

基于着色时间 Petri 网的实时系统的形式验证^{*}

吴 琼¹ 邵志清¹ 刘 刚² 陈丽琼¹

(华东理工大学信息科学与工程学院 上海 200037)¹ (华南师范大学南海校区计算机工程系 佛山 528225)²

摘 要 嵌入式实时系统多数应用在安全性要求较高的场合,因此需要保证系统的正确性。复杂性不断增加的实时系统迫切需要在系统开发早期引入形式化分析技术来验证系统的期望性质。时间 Petri 网是有严格数学基础的图形表达工具,适合对实时系统建模;时间自动机(Timed Automata, TA)有成熟的验证工具,被广泛用于实时系统的模型检验和验证。本文提出一种基于着色时间 Petri 网(Colored Time Petri Net, CTPN)的实时系统的验证方法,用 CTPN 对带有控制流和数据流的实时系统建模,通过转换规则将 CTPN 模型转换成语义等价的 TA 模型,利用模型检验工具 UPPAAL 验证系统的性质。最后,用实例证明此方法有效。

关键词 着色时间 Petri 网,时间自动机,转换算法,模型检验

Formal Verification of Real-time System Based on Colored Time Petri Net

WU Qiong¹ SHAO Zhi-qing¹ LIU Gang² CHEN Li-qiong¹

(School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200037, China)¹

(Department of Computer Technology, Nanhai College, South China Normal University, Foshan 528225, China)²

Abstract Most Real-time Embedded Systems have been used in safety-critical situation, which is necessary to ensure the correctness of systems. The growing complexity of real-time embedded systems makes it imperative to apply formal analysis techniques at early stages of system development. Time Petri Net is a graphical formal method based on strict mathematics, which is suitable to model real-time systems. Timed Automata (TA), which has mature validation tools, has been widely used to verify and model check of real-time systems. This paper considers a verification method of real-time systems based on Colored Time Petri Nets (CTPN), using CTPN to accurately describe the systems both with control flow and data flow, the mapping rules from CTPN to TA that preserves the behavioral semantics (timed bisimilarity) of the initial CTPN is given, on the base of this, the dynamic behavior of system is analyzed by UPPAAL. Finally, an example is provided to demonstrate the modeling process and formal verification.

Keywords Colored time petri net, Timed automata, CTPN-to-TA, Model checking

1 引言

实时系统在现代生活中无处不在,并大量应用在安全性要求较高的场合。因此,需要有能保证系统正确性的高度精确的工具和技术^[1,3]。时间 Petri 网(Time Petri Net, TPN)和时间自动机(Timed Automata, TA)是被广泛用来建模和分析实时系统的形式化工具。目前的一种研究趋势是通过将 TPN 转换成 TA,利用 TA 成熟的验证工具来分析 TPN。

Lime 和 Roux^[1]扩展了状态类图结构,并用时间迁移系统的概念定义了 TPN 的形式语义,提出一种从 TPN 到 TA 的结构转换方法,并证明转换过程能够保持 TPN 的行为语义,即时间互模拟等价(timed bisimilarity)。这种方法依赖于 TPN 的状态图,而且仅适用于有限 TPN。Zonghua Gu^[2]等人研究了变迁上带有时间关联的 TPN 模型,并详细描述了从 TPN 到 TA 的转换规则,并用 UPPAAL 验证了航空系统的一个实例。作者没有区分描述模型的静态结构和行为语义,转换过程不够清晰。周清雷^[8]等人将一个描述实时系统的时延 Petri 网模型转换成与其语义等价的一组时间自动机模型,

并验证了系统的局部时间性。本文考虑了同时带有数据流和控制流的实时系统,用着色时间 Petri 网(Colored Time Petri Net, CTPN)对系统建模,再将 CTPN 模型从静态结构上转换成 TA,并保持 CTPN 行为语义的一致性,最后用 TA 验证工具 UPPAAL 验证系统的性质。

