

多媒体查询语言及其评价准则

曹忠升 吴宗大 王元珍

(华中科技大学计算机学院数据库和多媒体技术研究所 武汉 430074)

摘要 随着多媒体技术和多媒体相关应用的发展,对有效检索多媒体信息的要求越来越迫切。查询语言作为信息检索的有效工具,其研究也越来越受到关注。对现有的多媒体查询语言进行了全面的综述,将它们分为两类:专用语言和通用语言。尽管查询语言对多媒体信息系统提供有效的查询服务至关重要,但目前仍没有较好的多媒体查询语言评价准则。因此,针对多媒体查询语言的查询表达能力设计准则共 16 条,并根据这些准则对多媒体查询语言进行了评价。评价结果表明,这些语言能满足用户的基本查询要求,但在高级语义查询和不确定查询等方面还有欠缺。最后展望了多媒体查询语言的未来研究方向。

关键词 多媒体信息检索,查询语言,评价准则

Multimedia Query Languages and their Evaluation Criteria

CAO Zhong-sheng WU Zong-da WANG Yuan-zhen

(College of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract The development of multimedia technologies demands support in effective multimedia information retrieval. So as a crucial component of information retrieval, many multimedia query languages had been proposed. The paper made an overview on existing multimedia query languages, which are classified into: special languages and unified languages. Then we proposed sixteen properties as the criteria, and based on them made the evaluations for these existing multimedia query languages. The result shows most languages satisfy basic query requirements, but in advanced requirements there are rooms for further enhancement. Last, the paper concludes by providing several research directions for multimedia query languages.

Keywords Multimedia information retrieval, Query language, Evaluation criteria

1 研究背景和概述

随着多媒体技术及其相关应用的迅速发展,多媒体数据大量增加。如何有效查询多媒体数据,已成为多媒体信息检索技术中最迫切的问题之一。因此,作为多媒体数据查询的有效工具之一,多媒体查询语言应运而生。常见的多媒体数据类型有图形、图像、音频、视频、动画、多媒体文档等,虽然形式多样,但概括来说,其信息主要分布在 3 个层次上:

(1) 底层特征。通常它们是从原始多媒体数据中利用计算机自动分析技术提取出来的特征,如图像的颜色、纹理等。

(2) 内容信息。观看多媒体数据时所感知到的内容,“多媒体数据中蕴含着些什么以及它们怎么样”。

(3) 时空信息。多媒体实体通常蕴含时间和空间信息,因而它们之间存在着时空关系。如视频中事件的时序关系等。

20 世纪 90 年代以前,多媒体信息自动提取技术尚处于初始阶段,多媒体数据的高层信息难以被自动提取,因而该时期的多媒体信息检索多采用基于底层特征的实例查询(Query By Example, QBE),并不需要专门的查询语言,典型的有

MULTOS^[1], QBIC^[2], VIMSIS^[3], CORE^[4] 等。由于多媒体数据结构复杂,其蕴含的丰富信息难以用底层特征简单描述,使用户经常难以检索到感兴趣的记录,并且有些情况下也难以给出查询样例。随着多媒体处理技术的发展,对原始多媒体数据实现更细粒度的处理成为了可能。系统开始以人们所理解的方式来表达和存储多媒体数据中的信息。使用各种实体(包括现实世界中的具体事物、抽象概念等)以及实体之间的关系表示多媒体内容,称作内容信息和时空信息。因而从 20 世纪 90 年代初开始出现专门针对多媒体的查询语言,经历了从专用查询语言到通用查询语言的发展过程。专用多媒体查询语言是指针对某一特定多媒体数据类型(如图像数据库、视频数据库等)的查询语言,它们的语言要素均是针对其服务的多媒体数据类型专门设计的,如针对图像的 SEQ^[7], FOQL^[8] 等;针对视频的 VideoSQL^[13], VQL^[14] 等。它们的共同不足是:只限于特定类型数据库,难以移植到其它类型的数据库上。随着多媒体数据大量增加,迫切需要一个统一的多媒体数据库平台来存储和检索各种类型的多媒体数据,于是通用多媒体查询语言应运而生,它的特征主要表现为:一是可以表达多媒体数据中出现的各类实体,以及这些实体间的上

到稿日期:2008-04-16 本课题得到国家 863 高技术研究发展计划项目基金(编号:2006AA01Z430)资助。

曹忠升 副教授,主要从事多媒体数据库研究;吴宗大 博士研究生,主要从事多媒体查询语言及其查询处理研究, E-mail: zongda1983@163.com;王元珍 教授,主要从事现代数据库理论及实现技术研究。

下文关联;二是能对各类型的多媒体数据以统一的方式实现检索。

纵观多媒体查询语言的发展,各种语言在查询表达能力、支持媒体类型、语法形式等方面既有所同,又有不同,人们尚缺乏一个较为全面的规律性认识。为此,我们从阅读相关的文献入手,通过对比分析总结出对多媒体查询语言进行评价的统一认识和标准,进一步引申出多媒体查询语言的深入研究。

