

一种高稳定性低延迟的应用层组播生成树算法

崔建群¹ 陈爱玲¹ 夏振厂¹ 吴黎兵²

(华中师范大学计算机学院 武汉 430079)¹ (武汉大学计算机学院 武汉 430079)²

摘要 由于应用层组播技术依靠终端主机转发组播数据,任意中间节点的退出都将造成系统的稳定性问题。同时,应用层组播技术对延时有严格的要求。为了提高应用层组播系统的稳定性和数据传输效率,根据影响应用层组播稳定性和延时的因素,抽象出基于节点稳定概率的度约束的最小延时应用层组播生成树问题模型 SDMD (Spanning tree based on stability probability, degree-constrained, and minimum diameter for ALM),并且证明了该问题属于 NP-hard 问题。为了解决该问题,给出了基于节点时间增益因子的 TG-S 近似算法。仿真实验表明, TG-S 算法生成的组播树在平均延时、最大延时和累积中断次数等方面有明显优势。

关键词 应用层组播,稳定性,最小延时, NP-hard, 时间增益因子

中图分类号 TP393 文献标识码 A DOI 10.11896/j.issn.1002-137X.2016.6.016

High Stability Low Delay Spanning Tree Algorithm for Application Layer Multicast

CUI Jian-qun¹ CHEN Ai-ling¹ XIA Zhen-chang¹ WU Li-bing²

(School of Computer Science, Central China Normal University, Wuhan 430079, China)¹

(School of Computer Science, Wuhan University, Wuhan 430079, China)²

Abstract Since application layer multicast(ALM) relies on terminal hosts to forward multicast data, any intermediate node fails or quits would result in the system stability problem. Meanwhile, ALM has strict requirement for multicast tree delay. To improve stability and data transmission efficiency, this paper introduced a Spanning tree problem model SDMD based on stability probability, degree-constrained, and minimum diameter for ALM, according to the influencing factor of ALM stability and delay. The SDMD problem was proved to be NP-hard and an approximation algorithm TG-S was proposed to solve it. The simulation result demonstrates the advantages of TG-S algorithm in average receiving delay, multicast tree delay and accumulative interrupt times.

Keywords Application layer multicast, Stability, Minimum delay, NP-hard, Time gaining

1 引言

组播^[1] (multicast) 是一种高效的信息分发方式, 目前被广泛地应用于网络视频会议、远程教育、多方在线游戏、媒体广播等系统^[2]。与单播相比, 组播通过借助通信设备的复制转发功能, 能够有效降低服务器的负载, 节省网络主干的带宽资源, 实现数据的高效并发传输。

IP 组播^[3] 是最早提出的组播机制, 其特点是路由算法在网络层实现核心功能, 由组播路由器实现数据转发和控制功能。IP 组播具有较高的数据传输效率和网络带宽利用率, 是一种理想的组播方式, 但 IP 组播改变了 Internet 基于单播的设计原则, 需要改变现有的网络设施, 并且存在着众多问题^[4-7], 例如难以提供有效的用户认证和计费手段、缺乏成熟的商业化基础等, 致使其在全球范围内的部署进展缓慢。

由于 IP 组播所存在的上述缺陷限制了其自身的发展, Francis^[8] 和 Zhang^[9] 在 1997 年和 1998 年各自独立地提出了

应用层组播 (Application Layer Multicast, ALM) 的思想, 其特点^[1] 是由终端系统代替路由器实现组播数据的复制转发功能, 且不必改变现有的网络设施, 不需要特殊设备的支持, 易于部署。目前, 已经有不少应用层组播协议得到了成功的运用, 例如 Yoid^[8]、Narada^[9]、HMTP^[10]、NICE^[11]。