第 2 部分介绍着色时间 Petri 网模型和时间自动机概念;第 3 部分用着色时间 Petri 网作为实时系统的建模工具,重点介绍从 CTPN 到 TA 的转换规则,转换得到的结果是一组 TA,可以直接作为验证工具 UPPAAL 的输入模型;第 4 部分介绍 UPPAAL 的验证原理和使用方法;第 5 部分用门禁系统实例说明此方法的有效性;最后对全文总结。

2 相关概念和定义

定义 1 满足下列条件的六元组 $N = (P, T, F, C, M, \Omega)$ 称作着色时间 Petri 网^[7],

- (1) P 是一个非空有限的库所集合;
- (2) T 是一个非空有限的变迁集合;
- (3) $F: (P \times T) \cup (T \times P) \rightarrow 2^{N \times C}$ 是库所和变迁的流关系,

^{*} 本文受国家自然科学基金(No. 60373000)、国家 863 高技术研究发展计划(Nos. 863-317-01-04-99, 2001AA115126)基金资助。吴 琼 硕士研究生,主要研究领域为软件形式化、嵌入式软件设计;邵志清 教授,博士生导师,主要研究领域为软件开发与验证方法;刘 刚 博士,主要研究领域为软件形式化、嵌入式软件设计;陈丽琼 博士研究生,主要研究领域为软件形式化、Petri 网应用、嵌入式软件设计。

其中 N 是一个非负整数集,网中的每个有向边与一个整数-着色对 $\{(k,c) | k \in N, c \in C\}$ 相对应;

(4) $M: P \rightarrow 2^{N \times C}$ 是标识,表示 N 中各个库所中着色令牌的分配状况; M_0 表示系统的初始标识;

(5) $\Omega: T \rightarrow N \times (N \cup \infty)$ 是变迁的相对触发间隔, $\Omega(t) = (\alpha, \beta)$ 。其中 $t \in T$, α, β 分别表示变迁 t 的最早触发时间和最迟触发时间。

定义 2 设 $N = (P, T, F, C, M, \Omega)$ 是一个着色时间 Petri 网。对于 $x \in P \cup T$, 记

$$^*x = \{y | y \in P \cup T \wedge (y, x) \in F\}$$

$$x^* = \{y | y \in P \cup T \wedge (x, y) \in F\}$$

称 *x 为 x 的输入集, x^* 为 x 的输出集。

定义 3 $N = (P, T, F, C, M, \Omega)$ 是一个着色时间 Petri 网, 如果 $\forall p \in P: p \in ^*t_i \rightarrow M(p) \geq F(p, t_i)$, 则称 t_i 在标识 M 下有发生权, (记作 $M[t_i >]$), 称 $ET(M) = \{t_i | t_i \in T \wedge M[t_i >]\}$ 为标识 M 的有发生权的变迁集。

定义 4 设 $N = (P, T, F, C, M, \Omega)$ 为着色时间 Petri 网, 满足下列条件的二元组 $S = (M, I)$ 称为 N 的状态:

(1) M 为 N 的一个标识;

(2) $\forall t_i \in ET(M), I(t_i) = [\alpha_i, \beta_i]$ 是在标识 M 下有发生权的变迁的有效触发时间。

CTPN 保留了 TPN 作为一种直观的图形建模工具和一种具有丰富数学基础的形式化模型的优点, 非常适合描述具有并发、异步和分布式特征的系统, 同时加强了对系统的不同资源的描述能力。图 1 描述了一个 CTPN 模型, 其中库所中的圆点表示令牌, 圆点的不同着色代表着系统的不同资源, 而标注在有向边上的整数-着色对表示由 F 定义的权值。本文中的两种着色分别代表数据资源和控制资源。

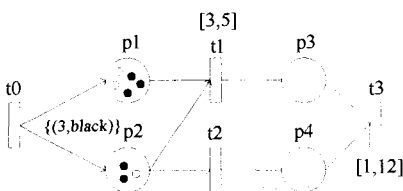


图 1 着色时间 Petri 网

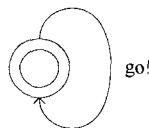


图 2 Dummy 自动机

定义 5 满足下列条件的六元组 $A = (L, l_0, \Sigma, X, I, E)$ 称为时间自动机, 其中:

(1) L 是有限模式集合;

(2) $l_0 \in L$ 是初始模式;

(3) Σ 是有限同步标签集合;

(4) X 是时间变量集合;

(5) $I: L \rightarrow C(X)$ 是从模式到时间限制的映射函数;

(6) $E \subseteq L \times C(X) \times \Sigma \times \rho(X) \times L$ 是变迁的集合, $\rho(X)$ 是该变迁要重置的时钟的集合。

定义 6 设 $A = (L, l_0, \Sigma, X, I, E)$ 是时间自动机, 系统状态 $S = \{(l, v) | l \in L, v \text{ 满足 } I(l)\}$, 其中 $v: X \rightarrow \mathbb{R}^{\geq 0}$ 是 X 上的时钟解释。

定义 7 设 $A = (L, l_0, \Sigma, X, I, E)$ 是时间自动机, $(l_1, v), (l_2, w) \in S, a \in \Sigma, (l_1, v) \xrightarrow{a} (l_2, w)$ 为系统变迁, 当且仅当: $\exists (l_1, g, a, \gamma, l_2) \in E \Rightarrow v|_{\gamma} = g, w|_{\gamma} = I(l_2), w = v[\gamma := 0]$

时间自动机可以被看作是添加了时钟变量与时间约束的有穷自动机, 时间自动机的操作语义定义了由当前位置和当前时钟值所组成的状态之间的转换。研究表明, 用时间计算

树逻辑 (Timed Computation Tree Logic, TCTL) 描述的系统性质的模型检验对于时间自动机或某些扩展时间自动机而言是可判定的, 而能够对时间自动机有效进行模型检验的工具 (如 UPPAAL, KRONOS 和 CMC) 也随之产生, 它们能够较好地描述、验证实时工业用例。

3 CTPN 向 TA 转换

TPN 和 TA 非常相似, 都能描述系统的动态变化。我们认为某种程度上 TA 比 TPN 的表达能力更强, 因为 TA 是更低层次的形式化方法^[5]。考虑将 CTPN 模型转换成语义等价的 TA 模型, 不仅能进一步明确系统的时间性质, 还可以利用已有的模型检验器来分析 CTPN。从 CTPN 到 TA 的转换规则, 需要考虑模型的静态语法和动态语义两方面, 将 CTPN 模型从结构上 (即模型的语法) 转换成 TA 模型的过程中, 要保持 CTPN 对系统动态行为 (即模型的语义) 的描述能力。转换过程如下:

1) 对 CTPN 每个库所 $p_i \in P$, 在 TA 模型中申明两个全局整型变量 p_i^1, p_i^2 , 对应不同着色令牌的数目。如: $p_1^1 = 1, p_1^2 = 0$, 表明库所 p_1 中有 1 个控制令牌 (黑色), 没有数据令牌 (白色)。全局变量 p_i^1, p_i^2 在所有自动机之间共享。

2) 为 CTPN 中的每个变迁 t_i 定义一个时间自动机模板 A_i , 每个时间自动机有个局部的时钟变量 x_i 。每个时间自动机有两个状态 (location), 分别为 enabled 和 disabled, 默认 disabled 为终止态。

3) 为每个变迁上的有效触发间隔 $[\alpha, \beta]$ 定义局部整型变量 lb, ub 。局部整型变量仅在时间自动机内部有效。

4) 对 CTPN 中的流关系 $F \subseteq P \times T$, 从 disabled 向 enabled 添加一条边, 对流关系上的整数-着色对 (k, c) , 定义两个局部整型变量 in_wgt_w, in_wgt_b ; 对 $F \subseteq T \times P$, 从 enabled 向 disabled 添加一条边, 对流关系上的整数-着色对 (k, c) , 定义两个局部整型变量 out_wgt_w, out_wgt_b 。