2 现有多媒体查询语言综述

本节介绍现有的多媒体查询语言。表1列出了本文涉及的查询语言以及它们支持的媒体类别、语法形式和主要特点。

表1 现有多媒体查询语言概况

名称	类别	形式	主要特点
PSQL	图像	描述性	支持图像空间信息查询
ESQL	图像	描述性	基于关系数据库模型
SEQL	图像	描述性	支持空间信息和图像实体的版本查询
FOQL	图像	描述性	支持图像空间信息和模糊查询
IFQ	图像视频	图符式	支持图像特征、时空查询和视频查询
CB_VMQL	图像视频	图符式	支持图像特征、时空查询和视频查询
VideoSQL	视频	描述性	基于标注模型,通过属性来表达查询
VQL	视频	描述性	基于标注视频模型
SVQL	视频	描述性	基于AVIS模型,支持事件对象蕴含查询
CVQL	视频	描述性	支持视频对象的时空查询
BVQL	视频	描述性	支持底层特征、轨迹、时空、聚集查询
EVA	通用	逻辑性	面向对象语言,不支持空间和视频查询
FBQL	通用	逻辑性	大量使用谓词和子函数来完成查询表达
MQL	通用	描述性	支持对象、模式匹配和版本查询
MOQL	通用	描述性	支持底层特征和时空信息查询
MMOQL	通用	描述性	支持底层特征、内容和时间信息查询
UMQL	通用	描述性	支持底层特征、内容和时空信息查询
MQuery	通用	图符式	基于ER-OO模型,支持内容和时空查询
WS-QBE	通用	图符式	基于查询表,支持相似性查询

2.1 专用多媒体查询语言

专用多媒体查询语言是早期多媒体信息检索技术研究的产物,它们只针对特定类型的多媒体数据库,难以移植到其它类型的数据库上。根据针对类型的不同,专用多媒体查询语言又可以分为很多子类别,如图像查询语言、视频查询语言等。

2.1.1 图像查询语言

图像查询语言是针对图像数据库的专用查询语言,要求能够表达图像的底层特征(即视觉特征,如颜色、纹理等);图像内部实体的特征,以及实体之间的空间关系。早期的图像检索系统只支持基于图像视觉特征实例查询^[1-4],虽简单易用,但查询表达能力弱。上世纪90年代后,专门图像查询语言开始盛行,主要方法有:基于文本的描述性语言,如PSQL^[5],ESQL^[6],SEQL^[7]等;基于图形的可视化语言,如IFQ^[9],CB_VMQL^[10]等。

PSQL^[5]在SQL的基础上添加空间算子,因而能够支持空间直接搜索、空间直接计算和空间非直接搜索。ESQL^[6]的模型基础是支持图像类型的关系数据库,该模型使用关系表来对图像结构建模,因而ESQL的语法形式与SQL一致,且其图像查询是通过查询图像实现的。这要求用户必须了解各图像表的构造,并把图像查询转换为对图像表的查询,这显然是繁琐的。SEQL^[7]在SQL基础上增加了空间、时序和演化条件,分别以WHERE,WHEN和WHICH子句表达。以

上3种图像查询语言的查询表达均是精确的,但有时用户的多媒体查询需求却是模糊的,如找出所有类似某一给定目标的图像。这需要查询语言能够支持模糊查询。FOQL^[8]是一种能支持模糊查询的类OQL语言,但缺少在语言层次上对空间查询的支持。文本式查询语言的用户界面相对不够友好,为此提出可视化查询语言。IFQ^[9]和CB_VMQL^[10]采用图形符号来描述图像信息,使用户能够以可视化的方式表达其多媒体查询需求。经过少量的扩充,IFQ和CB_VMQL还能支持视频内容查询。可视化查询语言通常基于某一文本式查询语言,因而它们的查询描述能力均要弱于(至多等于)它们所基于的文本查询语言。

其它的还有:支持演化和时序查询的PICQUERY+^[11]、支持模糊查询图像检索系统SCORE^[12]等。综合来说,目前的图像查询语言语法各异(扩充SQL或OQL的、全新的、图符式等),但就查询能力而言,基本满足图像数据库的检索需要。

2.1.2 视频查询语言

视频查询语言是针对视频数据库的专用查询语言,要求能够描述视频的底层特性;视频的各种内部实体,包括物理实体(如视频段)和逻辑实体(如事件);视频内部实体的时空信息;视频的内部实体之间的上下文关联。