但应用层组播同样也存在一系列的问题, 其中最为突出的就是应用层组播的稳定性问题^[12-14]。在应用层组播中, 当作为中间节点的主机退出或者失效时, 该节点的子孙节点都将中断与组播树的数据传输, 这种现象被称作为由节点的动态行为^[2] 所引起的应用层组播的稳定性问题。如图 1 所示, 节点 B 的退出或失效将导致其子孙节点 D、E、H、I、J、K 中断与组播树的连接, 导致这些节点均需要重新加入到组播树中以获取组播数据。如果在某次组播会话中频繁地出现中间节点的退出或失效情况, 将导致大量节点重新加入组播树的行为, 这不仅会降低用户体验, 还会降低组播的效率。同时, 应用层组播技术对网络延时有着严格的限制, 因此如何生成高

到稿日期: 2015-06-23 返修日期: 2015-09-02 本文受国家自然科学基金面上项目 (61170017, 61272112, 61370108), 湖北省科技支撑计划 (2013BAA004) 资助。

崔建群 (1974-), 女, 博士, 教授, 主要研究方向为网络服务质量、网络管理、高性能计算和应用层组播, E-mail: jqcui@126.com; 陈爱玲 (1990-), 女, 硕士生, 主要研究方向为应用层组播; 夏振厂 (1987-), 男, 硕士生, 主要研究方向为应用层组播; 吴黎兵 (1972-), 男, 博士, 教授, 博士生导师, 主要研究方向为计算机网络、网格计算和物联网技术。

稳定性、低延时的应用层组播树是迫在眉睫的关键问题。

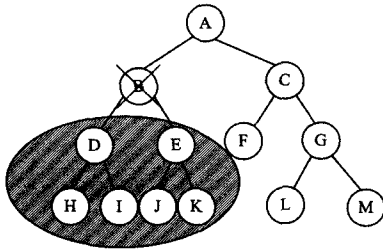


图1 应用层组播稳定性问题示意图

针对应用层组播对稳定性和延时的要求,提出了一种高稳定性、低延时的应用层组播树生成方案。仿真实验表明,该方案在提高组播树的稳定性以及降低组播延时方面有较明显的优势。

本文第2节分析了当下对应用层组播稳定性问题的相关研究;第3节给出了节点的稳定概率的计算方法;第4节介绍了本文提出的SDMD问题模型;第5节给出了对SDMD问题模型的解决算法;第6节对基于SDMD模型的算法进行仿真实验,并对实验数据进行分析;最后总结全文并进行展望。

2 相关工作

罗建光等人^[15]通过对超过1000万条实际视频直播系统的真实用户记录进行统计分析,验证了文献^[16]中得到的“用户的在线时间符合对数正态分布”的结论,同时通过对用户在线时间的进一步统计分析,指出用户平均剩余在线时间随着已在线时间的增大而增大;构建组播树时通过为新加入节点选择在线时间长的节点作为其父节点来降低组播树的中断频率,并据此提出了一种适用于视频直播的应用层组播树生成算法。但该文没有考虑节点之间的通信延迟以及节点的各个祖先节点的不稳定性的累积问题。

曹佳等人^[17]提出了一种度约束和点、边均带权的最小延时应用层组播生成树问题模型DCMD;并针对该问题模型,提出了两种启发式近似算法:基于度的算法和基于最大延迟路径的算法。但该文同样没有考虑组播树的稳定性问题。

本文针对高稳定性、低延时应用层组播树的生成,抽象出一个更加全面的问题模型:基于节点稳定概率的度约束的最小延时应用层组播生成树问题SDMD(Spanning tree based on stability probability, degree-constrained, and minimum diameter for ALM),该模型同时考虑边权、点权、节点出度约束这3方面的问题。本文所做的工作就是定义这3个因素约束的问题模型,并给出解决该问题的算法。

3 节点的稳定概率计算方法

3.1 相对离开概率的评估

节点的在线时间呈现对数正态分布^[16]。假设该分布的均值和均方差分别为 μ 和 σ ,则其概率密度函数 $f(t)$ 和累计分布函数 $F(t)$ 分别如式(1)和式(2)所示,其中 $t > 0$ 。

$$f(t) = \frac{1}{t\sigma\sqrt{2\pi}} e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}} \quad (1)$$

$$F(t) = \frac{1}{t\sigma\sqrt{2\pi}} \int_0^t e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}} \quad (2)$$