5) 声明一个全局紧急通道 go , 各时间自动机之间利用该通道通信。构造 Dummy 自动机, 如图 2 所示。Dummy 用来协调各自动机的行为, Dummy 向各自动机的终止状态发送 $go!$, 自动机接收到 $go?$, 满足条件的状态转移可以无延迟地发生。

6) 每个自动机的初始状态取决于原始 Petri 网中的初始标识 M_0 。如果 $M_0 >^* t$, 则初始状态是 enabled, 否则是 disabled。

7) 假设一个 CTPN 变迁 $t \in T$ 有相应的相对触发时间 $[lb, ub]$, k 个输入库所 $p_1^i, \dots, p_k^i$, m 个输出库所 $p_1^o, \dots, p_m^o$ 。

a) 在 enabled Location 上添加一个不变的条件 $c \leq ub$ 。

b) 在从 disabled 指向 enabled 的边上增加同步标签 $go?$, 使能条件为 $M(p_1) \geq F(p_1, t_i), \dots, M(p_k) \geq F(p_k, t_i)$, 赋值标签 $x_i := 0$ 。表示当库所包含的着色令牌数大于触发变迁所需的着色令牌数时, 变迁能够发生。

c) 在从 enabled 指向 disabled 的边上增加使能条件为 $M(p_1) \geq F(p_1, t_i), \dots, M(p_k) \geq F(p_k, t_i), c \geq lb$, 赋值标签为 $M(p_1^i) := M(p_1^i) - F(p_1^i, t_i), \dots, M(p_k^i) := M(p_k^i) - F(p_k^i, t_i), M(p_1^o) := M(p_1^o) + F(t_i, p_1^o), \dots, M(p_m^o) := M(p_m^o) + F(t_i, p_m^o)$ 。表示变迁触发后, 系统状态发生改变。

按照上述规则, 在系统配置阶段, 为每个 CTPN 转换具体生成一个自动机实例, 将参数以列表的形式作为适当的全局变量, 代表转换中的输入、输出库所。以一个输入库所、一

个输出库所的 TPN 变迁为例,参数列表为 List(int inw,inb, outw,outb; const int in_wgt_w,in_wgt_b,out_wgt_w,out_wgt_b; const int lb,ub),转换后的 TA 如图 3 所示。

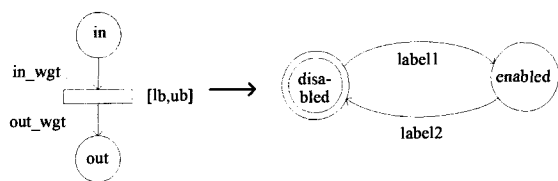


图 3 从 CTPN 到 TA

其中, $label(inw \geq in_wgt_w; inb \geq in_wgt_b; go?; x:=0)$, $label2(inw \geq in_wgt_w; inb \geq in_wgt_b; x \geq lb; intw := intw - in_wgt_w; intb := intb - in_wgt_b; outw := outw + out_wgt_w; outb := outb + out_wgt_b)$ 。

4 CTL 模型检验

4.1 CTL 公式

时态逻辑表示系统各个事件的发生在时间上的次序,给定一个描述系统模型的状态变迁系统 S 和一个描述系统性质 ϕ 的时态逻辑公式 F ,这样系统是否具有所期望的性质就转化为数学问题: $S \models F?$ [4]。对于有穷状态变迁系统来说,这个问题是可判定的。实时系统的模型检验可判定的一个重要原因是无穷多个时钟解释可以被划分为有穷多个域,属于同一个域的时钟解释满足同样的性质。

一个系统的运行可以看成是系统状态的变化,计算树逻辑(Computation Tree Logic,CTL)可以描述系统状态的前后关系和分歧情况。CTL 公式由原子命题、逻辑连接符和模态算子组成。CTL 的逻辑连接符包括:CTL 的逻辑连接符包括: $\neg$ (非), $\vee$ (或), $\wedge$ (与);CTL 的模态算子包括: E (Exist), A (Always), U (Until), X (Next-time), F (Future), G (Global)。