由于视频语义信息自提取技术的限制,早期的视频查询语言的基础通常是基于标注的视频库,它在视频流上叠加一个标注层,表示视频的内容信息。VideoSQL^[13]是针对OVID模型而提出的。OVID把视频中的各逻辑视频流看作视频对象,它们相互独立且拥有各自的属性集合。VideoSQL是类SQL语言,其查询描述能力依赖于视频属性集合。VQL^[14]的WHERE子句由视频、对象、事件、图像特征、相似性、空间关系、时间关系等条件表达式组成,能够表达视频的各层次信息,其基础模型也是基于标注的。基于标注的视频模型本身固有的缺陷限制了其查询语言的表达能力,如模型缺乏对语义标注之间关系的支持,使查询语言无法基于实体之间复杂的上下文关联来查询目标。所以从90年代后期开始,出现了基于语义的视频模型,相应的查询语言也随之出现。AVIS^[15]模型将视频分割成固定大小的视频段,并以树存储。树中的每个节点保存各视频段中出现的对象和事件。SVQL^[16]便是针对该模型提出的,其WHERE子句指定目标视频的特性,分为对象条件和事件条件,分别用于描述视频序列包含的对象和事件。但SVQL不支持对象的空间查询和事件的时序查询,也不支持视频段的时空查询。基于内容的查询语言CVQL^[17]的查询语句由Range、Predicate和Target3个部分组成:Range指定查询范围;Target指定查询结果的形式;Predicate指定查询条件。该语言能表达视频对象的时空信息,但它把视频对象看作二维空间中的一个点,因而无法支持空间拓扑查询,也不支持对象相似性查询以及对象运动轨迹查询。BilVideo^[18]是一个视频数据库管理系统,提供对视频的时空查询和语义查询的支持,其查询语言是BVQL。BVQL是类SQL查询语言,通过扩充谓词和算子,BVQL能够支持对象、时空、对象轨迹、底层特征、语义等查询。

其它的还有:支持空间关系查询的SpatialSQL^[19]、支持视频对象存在性查询的MMQL^[20]等。综合来说,由于视频模型的不完善,难以给查询提供足够丰富的服务,目前的视频

查询语言在基本的视频信息描述能力(如视频底层特性)上基本满足用户的要求,但在高级语义信息方面还有所欠缺。

2.2 通用多媒体查询语言

通用多媒体查询语言用于统一检索各种类型的多媒体数据。虽然多媒体数据类型丰富,查询需求各异,但它们需要描述的信息主要表现在3个层次上:底层信息、内容信息和时空信息。通用多媒体查询语言正是基于此设计的。

EVA^[21]是早期的多媒体查询语言,它把功能性语言的特点和传统的代数理论相结合,使用存储代数算子来表达查询,能支持对时序查询和集合操作,但并不支持空间查询和视频查询。文献[22]提出了一个独立于具体的多媒体应用形式化的多媒体信息系统框架,并给出了该框架下的逻辑性查询语言FBQL。EVA和FBQL都是逻辑性查询语言,这类语言的表达能力较强,但不易使用,它们更适合作为较高级的语言(如描述性语言)和较低级语言(如查询代数)的转换中介语言。

MQL^[23], MOQL^[24], MMOQL^[25], UMQL^[26], MQuery^[27], WS-QBE^[28]和VMQ^[29]均是多媒体数据库管理系统层次的多媒体查询语言,能支持所有常见的多媒体数据类型。MQL^[23]包括DDL和DML两部分,能支持对象查询、模式匹配查询和版本查询,这显然不能满足复杂的多媒体查询需求。MOQL^[24]是OQL的扩展,它在WHERE子句中引入了空间表达式、时间表达式和蕴含谓词,其中空间表达式包括空间对象、空间函数和空间谓词;时间表达式包括时间对象、时间函数和时间谓词;蕴含谓词则是用于判断一类多媒体对象是否蕴含另一类特定的内部对象。MMOQL^[25](Multimedia Object Query Language)以数字图书馆应用为背景而提出,它在SQL的基础上增加了WHEN,STRUCT和SYNCH子句,分别用于说明目标满足的时间条件、结构特征和时序关系,但MMOQL缺乏对空间查询的支持。UMQL^[26]在SQL的基础上引入结构条件、特征条件和时空条件,能够表达多媒体数据的内容信息、结构信息以及进一步表达各多媒体实体之间的各种关系。MQuery,WS-QBE和VMQ均是基于图形符号的可视化查询语言。MQuery^[27]基于实体关系模型(ER)和面向对象模型(OO),定义了相关的查询表达构造子,用于表达对象之间的各种关系。此外还有构造子:谓词盒子、结果边缘、IVE盒子等,其中谓词盒子说明目标对象的属性;结果边缘说明结果对象的投影属性;IVE盒子说明IVE对象。WS-QBE^[28]是在QBE基础上提出来的,其基本的语言元素是查询表,包括权重表、时序表和空间表。WS-QBE使用方框来描述一棵条件树。WS-QBE能较好地支持相似性查询,但其相关文献没有提及WS-QBE如何表达多媒体时空信息。VMQ^[29]是基于UMQL的可视化查询语言,其表达能力与UMQL等价。