利用节点的在线时间评估其相对离开概率的基本思想是

将时间轴划分为多个间隔相等的区间,即 $[t_0, t_1), [t_1, t_2), [t_2, t_3), \dots, [t_{n-1}, t_n), [t_n, t_{n+1}), \dots$,其中 $t_0 = 0$,称区间长度 $t_{i+1} - t_i (i \geq 0)$ 为相对离开概率的更新周期 ΔT 。假设节点 v_i 的当前在线时间为 t ,且 $t \in [t_i, t_{i+1}) (i \geq 0)$,则此刻该节点 v_i 的相对离开概率为:

$$P_{rl}(v_i, t) = F(t_{i+1}) - F(t_i) \quad (3)$$

当更新周期 ΔT 为不同数值,且 $\mu = 5.19, \sigma = 1.44$ 时,相对离开概率的评估结果如表1所列。

表1 相对离开概率评估结果

ΔT (s)	$P(v_i, t) \cdot 10^{-4}$						
	$[t_0, t_1)$	$[t_1, t_2)$	$[t_2, t_3)$	$[t_3, t_4)$	$[t_4, t_5)$	$[t_5, t_6)$	$[t_6, t_7)$
120	3899	1901	1056	672	463	336	254
360	6856	1471	610	322	194	127	87
600	7990	1075	388	188	106	66	44
1200	9065	576	172	74	38	22	14

3.2 节点的稳定概率

文献^[15]仅考虑了当前节点的各级祖先节点中在线时间最长的节点对当前节点的影响。节点的稳定性是节点对祖先节点的稳定性的一种累积的感知,能够反映其祖先节点对其影响的基础指标。累积的节点稳定性比特定的祖先节点的稳定性对节点更有实际意义。

定义1 节点 v_i 的受影响概率^[2]如式(4)所示,其中, $ance(v_i)$ 指的是节点 v_i 的祖先节点。

$$P_{affect}(v_i, t) = \sum_{v_j \in ance(v_i)} P_{rl}(v_j, t) \quad (4)$$

定义2 节点 v_i 的稳定概率定义如式(5)所示,其中 $P_{affected}(v_i)$ 是节点 v_i 的受影响概率。

$$SP(v_i, t) = 1 - P_{affected}(v_i, t) \quad (5)$$

4 问题模型

4.1 SDMD问题

对于应用层组播树,若仅以优化组播树的稳定性为目标,可能导致算法生成延时较大的组播树,这显然不能满足应用层组播技术实时化的要求。现将求解基于稳定概率的度约束最小延时应用层组播生成树问题SDMD定义如下。

定义3(SDMD问题) 给定完全无向图 $G(V, E)$,对于任意的节点 $v_i \in V, SP(v_i)$ 表示节点的稳定概率, $deg(v_i) \in N^+$ 表示节点 v_i 的度, $deg_{max}(v_i) \in N^+$ 表示节点的最大度限制,对于任意的边 $e(v_i, v_j) \in E$,其中 $v_i, v_j \in V (i \neq j), d(e(v_i, v_j))$ 表示该边的传输延时。求满足以下条件的生成树 T : 1)节点 $s \in V$ 为组播树 T 的源节点,且 V 中所有的节点都在树 T 中; 2) $deg(v_i) \leq deg_{max}(v_i)$; 3)对于任意满足条件1)、2)的树 T' ,使得 $delay(T') \leq delay(T)$,其中 $delay(T)$ 表示应用层组播树的延时。

4.2 问题难度分析

SDMD问题可以规约至具有NP-complete难度的度受限最小延时生成树问题MDDL(Minimum Diameter, Degree-limited Spanning tree Problem)^[17]。MDDL问题描述这样一类问题:给定一个完全图 $G(V, E), deg_{max}(v_i) \in N^+$ 表示节点 v_i 的最大度限制,对于任意的边 $e(v_i, v_j) \in E$,其中 $v_i, v_j \in V (i \neq j), d(e(v_i, v_j))$ 表示该边的传输延时,求满足最大度限制的最小延时生成树。

