

易于扩充的图像检索平台实现

朱 贇, 韩 炜, 吴 炜

(上海师范大学 计算机研究所, 上海 200234)

摘 要: 介绍一个易于扩充的图像检索平台之设计思想. 该平台通过在较高抽象层次上的对象描述技术, 实现了图像特征量的易扩充性、细节信息的高隐蔽性以及较高的系统安全性.

关键词: 多级抽象; 指针包装; 参照计数器; 实体生成类; 函数类

中图分类号: TP311.11 **文献标识码:** A **文章编号:** 1000-5137(2002)04-0028-05

0 引 言

基于内容的图像检索系统需要综合使用多类图像特征数据. 随着图像分类技术的发展, 图像特征的描述方法, 特征数据的综合使用手法随时都会发生变化, 从而导致图像检索系统发生大量的程序修改. 为了适应图像分类技术的频繁改良要求, 期望有一个高安全性, 易于扩充的系统开发平台. 概括地说, 这样的开发平台应满足下列条件:

易扩充性: 图像特征量的种类及描述方法发生变化时, 重编译模块仅限于发生变化的图像特征量描述模块. 即需要实现特征量相对于应用程序的透明.

高抽象度描述: 图像特征量的描述变更不影响使用该特征量的处理模块(即无需进行处理模块的重编译); 反之, 处理算法(如检索算法)的变更也不应影响作为多类图像特征量之集合的图像特征模块和各类特征量的描述模块.

系统安全性: 通过较高抽象层次的处理对象描述和隐蔽详细的对象生成以及管理方法等减少应用程序编制过程中的错误因素.

基于参考文献[1]中所介绍的图像检索算法, 实现了一个针对上述要求的图像检索平台. 平台使用 C++ 程序设计语言开发, 在综合运用如多级抽象类型定义、实体生成类(Factory class)、指针隐蔽等基于 C++ 的程序设计的基本技法的基础上, 以及不依存于具体程序设计语言的更高抽象层次上, 对对象抽象描述和降低类间耦合度进行了大量的工作. 通过对该平台测试系统和参考文献[2]中描述的系统之实装的比较, 证实了本平台的有效性.

本文将首先介绍系统的整体概念结构图, 然后介绍主要实现技术的详细内容.

收稿日期: 2001-01-12

作者简介: 朱 贇(1978-), 女, 上海师范大学数理信息学院研究生. 吴 炜, 男, 上海师范大学数理信息学院兼职教授.

1 系统整体概念结构

系统的整体概念结构图如图1所示。

系统由图像特征量描述模块和检索处理模块两大部分组成。特征描述模块定义用以分类图像所需的各类图像特征量以及特征量的存储方法,检索处理模块则定义运用各类图像特征进行检索的方法。

特征描述模块通过接口 `Interface_Image_Feature_Set` (图像特征集合接口) 和特征处理模块相关联。该接口类通过对抽象的集合元素之顺序和对元素的访问方法的定义隐蔽了具体集合的实现方法,从而极大降低了特征处理模块和特征描述模块的耦合度。

不同的检索处理算法实现也同样地通过接口类 `Interface_Retrieval_Alg` 和特征描述模块相关联。这使得特征描述不再依存于具体的检索算法实现。

`Image_Feature_Array` 是最基本的图像特征量数列。通过对数列元素定义不同的顺序实现具体的图像集合数据结构(如树,索引表等)。为了提高检索处理速度, `Image_Feature_Array` 对图像特征量的管理采用了“纵列”结构,由复数个(可变)的单一特征量之数列构成。不同的单一特征量数列通过接口 `Interface_Feature_Array` 屏蔽了其各自的差异,使得 `Image_Feature_Array` 能使用统一的方法存储和访问其所管理的特征量数据。

`Image_Feature` 存储一幅图像的所有特征量数据。该类的实体实际上不存在,只当外部向 `Image_Feature_Array` 请求一幅具体图像的特征量时,系统才通过拷贝 `Image_Feature_Array` 中相应数据生成一个临时的 `Image_Feature` 实体。设计 `Image_Feature` 的目的是为了提供一个更加直观的图像特征描述。为了使 `Image_Feature` 能使用统一的方法存储和匹配不同的具体特征量,引入了关于特征量的抽象接口类 `Interface_Individual_Feature`,它定义了抽象的特征量匹配操作接口。

