

基于 mybatis 的面向数据库自动生成技术

黄艳秀

(河南省人才交流中心,河南 郑州 450003)

摘要:基于在项目开发中减少代码对数据库语言依赖性的目的,采用 mybatis 框架,通过自动生成类、接口和代码,结合 Spring 翻转注入的工厂方法,完成了对程序员的解放,降低了代码耦合度,增加项目的可移植性。

关键词:持久;映射;数据库连接;接口;耦合度;封装

中图分类号:TP311 **文献标识码:**A **文章编号:**1003-5168(2014)04-0021-02

在以 JAVA 语言为基础的面向对象设计的项目中,肯定需要对数据库进行连接。而 JAVA 中对对象类都需要在数据库中有相对应的表,所以就需去编写每一个类的每一个属性的数据库语句。但是在项目建设过程中会经常对对象类或者数据库中的表进行修改,那么每一次修改都需要再去修改相对应的数据库语句,那么每一次工作中,代码量是十分巨大的。Mybatis 在这里向大家提供了一种可以根据数据库中的表自动生成数据库和 java 类的方法,十分便捷。

1 mybatis 介绍

Mybatis 本来是阿帕奇公司(即 apache,tomcat 出品公司)研究的一个面向广大用户的开源项目“ibatis”。其本意是 Internet 和 abatis 两个单词的组合,是提供给大家的一个持久层框架^[1]。后来 apache 公司将其转让给了谷歌公司,并且更名为 mybatis。Mybatis 自动生成方式替代了几乎所有的 JDBC 代码和参数设置以及结果搜索。它通过简单的 XML 文件对原始映射和属性进行配置,直接生成接口和 JAVA 对象的数据库表或者将数据库表生成接口和 JAVA 类。Mybatis 主要有三层主要功能框架:

1.1 API 接口层:将自动生成的 JDBC 代码封装提供给生成类以外的类或程序使用的接口 API,开发人员通过这些本地 API 对数据库进行操作。接口层一接收到外来调用请求就会调用数据处理层来完成具体的数据处理。

1.2 数据处理层:有具体的 JDBC 代码,是接口层接口的具体化代码,包括具体的 SQL 查找、解析、执行和执行结果反应处理等功能。它主要的目的是根据调用的 API 请求转为具体的代码,然后由代码完成一次数据库操作。

1.3 基础支撑层:负责最基础的管理功能,包括数据库连接和事务管理、数据库配置加载和缓存处理等内容,这些本应该是最基础的东西,将他们合成一个最基本的组件,为上层的数据处理层提供服务。

2 项目中 mybatis 应用方法

2.1 数据库配置

在底层数据库要将每一个需要调用的类配置成一个表的形式,并将数据类型和长度按照需要配置,如图 1。

名	类型	长度	小数点	允许空值	
id	int	10	0	<input type="checkbox"/>	1
assets_code	varchar	50	0	<input checked="" type="checkbox"/>	
dev_code	varchar	50	0	<input checked="" type="checkbox"/>	
dev_cate_code	varchar	30	0	<input checked="" type="checkbox"/>	
name	varchar	200	0	<input checked="" type="checkbox"/>	
dev_type	varchar	100	0	<input checked="" type="checkbox"/>	
node_code	varchar	20	0	<input checked="" type="checkbox"/>	
manufacturer_cate_code	varchar	30	0	<input checked="" type="checkbox"/>	
net_level_cate_code	varchar	30	0	<input type="checkbox"/>	

图 1

在表中,一定只能配置一个主键,并且将主键的配置完成。如图中,id 即为主键,且每一个属性名的类型都一定要选择清除,将长度配置完毕。

2.2 XML 文件配置

Mybatis 基本只需要两个最主要的 XML 文件,一个是 mybatis-config.xml,一个是 generatorConfig.xml,第一个是 mybatis 的功能配置文件,第二个是 mybatis 自动生成功能对表和类以及接口的配置文件。但是一般在项目中,mybatis 是和 Spring 框架结合使用的,在 Spring 框架的 applicationContext-database.xml 配置文件中,要将 mybatis 配置进去,,代码如下:

```
<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
    <property name="dataSource" ref="dataSource" />
    <property name="configLocation" value="classpath:mybatis-config.xml" />
</bean>
```

可以看到,Spring 也是讲 mybatis 功能当做一种反转注入的类工厂 (Factory),用以数据 (类) 的调用。这样,mybatis 就和 Spring 框架结合在一起了。在 mybatis-config.xml 中,将自动生成的所有接口的 JDBC 语言实现代码所在的文件配置进来,这样,mybatis 工厂就可以自动查询到每一个接口及其代码。其方式如下:

```
<mapper resource="maps/system/FieldConfigMapper.xml" />
```