综合来说,早期的通用查询语言对多媒体语义信息的表达不甚理想,而后期的查询语言则能较好地表达各类型多媒体数据的内容和语义信息,但对于高级语义信息的表达则欠佳。

3 评价准则和现有语言的评价

从第2节可知,语言的设计者通常只给出语言的语法形式,没给出语言的设计要求。本节就多媒体查询语言应具备

的查询表达能力给出了全面的评价准则,并根据这些准则评价现有的查询语言。这些准则是针对通用多媒体查询语言的,但由于视频是最复杂的多媒体数据类型,其查询需求与多媒体查询需求基本一致,因而也适用这些准则。据此,对表1的5种视频查询语言和8种通用查询语言进行了评价。

评价准则是以规则的形式给出的,分为3类:内容信息表达能力、时空信息表达能力以及不确定信息表达能力。

3.1 内容信息表达能力

多媒体内容信息是人们在观看(或倾听)多媒体时所感知到的内容,主要分布在4个层次上:概念内容;感知特性,如视觉特性和听觉特性;逻辑关系,即语义上下文关联等;特定领域的特征,如人的面部特征,需要特定领域知识的支持。本节总结出多媒体内容信息表达能力的6条评价准则:C1—C6,并据此对各查询语言进行了评价,结果如表2所列。

表2 准则C1—C8的评价结果

名称	C1	C2	C3	C4	C5	C6	C7	C8
VideoSQL	√	×	×	×	√	×	×	×
VQL	√	√	⊙	×	×	×	√	√
SVQL	×	×	×	×	×	√	×	×
CVQL	×	×	×	×	×	×	√	×
BVQL	√	√	√	√	×	√	√	√
EVA	√	×	√	√	√	×	√	√
FBQL	√	√	⊙	√	√	×	√	√
MQL	√	×	×	×	×	×	×	×
MOQL	√	√	×	×	√	×	√	√
MMOQL	√	√	√	×	×	×	×	×
UMQL	√	√	√	√	√	×	√	√
MQuery	√	×	×	×	√	×	√	√
WS-QBE	√	√	√	×	×	×	×	×

√—支持;×—不支持;√—一部分支持;⊙—不确定

C1:是否支持实体自定义属性查询。属性刻画了实体的特性,因而多媒体实体可以用与领域相关的自定义属性进行描述。大部分语言均能支持多媒体实体的自定义属性查询,但形式有所不同:VideoSQL使用谓词来指定实体的属性值;逻辑性语言EVA和FBQL使用方法来获取属性值;可视化查询语言MQuery和WS-QBE分别使用谓词盒和表格来描述实体的属性;其余语言直接引用实体属性。

C2:是否支持实体底层特征查询。不同多媒体数据表现出不同的底层特征,如图像的颜色、纹理;视频的镜头切换类型、摄像头的运动等。尽管单纯的底层特征查询无法满足用户丰富的多媒体查询需求,但是否包含底层特征查询仍然是衡量多媒体查询语言的查询表达能力的准则之一。表2显示共有7种查询语言支持底层特征查询。

C3:是否支持实体相似性查询。查找类似于目标的实体是多媒体数据库中常见的查询类型,这种相似性度量通常基于某种或某几种底层特征。VQL提供相似性表达式来支持相似性查询。BiVideo以数组记录了视频中对象的运动轨迹,因而其查询语言BVQL能够支持对象运动轨迹相似性查询。EVA使用存储代数算子ImageSim和AudioSim来支持图像和音频的相似性度量。MMOQL和UMQL分别使用算子和特征函数来支持图像的相似性查询,而图表语言WS-QBE则以表格完成这一过程。

C4:是否支持实体概念性查询。概念性内容用于表达多媒体数据的对象语义,如对于图像的内部实体,可以是各种概

念性的对象(如马、牛、羊等)。因而,能否根据多媒体内部实体的概念特征来区分它们的类别是多媒体查询语言的重要评价准则之一。BVQL定义了一套完整的函数来判断视频段中的实体概念性类别。EVA和FBQL把多媒体的内部实体按照它们代表的逻辑概念分散在若干个表中进行存储,自然而然地支持了该类查询。UMQL提供系统函数IS用于指定一类实体所代表的概念性语义。MQuery则用谓词盒来完成这一过程。

C5:是否支持实体之间的蕴含关系查询。多媒体实体之间的蕴含关系是一类上下文关联语义,是多媒体实体间很重要的一类关系。蕴含关系查询主要分为两个层次:是否能表达两类实体之间的蕴含关系;是否能指定被蕴含实体的数目。VideoSQL提供CONTAINS关键词用于指定视频事件所蕴含的帧数目,或声明事件所蕴含的对象。EVA,FBQL和MOQL使用Isin谓词来指定各类别实体间的蕴涵关系。UMQL则比较特殊,它使用结构条件式来声明实体蕴含的子实体类别并且可以指定各类别子实体的数目,然后在后续的条件中再描述它们的特征。MQuery则采用OO模型图的方法来完成蕴含关系查询。