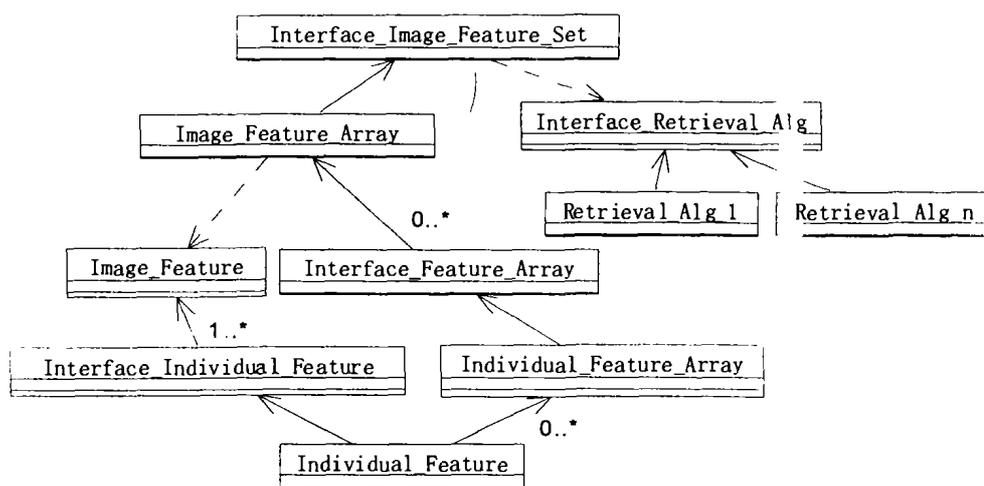


图1 系统整体结构

2 图像特征描述

一幅图像的分类特征是由多类特征量综合描述的,每一类特征量称为一种单一特征量。每种单一特征量的表现方式、抽取算法各不相同。一个实用的检索系统必须实现各种单一特征量的均一化,并对之进行有效的管理,才能减少特征量变更对程序的影响,提高系统性能。

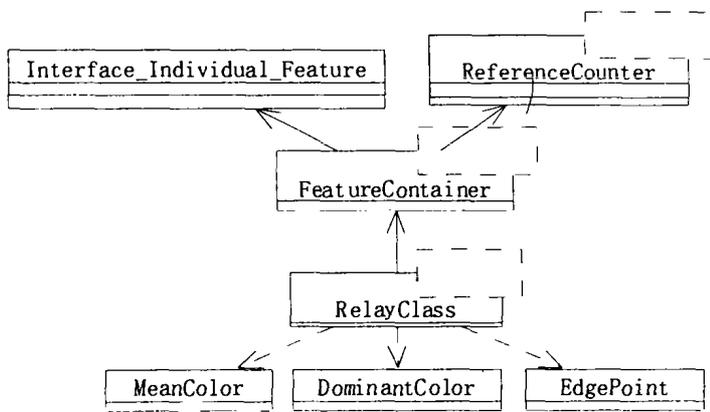


图2 特征实现的结构

2.1 单一特征的实现

我们为每一种特征建立相应的特征类,类内部定义了特征的存储空间以及特征抽取、匹配的方法.类的实体即图像的某一种具体特征(向量)值.特征实现的结构如图2所示.

接口类 Interface_Individual_Feature 实现了特征量的均一化,使对图像的各类特征量处理能够不依赖于特征的具体实现.同为父类的参照计数类 ReferenceCounter 则用来管理特征值数据,实现数据空间的

自动释放.通过参照计数类的使用,实现了不使用显式指针的编程,从而降低了因为指针释放时刻错误而造成的程序执行错误,降低了编程的复杂性.

模板类 FeatureContainer 用以存储特征量数据,起了容器(Container)的作用,其模板参数指明了所存特征量的类型.该类体现了特征量的基本功能,同时又隐蔽了特征量的内部细节.它能够根据模板参数为特征值开辟相应类型的存储空间,自动调用相应特征类的特征操作方法.外部应用程序面对的只是该类提供的特征操作函数,不需要考虑具体特征的存储方式,从而避免了因特征量的变更而引起的相关程序修改工作.