下面重点介绍一下 generatorConfig.xml 文件,在其中,要将数据库的连接方式、账号密码等内容配置好,将自动生成的接口和 JDBC 语句所在文件夹配置好,还要将底层数据库的表和相对应的类配对好,写在一个 table 标签中,如下:

```
<jdbcConnection driverClass="com.mysql.jdbc.Driver"
    connectionURL = "jdbc:mysql://192.168.1.2/kiwi"
    userId="kiwi" password="kiwi">
</jdbcConnection>
<table tableName="port" domainObjectName="Port" >
<generatedKey column="id" sqlStatement="MySql"
    type="post" identity="true"/>
</table>
```

其中 jdbcConnection 标签是数据配置,table 是表和类配置,名字要一一对应。

在下一部分命令运行中,所有的数据库表自动生成类、接口和数据库语句都是通过 generatorConfig.xml 配置好后,才能自动生成的,在以后项目建设的日子,每当你类和数据库进行了修改后,都只需要运行命令,就可自动生成文件,不再需要去人工修改代码、参数和配置^[2]。

2.3 命令运行

在开发环境的服务器中,配置一句 mybatis-generator:generate,运行,即可在配置文件所写的文件位置中自动生成需要的类、API 接口和代码。这对于开发人员来讲,是一个解放程序员的进步。

3 应用结果

结合 RGB 空间和人工蜂群算法的彩色图像边缘检测

柳 欢

(安徽理工大学 电气与信息工程学院,安徽 淮南 232001)

摘 要:由于传统的梯度算子搜索边缘会丢失一些边缘,提出了一种在 RGB 彩色空间的边缘检测方法。该方法先获得 RGB 梯度图像,而阈值则通过人工蜂群算法搜索得到,进而得到边缘图像。

关键词:RGB;人工蜂群算法;边缘检测

中图分类号:TP391 **文献标识码:**A **文章编号:**1003-5168(2014)04-0022-02

1 引言

边缘蕴含了图像的很多讯息,是图像很重要的特征。传统的边缘检测^[1-2]方法有基于梯度的边缘检测,而这种方法,主要是针对灰度图,忽略了色度信息,容易丢失边缘。而在 RGB 颜色空间的边沿检测^[3]则能很好用到色彩信息,获得更丰富的边沿信息。但基于 RGB 彩色空间的边缘检测的阈值不好确定,本文将引入人工蜂群算法^[4]来获得阈值。

2 基于 RGB 空间的彩色图像边缘检测

在 RGB 空间中,令 $\vec{r}, \vec{g}, \vec{b}$,分别为 R, G, B 轴的单位向量,则 RGB 空间中任意一个向量可表示为:

$$\vec{q} = a1(x, y)\vec{r} + a2(x, y)\vec{g} + a3(x, y)\vec{b} \quad (1)$$

针对向量函数扩展梯度的概率,定义两个向量分别为

$$\vec{u} = \frac{\partial R}{\partial x}\vec{r} + \frac{\partial G}{\partial x}\vec{g} + \frac{\partial B}{\partial x}\vec{b} \quad (2)$$

$$\vec{v} = \frac{\partial R}{\partial y}\vec{r} + \frac{\partial G}{\partial y}\vec{g} + \frac{\partial B}{\partial y}\vec{b} \quad (3)$$

那么,

$$g_{xx} = \vec{u} \cdot \vec{u} = \left| \frac{\partial R}{\partial x} \right|^2 + \left| \frac{\partial G}{\partial x} \right|^2 + \left| \frac{\partial B}{\partial x} \right|^2 \quad (4)$$

$$g_{yy} = \vec{v} \cdot \vec{v} = \left| \frac{\partial R}{\partial y} \right|^2 + \left| \frac{\partial G}{\partial y} \right|^2 + \left| \frac{\partial B}{\partial y} \right|^2 \quad (5)$$

$$g_{xy} = \vec{u} \cdot \vec{v} = \frac{\partial R}{\partial x} \frac{\partial R}{\partial y} + \frac{\partial G}{\partial x} \frac{\partial G}{\partial y} + \frac{\partial B}{\partial x} \frac{\partial B}{\partial y} \quad (6)$$

那么 \vec{q} 的变化率最大的方向为

$$\theta(x, y) = \frac{1}{2} \arctan \left[\frac{2g_{xy}}{g_{xx} - g_{yy}} \right] \quad (7)$$

而沿着这个方向的变化率值为

$$F_{\theta}(x, y) = \left\{ \frac{1}{2} [(g_{xx} - g_{yy}) \cos 2\theta(x, y) + (g_{xx} + g_{yy}) + 2g_{xy} \sin 2\theta(x, y)] \right\}^{\frac{1}{2}} \quad (8)$$