定理1 SDMD问题是NP-hard问题。

证明:已知 $MDDL \in NP\text{-complete}$, 为了证明 $SDMD \in NP\text{-hard}$, 只需证明 $MDDL \in SDMD$. 对于任意的实例 $I \in MDDL$, 经过一个多项式时间变换节点 v_i 的稳定概率 $SP(v_i) = 0$, 从而得到实例 $I' \in SDMD$. 其次求解 I' 的满足度约束的最小延时生成树 T' , 显然, T' 即为求解 $MDDL$ 得到的最小延时生成树 T . 故得证, 即 $SDMD \in NP\text{-hard}$.

精确解决 $SDMD$ 问题是困难的, 确切算法基于枚举, 时间复杂度是指数级别的, 不适合实际应用. 近似算法能够在可接受的多项式时间内求出满足性能的近似解, 具有实际意义. 下面将给出求解 $SDMD$ 问题的近似算法: 基于节点稳定概率和时间增益因子的低延时算法即 $TG\text{-S}$ 算法.

5 算法描述

基于节点稳定概率和时间增益因子的应用层组播树生成算法的基本思想是: 寻找稳定度较高、到源节点延时较小的节点作为上游节点进行转发工作, 以达到生成一棵高稳定性低延时的组播树的目的; 而将稳定性较小、到源节点延时较大的节点放在组播树的下游, 以尽量减小其对下游节点的影响.

定义 4 (时间增益因子) 将节点 v_i 的时间增益因子定义为该节点的在线时间与该节点至源节点的延时的比值, 如式 (6) 所示.

$$TG(v_i) = \frac{t_{on\text{-}session}(v_i)}{\sum_{j \in PE_i} e_j} \quad (6)$$

其中, $t_{on\text{-}session}(v_i)$ 表示节点 v_i 的在线时间, $\sum_{j \in PE_i} e_j$ 表示节点 v_i 至源节点 s 路径上各边的延时之和, 即节点 v_i 至源节点 s 的延时.

在应用层组播中, 若节点的在线时间相同, 则由与组播源节点延时较小的节点进行转发可以降低整个组播的延时; 若节点与组播源节点的延时相同, 则由具有较大的在线时间的节点进行转发会提高整个组播树的稳定性. 因此, 定义时间增益因子与节点的在线时间成正比、与节点至源节点的延时成反比是合理的. 在应用层组播中, 应当尽量将时间增益因子较大的节点放在组播树的高层, 以获得较小的延时和较高的稳定性.

5.1 $TG\text{-S}$ 算法

将覆盖多播的网络模型抽象成完全图 $G(V, E)$, 其中 $v_i \in V$ 是待加入组播树的节点构成的集合, s 是组播源节点, 待求树 $T(V, E')$, 其中 $E' \subseteq E$.

算法中所使用到的符号及其含意如下.

V_T : 已加入组播树的节点构成的集合.

$V_{T_candidateParent}$: V_T 中有剩余出度的节点构成的集合.

\overline{V}_T : 还未加入到组播树中的节点构成的集合.

$OE(v_i)$ (Optional Edge): 一条边的两个顶点分别在集合 V_T 和集合 \overline{V}_T 中的边构成的集合, 其中在集合 V_T 中的顶点为 v_i .

$DOE(v_i)$ (Degree-free Optional Edge): $DOE \subseteq OE$, 且 v_i 有剩余出度.

算法描述如下.

输入: 无向连通图 $G(V, E)$; 节点的属性, 即最大出度 $deg_{max}(v_i)$ 、在线时间 $t_{on\text{-}session}(v_i)$ 、边权值 $c(e(v_i, v_j))$.

输出: 基于累积影响评价和时间增益因子的应用层组播

树 $T(V, E')$.

