net.semanticmetadata.lire.impl;
31.
32.
      import net.semanticmetadata.lire.AbstractDocumentBuilder;
33.
      import net.semanticmetadata.lire.DocumentBuilder;
34.
      import net.semanticmetadata.lire.imageanalysis.ColorLayout;
35.
      import net.semanticmetadata.lire.utils.ImageUtils;
36.
      import org.apache.lucene.document.Document;
37.
      import org.apache.lucene.document.Field;
38.
39.
      import iava.awt.image.BufferedImage:
40.
      import java.util.logging.Logger;
41.
42.
43.
       st Provides a faster way of searching based on byte arrays instead of Strings. The method
44.
      * {@link net.semanticmetadata.lire.imageanalysis.ColorLayout#getByteArrayRepresentation()} is used
45.
       * to generate the signature of the descriptor much faster.
      * User: Mathias Lux, mathias@juggle.at
46.
47.
       * Date: 30.06.2011
48.
49.
      public class ColorLayoutDocumentBuilder extends AbstractDocumentBuilder {
50.
       private Logger logger = Logger.getLogger(getClass().getName());
51.
          public static final int MAX IMAGE DIMENSION = 1024;
52.
53.
          public Document createDocument(BufferedImage image, String identifier) {
54.
             assert (image != null):
55.
              BufferedImage bimg = image;
             // Scaling image is especially with the correlogram features very important!
56.
57.
              // All images are scaled to guarantee a certain upper limit for indexing.
58.
              if (Math.max(image.getHeight(), image.getWidth()) > MAX_IMAGE_DIMENSION) {
59.
                  bimg = ImageUtils.scaleImage(image, MAX_IMAGE_DIMENSION);
60.
61.
              Document doc = null;
62.
              logger.finer("Starting extraction from image [ColorLayout - fast].");
              ColorLayout vd = new ColorLayout();
63.
64.
              vd.extract(bimg);
65.
              logger.fine("Extraction finished [ColorLayout - fast].");
66.
67.
              doc = new Document();
              doc.add(new Field(DocumentBuilder.FIELD_NAME_COLORLAYOUT_FAST, vd.getByteArrayRepresentation()));
68.
              if (identifier != null)
69.
                  doc.add(new Field(DocumentBuilder.FIELD_NAME_IDENTIFIER, identifier, Field.Store.YES, Field.Index.NOT_ANALYZED));
70.
71.
72.
              return doc;
73.
74.
```

从源代码来看,其实主要就一个函数:createDocument(BufferedImage image, String identifier),该函数的流程如下所示:

- 1.如果输入的图像分辨率过大(在这里是大于1024),则将图像缩小。
- 2.新建一个ColorLayout类型的对象vd。
- 3.调用vd.extract()提取特征向量。
- 4.调用vd.getByteArrayRepresentation()获得特征向量。
- 5.将获得的特征向量加入Document,返回Document。

```
[java] 📳 📑
1.
      * This file is part of the LIRe project: http://www.semanticmetadata.net/lire
2.
       * LIRe is free software; you can redistribute it and/or modify
3.
      * it under the terms of the GNU General Public License as published by
4.
 5.
       st the Free Software Foundation; either version 2 of the License, or
6.
      * (at your option) any later version.
      * LIRe is distributed in the hope that it will be useful,
8.
 9.
       * but WITHOUT ANY WARRANTY; without even the implied warranty of
      * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
10.
11.
       * GNU General Public License for more details.
12.
13.
       * You should have received a copy of the GNU General Public License
       * along with LIRe; if not, write to the Free Software
14.
       * Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
15.
16.
       * We kindly ask you to refer the following paper in any publication mentioning Lire:
17.
18.
       * Lux Mathias, Savvas A. Chatzichristofis. Lire: Lucene Image Retrieval 鈥@
19.
20.
      * An Extensible Java CBIR Library. In proceedings of the 16th ACM International
21.
       * Conference on Multimedia, pp. 1085-1088, Vancouver, Canada, 2008
22.
23.
       * http://doi.acm.org/10.1145/1459359.1459577
24.
25.
       * Copyright statement:
26.
27.
       * (c) 2002-2011 by Mathias Lux (mathias@juggle.at)
      * http://www.semanticmetadata.net/lire
28.
29.
      package net.semanticmetadata.lire.impl:
30.
31.
32.
      import net.semanticmetadata.lire.AbstractDocumentBuilder;
33.
      import net.semanticmetadata.lire.DocumentBuilder;
34.
      import net.semanticmetadata.lire.imageanalysis.CEDD;
35.
      import net.semanticmetadata.lire.utils.ImageUtils;
      import org.apache.lucene.document.Document;
36.
37.
      import org.apache.lucene.document.Field;
38.
39.
      import java.awt.image.BufferedImage;
40.
      import java.util.logging.Logger;
41.
42.
43.
       * Provides a faster way of searching based on byte arrays instead of Strings. The method
      * {@link net.semanticmetadata.lire.imaqeanalysis.CEDD#qetByteArrayRepresentation()} is used
44.
       * to generate the signature of the descriptor much faster.
45.
      * User: Mathias Lux, mathias@juggle.at
46.
47.
       * Date: 12.03.2010
      * Time: 13:21:35
48.
49.
50.
      * @see GenericFastDocumentBuilder
51.
       st @deprecated use GenericFastDocumentBuilder instead.
52.
53.
      public class CEDDDocumentBuilder extends AbstractDocumentBuilder {
      private Logger logger = Logger.getLogger(getClass().getName());
54.
          public static final int MAX_IMAGE_DIMENSION = 1024;
55.
56.
57.
          public Document createDocument(BufferedImage image, String identifier) {
              assert (image != null):
58.
59.
              BufferedImage bimg = image;
              // Scaling image is especially with the correlogram features very important!
60.
61.
              // All images are scaled to guarantee a certain upper limit for indexing.
62.
             if (Math.max(image.getHeight(), image.getWidth()) > MAX_IMAGE_DIMENSION) {
63.
                  bimg = ImageUtils.scaleImage(image, MAX_IMAGE_DIMENSION);
64.
65.
              Document doc = null;
              logger.finer("Starting extraction from image [CEDD - fast].");
66.
67.
              CEDD \ vd = new \ CEDD();
68.
              vd.extract(bimg):
69.
              logger.fine("Extraction finished [CEDD - fast].");
70.
71.
              doc = new Document();
              doc.add(new Field(DocumentBuilder.FIELD NAME CEDD, vd.getByteArrayRepresentation()));
72.
              if (identifier != null)
73.
              doc.add(new Field(DocumentBuilder.FIELD_NAME_IDENTIFIER, identifier, Field.Store.YES, Field.Index.NOT_ANALYZED));
74.
75.
76.
              return doc;
77.
          }
78.
```

版权声明:本文为博主原创文章,未经博主允许不得转载。 https://blog.csdn.net/leixiaohua1020/article/details/13774637

文章标签: lire 源代码 索引 检索 lucene

个人分类:LIRe MPEG7/图像检索 所属专栏:开源多媒体项目源代码分析

此PDF由spygg生成,请尊重原作者版权!!!

我的邮箱:liushidc@163.com