RelayClass 是应用程序和 FeatureContainer 之间的接口,在不影响特征类的基本构成的情况下向应用程序提供更高级别的服务.

2.2 图像特征集 Image_feature 的实现

对一幅图像所包含的多类特征进行有效的组织,可以提高系统的运行速度(尤其是检索速度),也可以简化特征量的扩充和删除.为了实现上述目的,主要使用了指针包装技术.

图像特征集存储一幅图像的所有特征量数据.它为每类特征建立一个特征项.特征量的扩充和删除只是对相应特征项和对相应特征类的增删.对某特征的修改只在相应特征类中进行,不会影响其他的特征类和程序结构.图像特征集是特征集合的“容器”,只提供了作为“容器”所需的必要操作,所以不同的单一特征量可以很容易地加入或从中删除,系统可以方便地保存和管理各类特征量,在很大程度上简化了程序.结构如图3所示.

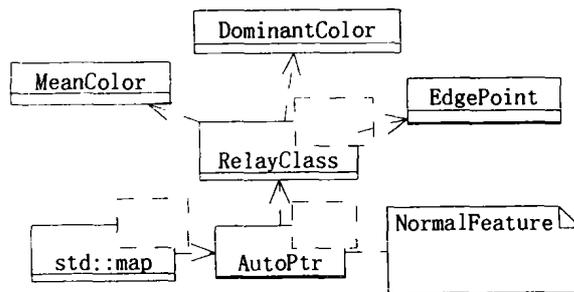


图3 图像特征集

由于不同特征类所对应的中继类 RelayClass 实体类型各不相同,我们使用 RelayClass 的父类 Interface_Individual_Feature 来存储各种中继类实体,以统一存储不同类型的特征.但是其父类是一个虚类,无法产生实体,为解决这一问题引入了指针包装的技术,生成代理类 NormalFeature,通过存放代理类的结构达到存放特征实体的目的(图3). NormalFeature 的结构为: AutoPtr<Interface_Individual_Feature, EntityFeatureInterface>. 其中, AutoPtr 模板类用成员变量存储类实体,类型为所存类实体的统一接口. EntityFeatureInterface 是从虚类 Interface_Individual_Feature 派生的空实

类, 作为成员变量的初始值, 这样就可以解决虚类无法产生实体的问题. 当存放特征值的 Relay-Class 实体被传入代理类, 代理类复制该实体, 然后将复制品赋值给其成员变量, 即完成了特征值的存储操作.

3 对应用程序隐蔽细节

图像检索系统的处理对象经常变更. 为保证系统的可靠性和易扩充性, 使用了实体生成类和函数类技术, 使系统平台能够向应用程序隐蔽内部的具体实现细节.

3.1 使用实体生成类隐蔽图像登录细节

图像特征库以单一特征量数列的形式, 存储了系统全部的图像特征. 其包含的特征种类往往是动态变化的, 如果由应用程序根据其需要来建立图像特征库, 必然会增加应用程序的负担. 为此, 我们使用了特征实体生成类 FeatureFactory、图像实体生成类 ImageFactory (图4) 和实体生成类接口 InterfaceFactory, 有效地隐蔽了检索平台内部的实现细节, 简化了应用程序和检索平台之间的接口.

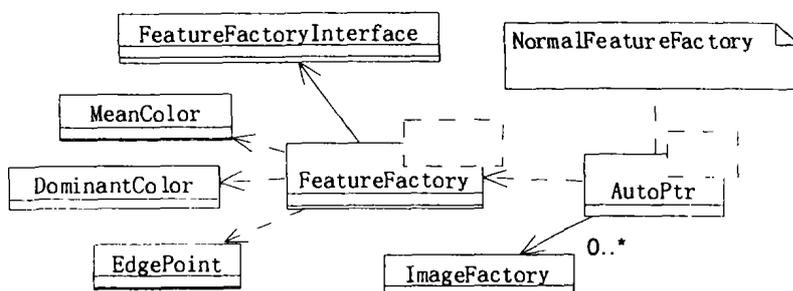


图4 图像实体生成类

特征实体生成类 FeatureFactory 用以均一化各种特征实体的生成操作. 该类可根据其模板参数自动对图像抽取不同类型的特征, 并将特征值以统一的接口类型返回给应用程序. 该类的实体即表示某一种特征类型.