定义 8(CTL 公式的语法表示)

$$\phi ::= \eta \mid \exists \square \phi' \mid \exists \phi' \mu \sim c \phi'' \mid \neg \phi' \mid \phi' \vee \phi''$$

其中, η 是 $B(\cup C_{ij}, \cup D_{ij}, \cup Y_{ij})$ 的一个时序谓词, ϕ' 和 ϕ'' 是 CTL 公式, $\sim \in \{<, \leq, =, \geq, >\}$, 且 $c \in N$ 。 $\exists \square \phi'$ 表示从现在状态开始,存在一个计算 ϕ' 永远为真。 $\exists \phi' \mu \sim c \phi''$ 表示从现在状态开始,存在一个计算 ϕ' 在时间约束 $\sim c$ 下当 ϕ'' 为真时才为真。

下面给出一些常用的表示系统性质的 CTL 公式:

1) $AG\phi$ 表示的是系统的不变特性。该公式通常表示“坏

的事情永远不会发生”。

2) $AF(\phi)$ 表示从初始状态最终总能进入到 ϕ 为真的状态,即表示“好的事情总会发生”。

3) $AG(\phi \rightarrow AF\phi')$ 表示从所有 ϕ 为真的状态,最终总能够到达 ϕ' 为真的状态,它也可以表示在任何状态只要有请求,则系统最终能响应的请求。

4.2 UPPAAL 工具

UPPAAL 是由瑞典的 Uppsala 大学与丹麦的 Aalborg 大学联合研发的一种自动验证工具,主要对实时系统进行模拟、仿真和验证,已成功用于实时控制器和通信协议的验证 [6,8]。UPPAAL 主要通过快速搜索机制来验证时钟约束和可达性。同其他工具相比,UPPAAL 的主要优点是高效性和实用性。2007 年 2 月 12 日,UPPAAL 推出了新版本 4.05,功能更加强大,本文应用的是 3.4 版。

UPPAAL 将系统描述为非确定的并行过程的积,每一个过程描述为由有限控制结构、实数值时钟和变量组成的时间自动机,过程之间通过管道和共享变量来进行通讯,管道用于保证不同自动机中的两个转换同时执行。UPPAAL 的用户界面包括 3 个部分:系统编辑器(System Editor)、模拟器(Simulator)和模型检验器(Model-Checker)。系统编辑器用于创建和编辑待分析的系统,UPPAAL 中使用的系统模型是时间自动机。模拟器可以模拟不同实体之间通过管道相互通信、控制的消息序列,初步检查所建系统模型可能的执行是否有错,以此在验证前发现一些错误。模型检验器利用需求规范语法对系统做进一步检验,通过快速搜索系统的状态空间来检查时钟约束和反应约束性质,并为系统要求的规范和文件提供了一个需求规范编辑器。

5 实例

5.1 系统的 CTPN 模型

我们采用的例子是一个门禁系统,使用者可以透过键盘输入账号,只要使用者输入正确的账号密码就可以开启大门;反之如果使用者连续输入错误三次,就会有警铃响起。警报进行一定时间后,自动停止。用 CTPN 对系统建模如图 4 所示,库所中的白色圆点表示数据令牌,黑色圆点表示控制令牌。为了使图形简明清晰,仅在图中标出了变迁产生和消耗的数据令牌数目。默认每个变迁发生需要消耗一个控制令牌。