C6:是否支持实体间的上下文关联查询。多媒体实体间的上下文关联是指实体间的逻辑关系,蕴含关系便是一类上下文关联,但上下文关联并不仅限于此,如视频事件的因果关联、视频对象在视频事件中的角色关联等。所以,这里指的是除蕴含关系之外的上下文关联,是更高层的语义。从表2可以看出,仅SVQL和BVQL能够支持角色关联查询。

3.2 时空信息表达能力

多媒体中包含丰富的时空信息,如视频中事件的时序关系以及图像中实体的空间关系等。多媒体的空间信息主要表现内部实体的空间关系(拓扑关系、位置关系等)。本节总结了多媒体时空信息表达能力的6条评价准则:C7-C12,并据此对各查询语言进行了评价,结果分别在表2和表3中给出。

表3 准则C9-C16的评价结果

名称	C9	C10	C11	C12	C13	C14	C15	C16
VideoSQL	×	×	×	×	×	×	×	√
VQL	√	×	√	×	×	×	×	√
SVQL	×	×	×	×	√	×	×	√
CVQL	√	×	×	×	×	×	×	√
BVQL	×	√	√	×	×	×	×	√
EVA	×	×	√	×	×	×	×	√
FBQL	×	×	√	×	√	×	√	√
MQL	×	×	×	×	×	×	×	√
MOQL	√	×	×	√	×	×	×	√
MMOQL	×	×	√	√	×	×	×	√
UMQL	×	√	√	×	×	×	×	√
MQuery	×	×	√	×	×	×	×	√
WS-QBE	×	×	×	×	×	×	×	√

√—支持;×—不支持;√—部分支持;○—不确定

C7:是否支持空间位置关系查询。在平面空间里,基本的位置关系有North, South, East, West, North-East, South-East, North-West和South-West。查询语言根据这8种位置关系来定义相应的谓词或函数来支持空间位置关系查询,如VQL, BVQL, MOQL, FBQL和EVA。CVQL使用视频函数AP和RP来分别计算视频帧内的实体的绝对位置坐标和相

对位置关系。在UMQL中,多媒体空间实体被看成一个长方体,实体之间的位置关系通过它们在XYZ方向上的投影关系分别予以描述,这样UMQL便能等价地表达8种基本位置关系。

C8:是否支持空间拓扑关系查询。基本的空间拓扑关系有:Disjoint, Touch, Inside, Contains, Overlap, Covers, Covered-By和Equal。类似地,VQL, BVQL, MOQL, EVA和FBQL均采用拓扑谓词或函数来支持拓扑的查询。MQuery通过图符的方式指定空间对象的拓扑关系。在UMQL中,多媒体空间实体的拓扑关系则是通过实体在XYZ上的投影来表达的。CVQL把空间实体看成一个点,因而无法表达实体间的拓扑关系。

C9:是否支持空间计算。在多媒体查询中,有时需要一些空间相关的计算,如计算空间线实体的长度、两实体之间的距离等。VQL通过谓词Near和Far来表达实体的距离关系,但不能给出实体间的距离,属于模糊谓词。CVQL提供二元函数RP计算实体之间的相对坐标,可获得实体之间的距离。针对这些空间对象MOQL定义了一系列的空间计算函数,如点点之间的Nearest和Farthest等。

C10:是否支持3D空间查询。前面的空间相关查询一般都是针对二维平面的,该准则是评价多媒体查询语言对三维空间内多媒体实体关系的表达能力。从表3中可以发现,除BVQL和UMQL外,其他查询语言均不支持3D空间查询。BVQL定义了一整套3D空间关系函数来支持3D空间查询。在BVQL中,视频内容逻辑关联都是通过函数来实现的,这种做法导致系统中出现大量的函数。而UMQL使用同一套时空谓词来描述空间实体在XYZ三个方向上的投影的关系,从而实现了对3D空间查询的支持,因而UMQL对时空查询的支持显得较为自然。

C11:是否支持时序关系查询,即多媒体数据的同步关系查询。两个时间间隔主要有7种不重复关系,分别是Before, Meets, Overlaps, During, Starts, Finishes和Equals。多数语言都是采用定义时序谓词或者时序函数的方法来实现对时序关系查询的支持,如VQL, MMOQL, UMQL, BVQL, EVA和FBQL等。MOQL能表达时间间隔之间的交、并和差关系,但这是不够完备的。