初始化应用层组播树, 将组播源节点 s 纳入 V_T 中.

① 计算出集合 $V_{T_candidateParent}$ 中稳定概率最高的节点 v_i 作为父节点.

② 根据 v_i 生成相应的 $DOE(v_i)$ 集合, 即获得 v_i 到达各待加入节点的延时, 并计算出 \overline{V}_T 中的节点通过相应的 DOE 的 TG 值, 将 $DOE(v_i)$ 中的边按照 TG 值降序排列并纳入至集合 $DOET$ 中.

③ 从集合 $DOET$ 中按照 TG 值由大至小选择 α ($\alpha > 1$) 个节点纳入到待加入节点集合 $DOES$.

④ 从 $DOES$ 中选择与源节点 s 延时最小的节点 v_j , 将 v_j 及相应的边 $e(v_i, v_j)$ 加入到 T 中, v_i 的出度减 1.

⑤ 更新集合 V_T 和 \overline{V}_T .

重复步骤①—⑤, 直至 $V_T = V$.

步骤③中, α 的取值可以根据情况进行选择. 当 α 较大时, 侧重选择与源节点延时较小的节点; 当 α 较小时, 侧重选择 TG 值较大的节点.

$TG\text{-S}$ 算法的伪代码如下:

Input: $G(V, E)$

$c(e(v_i, v_j))$ for any $v_i, v_j \in V_T$

maximum degree $deg_{max}(v_i)$

on session time $t_{on\text{-}session}(v_i)$

Root Node: $s \in V$

Output: $T(V, E')$

Procedure:

(1) $V_T = V_{T_candidateParent} \leftarrow \{s\}$

(2) $\overline{V}_T \leftarrow V - \{s\}$

(3) $DOET = \emptyset$; $DOES = \emptyset$

(4) sort according to $P_H(v_i, t)$ in ascending order in $V_{T_candidateParent}$

(5) while $\overline{V}_T \neq \emptyset$ do (6)–(19)

(6) Generate $DOE(v)$ for v

(7) for each edge $e(v, u) \in DOE(v)$ do (8)

(8) Calculate $TG(u)$

(9) $DOET \leftarrow$ sort according to $TG(u)$ in descending order in $DOE(v)$

(10) $DOES \leftarrow$ get top α of $DOET$

(11) sort according to delay(u) in descending order in $DOES$

(12) $u \leftarrow$ GetHead($DOES$)

(13) set v as u 's parent

(14) $deg(v) = deg(v) + 1$

(15) if $(deg_{max}(v) - deg(v) = 0)$ then (16)

(16) remove v from $V_{T_candidateParent}$

(17) if $(d_{max}(v) \geq 1)$ then (18)–(19)

(18) $V_T \leftarrow v$

(19) $\overline{V}_T \mapsto v$

(20) if $(\overline{V}_T = \emptyset)$ then (21)

(21) return $T(V, E')$

5.2 算法复杂性分析

分析算法可知, $TG\text{-S}$ 算法的时间复杂度为 $O(V^3 \cdot \log_2 V)$, $TG\text{-S}$ 算法的空间复杂度体现在存储完全图的边权值, 故其空间复杂度为 $O(V^2)$.

6 实验验证和性能评价

本文将 $TG\text{-S}$ 算法与 $NICE$ 协议在节点的平均接收延

时、最大延时以及组播树的累积中断次数方面进行比较,并分析实验结果。NICE^[1]是经典的分层分簇的应用层组播协议,它可以支持大量不同的数据转发树,有较强的可扩展性。

6.1 仿真实验环境

仿真工具为基于离散事件的、开源的多协议仿真器 OM-Net++。该仿真器可以运行于 Linux 和 Windows 等多个平台。

仿真的硬件环境为 Intel G630C 处理器,内存为 4GB,操作系统为 32 位 Windows 7。当节点数目为 100 时, TG-S 算法的仿真界面如图 2 所示。