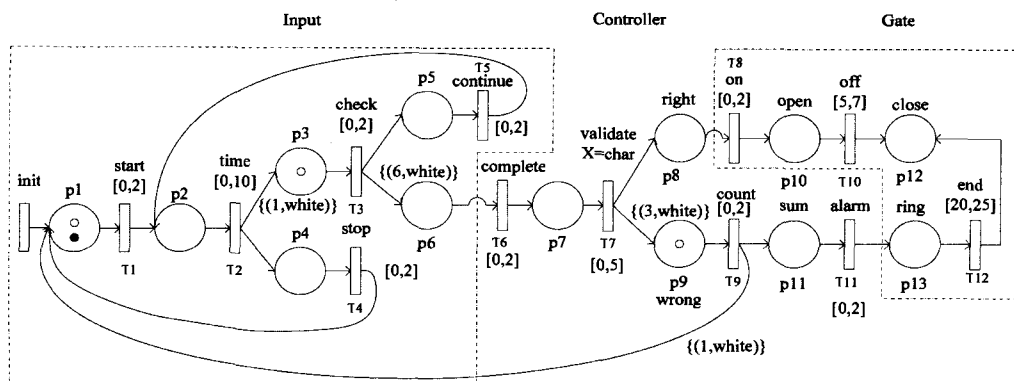


图 4 门禁系统

5.2 安全性验证

门卫系统必须满足以下两个性质:1)安全性,当密码不正

确时,门总是处于关闭状态;2)受限活性,即在门打开后的一段时间 Δt 中,门必须关上。用 CTL 描述上述性质为:

1) $A[\Box \rightarrow (\text{controller_wrong} \wedge \text{gate_open})]$

2) $A[\Box (\text{gate_open} \Rightarrow \text{gate_close}, c \leq 8)]$

为了用 UPPAAL 模型检验工具验证系统性质,需要将上述 CTPN 转换成 TA。按照上述方法,将变迁转换成 TA 模板,并将实际参数列表添加到过程说明中。为每个变迁声明一个局部时钟,各个时钟均以相同的速度计时。以安全性为例,对 T7 和 T8 进行转换,如图 5 所示。

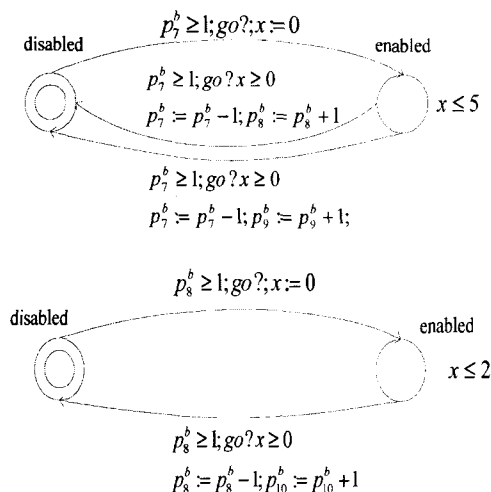


图 5 T7, T8 的 TA 模板

将 A7 和 A8 作为输入模型,用 UPPAAL 验证 $A[\Box (p_7^b = 1 \rightarrow p_{10}^b = 0)]$, 结果满足安全性。

结束语 本文对实时系统软件的建模和性质验证进行了研究。实例证明,着色时间 Petri 网能够很好地描述实时系统软件的领域特征,将 CTPN 模型转换成 TA 模型后,可以利

用已有的模型验证工具(如 UPPAAL)来分析 CTPN。目前模型检验器多使用由自动机组成的平面网作为系统模型,不易于模拟和调试大规模的工业系统。层次时间自动机(Hierarchical Timed Automata)模型中的每一个状态都可以被描述为另一个时间自动机,更接近实际系统。在本文工作基础上,下一步的工作主要包括:进一步完善此方法,增强 CTPN 的描述能力;面向工业应用,考虑将 CTPN 模型转换成层次自动机模型。