C12:是否支持时间计算。对携带时间信息的多媒体实体进行时间相关的计算,如求时间间隔的起始点和终止点等。从表3中可以看出,仅MOQL和MMOQL能够支持时间计算。MOQL对于时间计算的支持分为两个方面:一是定义了时间点计算函数;二是定义多媒体连续实体计算函数。MMOQL主要定义一些针对时间间隔实体的计算谓词,如BeginOf等。

3.3 不确定信息的表达能力

由于多媒体内容的丰富性,使数据库中存储的语义信息不完整或不确定,因而要求能够支持不确定查询。本节总结了多媒体不确定信息表达能力的4条评价准则:C13-C16,并据此对各查询语言进行了评价,结果在表3中给出。

C13:是否支持模糊查询。“查询黑头发、大眼睛且鼻子有点歪的男子”,这类查询是难以精确描述的,因而要求查询语言能够动态地返回结果。从表3中可以看出,只有SVQL和FBQL能够支持模糊查询。在FBQL中,系统定义了多媒

体特征的偏序关系集,当用户给定的条件无法得到满意结果时,系统会自动根据特征偏序关系集来修改查询条件,直到得到满意的结果。SVQL 则允许用户为视频事件和视频对象指定其包含的关键词时指定确定度,以此来实现模糊查询。

C14:是否支持增量查询。有时用户对目标并不清晰,他可能先定义一个查询,然后一步步地修改查询,直到获得目标。这需要查询语言能够支持增量查询,即查询可以在以前的查询结果上进行。目前的查询语言都没有提供该功能。

C15:是否支持查询改写。当多媒体查询结果为空时,原因可能是查询定义或匹配过程的不确定性。这时查询处理器应该自适应地减弱查询条件。在另一种情况下,如果查询返回的结果集过大,就需要对查询条件进行加强,以减少返回的结果。从表 3 中可以看出,仅 FBQL 能够支持查询自动改写。

C16:是否支持复合查询。该评价准则是评价查询语言是否能将各种基本查询条件复合成为一个复杂的查询条件。这通常的要求是语言提供布尔操作符(与、或、非)。图形化查询语言 MQuery 和 WS-QBE 只能指定目标应满足的多个条件(即与操作),但是不能支持或操作。

结束语 本文综述了多媒体查询语言,提出了 16 条关于查询语言查询表达能力的准则,涵盖了内容信息、时空信息和不确定信息 3 个方面。这些准则的作用有两点:(1)使读者更加清楚地了解现有的多媒体查询语言在查询表达能力方面的优缺点;(2)为研究者在后续的通用多媒体查询语言研究中提供一个清晰的需求。这些准则覆盖了多媒体蕴含信息的各方面,它们之间不存在不相容的问题。然后,根据这些准则对查询语言进行了评价。评价的结果显示,这些查询语言在基本的查询表达能力(如低层次的内容信息表达、时序关系、空间位置关系、拓扑关系)方面取得了一些成绩,可以满足用户的基本要求,但在高级语义查询方面还有所欠缺。而对于不确定查询,目前的查询语言还基本没有考虑。综合目前多媒体查询语言的发展现状和不足之处,未来的多媒体查询语言应该:(1)拥有很强的信息表达能力,能够很好地表达多媒体数据的各层次内容信息,支持时空查询和不确定查询。(2)具有简洁有效的语法形式,易于使用。逻辑性语言虽然具有良好的查询表达能力,但由于使用大量存储算子,导致其复杂难用;可视化查询语言提供易于使用的图形界面,但难以拥有强大的表达能力。所以,刚好介于两者中间的描述性文本语言成为了多媒体查询语言的主流。(3)具有良好的扩充性。随着多媒体应用的不断发展,新的多媒体数据类型不断涌现,这需要查询语言能够有良好的扩充性。

参 考 文 献

[1] Megalini C, Rabitli F. Conceptual modeling of multimedia documents. *IEEE Computer*, 1991, 24(10): 23-29

[2] Flicker M, Sawhney H. Query by image and video content: the QBIC system. *IEEE Computer*, 1995, 28(9): 22-31

[3] Gupta A, Weymouth T E, Jain R. Semantic queries with pictures; the VIMSYS model // *Proceedings of the 17th International Conference Very Large Databases*. Barcelona; Morgan Kaufmann Publishers, Inc., 1991: 69-79

[4] Wu J K. CORE: a content-based retrieval engine for multimedia information systems. *Multimedia Systems*, 1995, 3(1): 43-52

[5] Roussopoulos N, Faloutsos C, Sellis T. An efficient pictorial da-

tabase system for PSQL. *IEEE Transactions on Software Engineering*, 1988, 14(5): 639-650

[6] Ahad R, Basu A. ESQL: A query language for the relation model supporting image domains // *Proceedings of the 7th International Conference on Data Engineering*. IEEE Computer Society, 1991: 550-559

[7] Chu W W, Jeong I T, Taira R K. A semantic modeling approach for image retrieval by content. *The VLDB Journal*, 1994, 3: 445-477