又有 $\tan(\alpha) = \tan(\alpha \pm \pi)$, 如果 β 是式(7)的解, 那么 $\beta \pm \pi, 2$ 也是这个式子的解, 又因为 $F_{\theta} = F_{\theta \pm \pi}$, 所以 F 只需要对 θ 在 $[0, \pi]$ 之间计算。由式(7)可知, 在每个点 (x, y) 处有两个正交方向, 沿着这两方向, F 值获得最大。

3 人工蜂群算法

在人工蜂群(ABC)算法中, 蜂群被分为雇佣蜂、观察蜂和侦察蜂三类, 人工蜂群算法主要应用到寻找最优解, 本文将用它查找最佳阈值^[5]。

ABC 算法实现过程为: 先随机生成初始花蜜源, 再以类间方差作为适应度函数计算每个花蜜源的适应度值。雇佣蜂开始邻域搜索得到新蜜源并计算其适应度值, 与旧蜜源进行比较, 选择较为优秀的蜜源。观察蜂按轮盘赌法选择蜜源并进行邻域搜索, 再计算新蜜源的适应度值, 选择适应度值较高的花蜜源位置作为本次迭代的花蜜源位置。若是通过设置的限制次数循环后, 一个蜜源依然没有变化, 那么抛弃该蜜源, 随机生成新蜜源。算法如果达到了最大循环次数, 那就停止并给出最佳解, 否则继续进行。

4 基于 RGB 空间和 ABC 算法的边缘检测

本文提取彩色图像边缘方法步骤如下:

步骤 1: 对图像进行预处理。

步骤 2: 使用 sobel 算子分别得到 R, G, B 分量垂直方向和水平方向的偏导数, 并使用第二节的方法得到 $\theta(x, y)$ 和 $F_{\theta}(x, y)$ 。

在上面的命令运行后, 就在项目对应位置生成文件如图 2。其中 dao.* 中的 *Mapper.java 是所有关于 device 这个类的 API 接口, 包括增删改查和计数这些功能, 而且每种还根据用户需要提供了不同种类的接口。在 model.* 中主要是生成了数据库中 table 所对应的 java 类和类的 example, 用于类的调用和注入值等功能。Map.* 中是 dao.* 中接口的数据库代码化^[6]。其中把所有提供的接口和基础连接都已数据库语言表达出来。

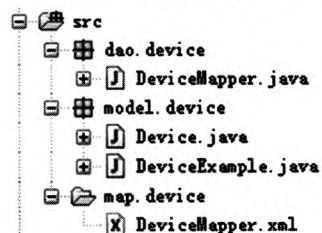


图 2

Mybatis 技术对 DAO 层进行了封装, 使得开发人员不需要在做后台和页面开发的同时还要去关系数据库的链接和调用。在配置文件中将 SQL 语句自动生成且封装, 避免了项目迁移时对数据库的依赖性, 降低了代码的耦合度, 增加了项目的可移植性。同时, 又能让你了解到底层数据库的搭建, 不至于对数据库完全不知晓。对 J 数据库语言的封装和 API 的生成, 减少了程序员在开发过程中使用 JDBC 的机会, 释放了大量开发时间, 解放了开发人员。

参考文献:

- [1]李澎林, 朱国清, 吴斌. 基于 iBatis SQL Map 的数据持久层实现[J]. 应用研究, 2008
- [2]夏汛, 陈玲. 基于 Spring MVC 和 Mybatis 的动态表单设计[J]. 计算机光盘软件与应用, 2012, 20.
- [3]徐雯, 高建华. 基于 Spring MVC 及 MyBatis 的 Web 应用框架研究[J]. 微型电脑应用, 2012.7.

4 总结

基于mybatis的面向数据库自动生成技术

作者: [黄艳秀](#)
作者单位: [河南省人才交流中心, 河南郑州, 450003](#)
刊名: [河南科技](#)
英文刊名: [Journal of Henan Science and Technology](#)
年, 卷(期): 2014(4)

参考文献(3条)

1. [李澎林;朱国清;吴斌](#) [基于iBatis SQL Map的数据持久层实现](#) 2008
2. [夏汛;陈玲](#) [基于Spring MVC和Mybatis的动态表单设计](#) 2012
3. [徐雯;高建华](#) [基于Spring MVC及MyBatis的Web应用框架研究](#) 2012(07)

引用本文格式: [黄艳秀](#) [基于mybatis的面向数据库自动生成技术](#)[期刊论文]-[河南科技](#) 2014(4)