参考文献

- [1] Cassez F, Roux O H. Structural translation from Time Petri Nets to Timed Automata- Model-Checking Time Petri Nets via Timed Automata [J]. The journal of Systems and Software, 2006, 79(10): 1456-1468
- [2] Gu Zonghua, Shin K G. Analysis of Event-Driven Real-Time Systems with Time Petri Nets; A Translation-Based Approach [C]. Norwell, MA: Kluwer Academic Publishers, 2002, 31(40): 31-40
- [3] Schmidt D D. MDE4DRE: Model-Driven Engineering for Distributed Real-time and Embedded Systems [C] // 13th IEEE Real-Time and Embedded Technology and Applications Symposium. Washington, 2007
- [4] Boucheneb H, Hadjidi R. CTL model checking for time Petri nets [J]. Theoretical Computer Science, 2006, 353(1): 208-227
- [5] Berard B, Cassez F, Haddad S, et al. Comparison of the Expressiveness of Timed Automata and Time Petri Nets [C] // 3rd International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS 05). volume 3829 of Lecture Notes in Computer Science. Uppsala, Sweden, Springer, September 2005
- [6] Boniol F, Wiels V, Ledinot E. Experiences in using model checking to verify real time properties of a landing gear control system [C]. 3rd European Congress Embedded Real Time Software, ERTS, 2006
- [7] Dotoli M, Fanti M P. A colored Petri net model for automated storage and retrieval systems serviced by rail-guided vehicles; a control perspective [J]. International Journal of Computer Integrated Manufacturing, 2005, 18(2/3): 122-136
- [8] 周清雷, 王静. 基于 Uppaal 的时延 Petri 网到时间自动机等价模型验证 [J]. 计算机应用研究, 2005, 22(6): 64-66

(上接第 251 页)

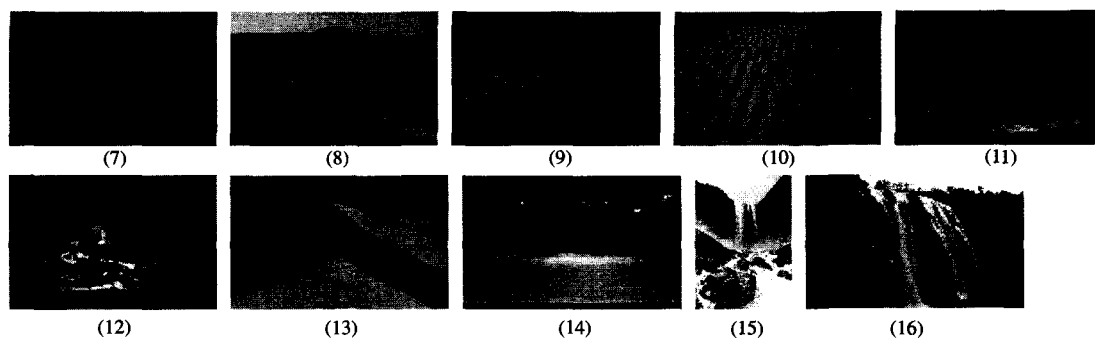


图 5 对图 4 中图像的检索结果

参考文献

- [1] Niblack W, Barber R, Equitz W, et al. The qbic project: Querying images by content using color, texture, and shape // Proceedings of the SPIE Conference on Storage and Retrieval for Image and Video Databases. San Jose, CA, 1993: 173-187
- [2] Pentland A, Picard R W, Sclaroff S. Photobook: Content-based manipulation of image databases. International Journal of Computer Vision, 1996, 18(6): 233-254
- [3] Zhang Yue, He Xing-Jian. Texture feature-based Image classification using wavelet package transform [J]. Lecture Notes in Computer Science, Springer-Verlag, LNCS 3644, 2005: 165-173
- [4] Han Jun-Hua, Huang D S. A novel BP-based image retrieval system [J] // The IEEE International Symposium on Circuits and Systems. Kobe, Japan, 2005: 1557-1560
- [5] Equitz W H. A new vector quantization clustering algorithm [J]. IEEE Transactions Acoustic, Speech, Signal Process, 1989, 37
- [6] 容观澳. 计算机图像处理 [M]. 北京: 清华大学出版社, 2002
- [7] Hu M K. Visual pattern recognition by moment invariants [J]. IRE Transactions on Information Theory, 1962, 8
- [8] Attneave F, Arnoult M D. The quantitative study of shape and pattern perception [J]. Psychological Bulletin, 1956, 53
- [9] Tversky A. Feature of similarity. Psychological Review, 1977