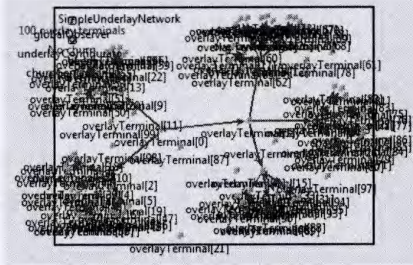


图 2 OMNet++ 仿真界面

6.2 实验参数设置

- 1) 节点的最大出度 $deg_{max}(v_i)$ 服从整数区间 $[2, 4]$ 之间的均匀分布;
- 2) 节点的在线时间 $t_{on-session}(v_i)$ 服从对数正态分布, 其参数设置为 $\mu=5.19, \sigma=1.44$;
- 3) 节点的相对离开概率更新周期 $\Delta T=120s$ 。

6.3 实验结果

将求解 SDMD 问题的 TG-S 算法与 NICE 算法在节点的平均接收延时、组播树延时、累积中断次数这 3 方面进行对比。

图 3、图 4 分别统计了两种算法在不同节点数目下的平均接收延时和树延时。

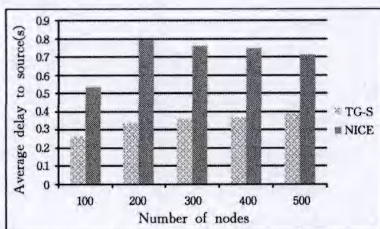


图 3 平均延时与节点个数的变化关系

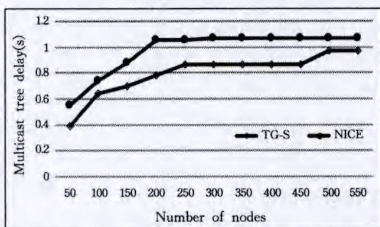


图 4 树延时与节点个数的变化关系

NICE 协议使用的数据传输树是基本路径树,该树不是一棵基于约束的优化树^[6]; TG-S 算法在 TG 值最大的 α 个节点中选取与源节点延时最小的节点作为孩子节点。因此在节点的平均接收延迟以及树延迟方面, TG-S 算法要优于 NICE

算法。仿真数据显示, TG-S 算法在平均接收延时方面比 NICE 低 51.5%, 在最大延时方面比 NICE 低 18.5%。

图 5 统计了两种算法在节点数目为 50 时的累积中断次数, 累积中断次数即当发生节点离开事件时, 因祖先节点离开而被迫中断与组播树连接的节点数目。

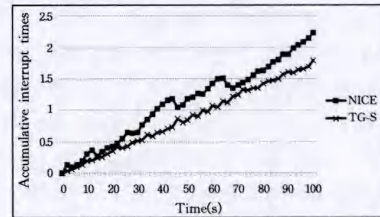


图 5 累积中断次数

在 NICE 协议中, 簇的领导节点是其拓扑结构的中心节点^[18], 这样的做法很可能会造成服务能力弱、在线时间短的节点处于转发树的高层, 使节点负载较高, 致使系统稳定性大为降低。仿真实验表明, TG-S 算法在累积中断次数方面较 NICE 算法降低 24.3%, 说明 TG-S 算法能够构建稳定性较高的应用层组播树。

TG-S 算法综合考虑了影响组播稳定性和延时的几个因素: 节点的最大度限制、节点的稳定概率和节点的在线时间, 因此利用 TG-S 算法生成的组播树具有较高的综合性能。

结束语 本文针对当前应用层组播对于系统的稳定性以及组播延迟方面的要求, 主要研究了如何构建高稳定性、低延迟组播树的问题。本文的主要贡献包括抽象出一种新的更加合理的问题模型——SDMD 模型, 并证明该问题属于 NP-hard; 针对该问题模型, 提出了基于节点稳定概率和时间增益因子的近似算法——TG-S 算法。

下一步, 将继续研究求解 SDMD 问题模型的算法, 以获得更优解。同时, 在真实的网络环境中, 必须考虑节点之间的连通度以及传输带宽问题, 这也是下一阶段的工作之一。