[8] Nepal S, Ramakrishna M V, James A T. A fuzzy object query language (FOQL) for image databases // *Proceedings of 6th International Conference on Database Systems for Advanced Applications*. 1999: 117-124

[9] Li W S, Candan K S, Hirata K, et al. IFQ: A visual query interface and query generator for object-based media retrieval // *Proceedings of IEEE International Conference on Multimedia Computing and System*. IEEE Computer Society, 1997: 353-361

[10] 龚声蓉,熊璋,刘旭东. 基于内容的可视化多媒体查询语言的设计. *北京航空航天大学学报*, 1998, 4(24): 479-482

[11] Cardenas A F, Jeong I T, Taira R K, et al. The knowledge-based object-oriented PICQUERY+ language. *IEEE Transactions on Knowledge and Data Engineering*, 1993, 5(4): 644-657

[12] Aslandogan Y A, Their C, Yu C T, et al. Design, implementation and evaluation of SCORE: a system for content based retrieval of pictures // *Proceedings of the 11th International Conference on Data Engineering*. 1995: 280-287

[13] Oomoto E, Tanaka K. OVID: design and implementation of a video-object database system. *IEEE Transactions on Knowledge and Data Engineering*, 1993, 5(4): 629-643

[14] Choi Y-I, Park Y-M, Lee H-S, et al. An integrated data model and a query language for content-based retrieval of video // *Proceedings of 4th International Workshop on Advances in Multimedia Information Systems*. 1998: 192-198

[15] Adali S, Candan K S, Erol K, et al. AVIS: an advanced video information system // *Proceedings of the 1st International Workshop on Multimedia Information Systems*. 1995: 369-373

[16] Hwang E, Subrahmanian V S. Querying video libraries. *Journal of Visual Communication and Image Representation*, 1996, 7(1): 44-60

[17] Kuo T C T, Chen A L P. Content-based query processing for video databases. *IEEE Transactions on Multimedia*, 2000, 2(1): 1-13

[18] Donderler M E. Data modeling and querying for video databases. Rochester, New York; The College School of Engineering and Applied Sciences, University of Rochester, 2002

[19] Egenhofer M. Spatial SQL: A query and presentation language. *IEEE Transactions on Knowledge and Data Engineering*, 1994, 6(1): 86-95

[20] Arisawa H, Tomii T, Salev K. Design of multimedia database and a query language for video image data // *Proceedings of IEEE International Conference on Multimedia Computing and Systems*. IEEE Computer Society, 1996: 462-467

[21] Dimitrova N, Golshani F. EVA: A query language for multimedia information systems // *Proceedings of Multimedia Information Systems*. 1992

FT 的实验结果如图 10 所示。ckpt/rst 开销随着处理器数目增加而增大。用 512 处理器时, checkpoint 开销为 23.82%, ckpt/rst 开销为 29.3%。1 个故障时, 我们用 2 个进程并行恢复, 开销低于 10%。无故障时, PR FT 与 NPB FT 的执行时间几乎相同。

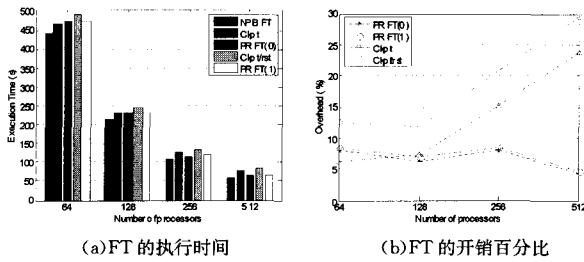


图 10 FT 的实验结果

5.2 PR 与 RPC 的性能比较

这组实验中 EP 的规模为 class B, 某个进程在主循环的正中间被杀掉。在 PRC 实验中, 每 120 次迭代做 1 次 checkpoint。RP EP 和 PRC EP 的执行时间和开销如图 11 所示。

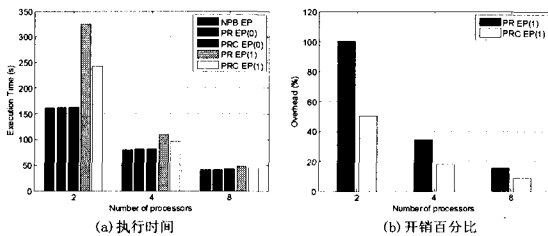


图 11 PR 和 RPC 的比较

从图 11 可知, 无故障时, PR EP 和 PRC EP 的开销都可以忽略不计。当发生 1 个故障时, RPC EP 的性能优于 RP EP。但是性能差异随着 N 增大而快速降低。因此, PRC 的性能优势与恢复过程的计算规模有关: 规模越大, 优势越明显。