图像实体生成类 ImageFactory 使检索平台能够自动抽取所需的各类特征. 将所有表示不同特征类型的 FeatureFactory 实体, 用代理类的形式, 预先统一存储在图像实体生成类中. 当抽取图像各类特征时, 图像实体生成类逐一取出 FeatureFactory 类实体, 由该实体生成相应的特征, 这样就完成了特征的自动抽取. 图像特征库的初始化也由图像实体生成类来完成. 该类通过特征实体生成类的实体生成相应类型的空特征列, 然后按照该特征的标示 ID, 将这些空列插入图像特征库, 完成初始化工作.

实体生成类接口 InterfaceFactory 是检索平台和外部应用程序之间真正的接口之一. 图像登录时, 外部应用程序只需调用类中的接口函数, 即可方便地建立图像特征库.

3.2 使用函数类隐蔽图像检索细节

一个实际的图像检索平台将包含多种检索方法. 为了方便管理和调用各种检索方法, 我们采用了函数类的技术思想. 结构如图5所示.

每个检索函数类 RetrieveFunction 包含一种检索方法的函数. 进行图像检索时, 根据检索需求, 把相应检索函数类传入检索调用类 RetrieveInvoke, 即可进行所需的检索.

使用了函数类, 在 C++ 中, 系统编译的时候可以进行合法性检查, 提高了系统的安全性. 只需

要将相应的函数类提供给不同用户,用户需求的更改也只涉及到其中某一个函数类,方便了版本的管理.使用统一的接口 RetrieveOperator,简化了调用过程,避免了传统的传递函数指针所带来的安全性低的问题.应用程序只需调用检索调用类的检索接口,隐蔽了系统内部所实施的图像检索的细节.

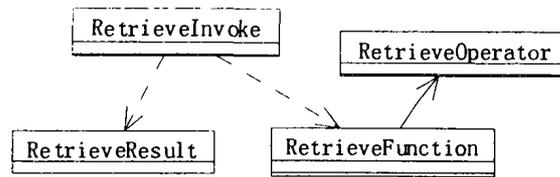


图5 函数类

4 结束语

本文介绍的图像检索平台运用多级抽象、指针包装、参照计数器、实体生成类、函数类等技术,隐蔽底层数据指针,封装数据和操作,向应用程序隐蔽实现细节,提高了数据描述、数据操作各模块之间的独立性,减少了程序的重复,简化了程序变更的过程.同参考文献[2]中的系统相比,在安全性、可扩展性等方面有明显优势,而且应用参照计数器作为专门管理底层数据的工具,既可以优化数据管理,又减少了系统程序的工作.该系统作为一个实用项目已经开发完成并投入运行使用.

参考文献:

- [1] 韩炜,吴炜.基于颜色和轮廓点分布特征的高速图像检索[J].上海师范大学学报(自然科学版),1999,28(3).
- [2] 韩炜,吴炜.一个基于颜色分布特征的实用图像检索系统[J].计算机工程与应用.2000,36(7).
- [3] 张世琨,王立福.基于体系结构的软件开发模式[J].世界科技研究与发展.1999,21(3):31-35.
- [4] 许幼鸣,徐锦.基于设计模式的软件重用[J].计算机工程,1999,25(3):13-14,36.

Design of an Extensible Image Retrieval Platform

ZHU Yun, HAN Wei, WU Wei

(Computer Research Center, Shanghai Teachers University, Shanghai 200234, China)

Abstract: Introduces a design of an extensible image retrieval platform. The platform uses an object description technique at a higher abstract level, which realizes easy extension of image characteristics and highly hidden details so as to ensure a relatively high system security.

Key words: multilevel abstract; pointer hidden; reference counter; factory class; function class