参考文献

- [1] Deering S E. Multicast routing in a datagram internetwork[R]. Stanford Univ CA Dept of Computer Science, 1991
- [2] Cao Ji-jun, Su Jin-shu. Delay-Bounded and High Stability Spanning Tree Algorithm for Application Layer Multicast[J]. Journal of Software, 2010, 21(12): 3151-3164 (in Chinese)
曹继军, 苏金树. 应用层组播的时延受限高稳定性生成树算法[J]. 软件学报, 2010, 21(12): 3151-3164
- [3] Deering S. Host extensions for IP multicasting. IETF RFC 1112, 1989[OL]. <http://www.ietf.org/rfc/rfc1112.txt>
- [4] Hosseini M, Ahmed D T, Shirmohammadi S, et al. A Survey of Application-Layer Multicast Protocols[J]. Communication Surveys & Tutorials IEEE, 2007, 9(3): 58-74
- [5] Yeo C K, Lee B S, Er M H. A survey of application level multicast techniques[J]. Computer Communications, 2004, 27(15): 1547-1568
- [6] Alpes P A I R, Project-Team P, Eric P, et al. Application-level Multicast Transmission Techniques over the Internet[J]. IEEE Transactions on Device & Materials Reliability, 2004, 12(2): 462-469
- [7] El-Sayed A, Roca V, Mathy L. A survey of proposals for an alternative group communication service[J]. Welding International, 2003, 17(1): 46-51

- [8] Francis P. Yoid; Extending the Multicast Internet Architecture [EB/OL]. <http://www.isi.edu/div7/yoid/docs/ycHtml/html-Root.html>
- [9] Chu Y, Rao S G, Zhang H. A Case for End System Multicast. 2000[J]. The Proceedings of Acm Sigmetrics, 2000, 28(1): 1-12
- [10] Zhang B, Jamin S, Zhang L. Host multicast: A framework for delivering multicast to end users[C]// Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies(INFOCOM 2002). IEEE, 2002: 1366-1375
- [11] Banerjee S, Bhattacharjee B, Kommareddy C. Scalable Application Layer Multicast[J]. ACM Sigcomm Computer Communication Review, 2002, 32(4): 205-217
- [12] Sripanidkulchai K, Ganjam A, Maggs B, et al. The Feasibility of Supporting Large-Scale Live Streaming Applications with Dynamic Application End-Points [J]. Proceedings of Acm Sigcomm, 2004, 34(4): 107-120
- [13] Su Jin-shu, Cao Ji-jun, Zhang Bo-feng. A Survey of the research on ALM stability enhancement[J]. Chinese Journal of Computer, 2009, 32(3): 576-590(in Chinese)
苏金树, 曹继军, 张博锋. 应用层组播稳定性提高技术综述[J]. 计算机学报, 2009, 32(3): 576-590
- [14] Roca V, El-Sayed A. A Host-Based Multicast (HBM) Solution for Group Communications [M] // Networking - ICN 2001. Springer, 2001: 610-619
- [15] Luo Jian-guang, Zhao Li, Yang Shi-qiang. An algorithm of constructing ALM tree based on user behavior analysis [J]. Journal of Computer Research and Development, 2006, 43(9): 1557-1563(in Chinese)
罗建光, 赵黎, 杨士强. 基于用户行为分析的应用层组播树生成算法[J]. 计算机研究与开发, 2006, 43(9): 1557-1563
- [16] Veloso E, Almeida V, Meira W, et al. A hierarchical characterization of a live streaming media workload[C]// Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurement 2002, Marseille, France, 2002: 117-130
- [17] Cao Jia, Lu Shi-wen. A Multimum Delay Spanning Tree Algorithm for the Application-Layer Multicast[J]. Journal of Software, 2005, 16(10): 1766-1773(in Chinese)
曹佳, 鲁士文. 应用层组播的最小延时生成树算法[J]. 软件学报, 2005, 16(10): 1766-1773
- [18] Cui Jian-qun, Xiong Nai-xue, Park J H, et al. A novel and efficient source-path discovery and maintenance method for application layer multicast[J]. Computers and Electrical Engineering, 2013, 39(1): 67-75
- [19] Zhang X C, Yang M H, Zhu X J, et al. A loss recovery approach for reliable application layer multicast[J]. Journal of Systems and Software, 2012, 85(5): 1198-1204