结束语 本文的容错方法并行复算采用了与现有方法如 checkpointing 和 message-logging 不同的容错方式, 它使得故障恢复并行完成。我们介绍了如何用并行复算实现容错的程序, 还提供了并行复算的一种优化方法, 将 checkpointing 与之结合。实验结果表明, 不管有没有发生故障, 并行复算的开销百分比都低于 checkpointing, 并且随着处理器数目增大而减小; RPC 的性能则更优于并行复算, 这种性能优势与问题规模有关, 问题规模越大, 则优势越明显。

参考文献

[1] Engelman C, Geist A. A diskless checkpointing algorithm for super-scale architectures applied to the fast fourier transform//

CLADE 2003; 1st International Workshop on Challenges of Large Applications in Distributed Environments, Seattle, WA, USA, 2003

[2] IBM Research. Blue gene Project overview. Online at <http://www.research.ibm.com/bluegene/>, 2002

[3] Elnozahy E N, et al. A survey of rollback-recovery protocols in message-passing systems. *ACM Computing Surveys*, 2002, 34(3): 375-408

[4] Chandy K M, Lamport L. Distributed snapshots: Determining global states of distributed systems. *ACM Trans. Comput. Systems*, 1985, 3(1): 63-75

[5] Plank J S, Li K. Ickp—a consistent checkpointer for multicomputers. *IEEE Parallel Distrib. Technol.*, 1994, 2(2): 62-67

[6] Stellner G. CoCheck: Checkpointing and process migration for MPI//Proc. 0th International Parallel processing Symposium, 1996

[7] Elnozahy E N, Johnson D B, Zwaenepoel W. The performance of consistent checkpointing // Proc. 11th Symposium on Reliable Distributed Systems, 1992

[8] Elnozahy E N, Zwaenepoel W. On the use and implementation of message logging // 24th International Symposium on Fault-Tolerant Computing, 1994

[9] Strom E, Yemini S. Optimistic recovery in distributed systems. *ACM Transactions on Computer Systems*, 1985, 3(3): 204-226

[10] Louca S, et al. MPI-FT: Portable fault tolerance scheme for MPI. *Parallel processing Letters (PPL)*, 2000, 10(4)

[11] Bouteiller A, et al. MPICH-V project: A Multiprotocol Automatic Fault-tolerant MPI. *International Journal of High Performance Computing Applications*, 2006, 20(9): 319-333

[12] Plank J S, Li K, Puening M A. Diskless checkpointing. *IEEE Trans. Parallel Distrib. Syst.*, 1998, 9(10): 972-986

[13] Silva L M, Silva J G. An Experimental Study About Diskless Checkpointing//Proceedings of the 24th EUROMICRO Conference, 1998

[14] Plank J S, Kim Y, Dongarra J. Algorithm-based diskless checkpointing for fault tolerant matrix operations//25th International Symposium on Fault-Tolerant Computing, Pasadena, CA, 1995

[15] Beguelin A, Seligman E, Stephan P. Application level fault tolerance in heterogeneous networks of workstations. *Journal of Parallel and Distributed Computing*, 1997, 43(2): 147-155

[16] Chakravorty S, Kal L V. A Fault Tolerance protocol with Fast Fault Recovery // IPDPS2007; 21st IEEE International Parallel & Distributed processing Symposium, Long Beach, California USA, 2007

[17] Fagg G, Dongarra J. FT-MPI: Fault Tolerant MPI, Supporting Dynamic Applications in a Dynamic World//Lecture Notes in Computer Science; Proceedings of EuroPVM-MPI 2000. Hungary: Springer Verlag, 2000

(上接第 13 页)

[22] Marcus S, Subrahmanian V S. Foundations of multimedia database systems. *Journal of ACM*, 1996, 43(3): 474-523

[23] Kau S-C, Tseng J C R. MQL: a query language for multimedia database. *Multimedia Communications*, 1994: 511-516

[24] Li J Z, Ozsu M T, Szafron D, et al. MOQL: a multimedia object query language//Proceedings of the 3rd International Workshop on Multimedia Information Systems, 1997: 19-28

[25] 田增平, 党华锐, 周傲英. 多媒体对象查询语言及其查询处理. *软件学报*, 1999, 10(7): 694-701

[26] Cao Z S, Wu Z D, Wang Y Z. UMQL: A unified multimedia que-

ry language//Proceedings of the 2007 IEEE International Conference on Signal Image Technology and Internet Based Systems, IEEE Computer Society, 2007: 101-107

[27] Dionisio J D N, Cardenas A F. MQuery: a visual query language for multimedia, timeline, and simulation data. *Journal of Visual Language and Computing*, 1996, 7(4): 377-401

[28] Schmitt I, Schulz N, Herstel T. WS-QBE: a QBE-like query language for complex multimedia queries//Proceedings of the 11th International Conference on Multimedia Modeling, IEEE Computer Society, 2005: 222-229

[29] 吴宗大, 曹忠升, 王元珍. 可视化多媒体查询语言的设计与实现. *华中科技大学学报*, 2008, 7(36): 45-49