(上接第 67 页)

结束语 本文提出了基于中心块的自适应权重检索算法,通过对背景色主颜色的分析,有效地消除了背景色对中心块主颜色的影响并避免了图像中与目标关系不大的背景纹理。本文提出了运用差分演化算法在图像库中自适应优化权重值的方法,实验结果表明本文提出的基于中心块的自适应权重特征检索算法的检索精度有了较大的提高。

参 考 文 献

- [1] Nai C Y, Wei H C, Chung M K, et al. A fast MPEG-7 dominant color extraction with new similarity measure for image retrieval [J]. Journal of Visual Communication and Image Representation, 2008, 2(19): 92-105
- [2] Liang Mei-li, Niu Zhi-xian. Improved image retrieval with integrated colour and texture features[J]. Computer Application and Software, 2014, 31(6): 228-231(in Chinese)
梁美丽, 牛之贤. 改进的综合颜色纹理特征图像检索[J]. 计算机应用与软件, 2014, 31(6): 228-231
- [3] Huang Ren, Hu Min. Content-based image retrieval Using color position and texture fused features[J]. Computer Science, 2014, 41(6A): 118-121(in Chinese)
黄仁, 胡敏. 综合颜色空间特征和纹理特征的图像检索[J]. 计算机科学, 2014, 41(6A): 118-121
- [4] Ahmed T, Massudi M, Husniza H, et al. A weighted dominant color descriptor for content-based image retrieval[J]. Journal of Visual Communication and Image Representation, 2013, 24(3): 345-360
- [5] Shi J C. Fuzzy information retrieval based on a new similarity measure of generalized fuzzy numbers[J]. Intelligent Automation and Soft Computing, 2011, 17(4): 465-476
- [6] Nidhi S, Kanchan S, Ashok K S. A Novel Approach for Content Based Image Retrieval[J]. Procedia Technology, 2012(4): 245-250
- [7] Miao Liang, Sun Li-juan. Image Feature Matching Algorithm Based on Gauss Formula[J]. Journal of Henan University (Natural Science), 2013, 43(2): 196-199(in Chinese)
缪亮, 孙利娟. 基于高斯公式的图像特征匹配算法研究[J]. 河南大学学报(自然科学), 2013, 43(2): 196-199
- [8] Sun Jun-ding, Ma Yuan-yuan. Summary of Texture research [J]. Computer Systems & Applications, 2010, 19(6): 245-250 (in Chinese)
孙君顶, 马媛媛. 纹理特征研究综述[J]. 计算机系统应用, 2010, 19(6): 245-250
- [9] Liu Li, Kuang Gang-yao. Overview of Image Texture Feature Extraction Methods[J]. Journal of Image and Graphics, 2009, 14(4): 622-635(in Chinese)
刘丽, 匡纲要. 图像纹理特征提取方法综述[J]. 中国图象图形学报, 2009, 14(4): 622-635
- [10] Haralick R M, Shanmugam K. Texture features for image classification[J]. IEEE Transaction on System Man and Cybernetics, 1973, 3(6): 610-621
- [11] Sun Jun-Ding, Yuan Fang. Content-based Image Retrieval[J]. Computer System & Applications, 2011, 20(8): 240-244(in Chinese)
孙君顶, 原芳. 基于内容的图像检索技术[J]. 计算机系统应用, 2011, 20(8): 240-244
- [12] Zhu Hua, Ji Cui-cui. Fractal Theory and Its Applications[M]. Beijing: Science Press, 2011(in Chinese)
朱华, 姬翠翠. 分形理论及其应用[M]. 北京: 科学出版社, 2011