ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.17, Supplement, November 2006, pp.233−242

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

一种面向窄带环境的远程屏幕同步方法*

蒋文斌[1+], 金 海[1], 过敏意[2], 邵志远[1], 朱漳楠[1]

[1](华中科技大学 集群与网格计算湖北省重点实验室,湖北 武汉 430074)

[2](上海交通大学 计算机科学与工程系,上海 200030)

A Low Bandwidth-Oriented Remote Screen Synchronization Approach

JIANG Wen-Bin[1+], JIN Hai[1], GUO Min-Yi[2], SHAO Zhi-Yuan[1], ZHU Zhang-Nan[1]

[1](Cluster and Grid Computing Laboratory, Huazhong University of Science and Technology, Wuhan 430074, China)

[2](Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200030, China)

+ Corresponding author: Phn: +86-27-87557047, E-mail: wenbinjiang@hust.edu.cn, http://www.hust.edu.cn

Jiang WB, Jin H, Guo MY, Shao ZY, Zhu ZN. A low bandwidth-oriented remote screen synchronization approach. *Journal of Software*, 2006,17(Suppl.):233−242. http://www.jos.org.cn/1000-9825/17/s233.htm

Abstract: This paper presents an approach for remote screen synchronization by applying the association of spatio-temporal redundancy reductions (ASTR). The temporal redundancy between the current changed screen area and the previous frame is taken into account. By applying an improved SSC-APDS (Subsampling Search Candidates in Adjustable Partial Distortion Search) algorithm, the speed of the block motion estimation is increased obviously. Experimental results approve that this approach can save more bandwidth in transferring the screen data and reduce the overhead of the client. It has been used in a practical wireless application for conference projecting and cooperating named FreeSpeech with good performance.

Key words: temporal redundancy; server-based computing; ubiquitous computing; remote screen synchronization; wireless network

摘 要: 综合考虑计算机屏幕在变化过程中时间与空间的双重冗余,提出一种新的远程屏幕同步方法,以进一步减少传输的数据量.当前屏幕的变化区域和前一帧相应区域间的时间冗余得到了充分考虑.同时采用了一种改进的 SSC-APDS(subsampling search candidates in adjustable partial distortion search)算法,显著地提高了时间冗余处理过程中块运动估计的速度.实验结果表明该方法,能够进一步减少屏幕传输的带宽消耗,并降低客户端的开销.该方法已经在一个实际的无线会议投影与协作系统 FreeSpeech 中采用,取得了较好的效果.

关键词: 时间冗余;基于服务器的计算;普适计算;远程屏幕同步;无线网络

计算机与网络技术的飞速发展给人类生产和生活带来了极大的便利.人类的虚拟活动空间得到了空前的扩张.新技术手段的不断涌现使得人们在网络上可以实现对远端设备各种形式的访问和操作.其中,服务器计算(server-based computing,简称 SBC)模式更是使得人们能像访问本地机器一样很方便地访问和控制远程计算机.近年来,出现了许多基于服务器计算模式的远程访问工具,如 VNC[1],PCAnywhere[2],Netmeeting[3].由于这种模式

* Supported by the National Natural Science Foundation of China under Grant No.90412010 (国家自然科学基金)
Received 2006-03-30; Accepted 2006-10-08

的方便和快捷,这类工具已经在许多实际系统中得到了广泛的应用[4−6].在 SBC 中,以图形化屏幕的压缩和传输为基础的远程屏幕同步(remote screen synchronization,简称 RSS)机制是其中的关键技术之一.由于被访问的远程计算机(我们称之为服务器)屏幕通常会频繁地发生变化,产生大量图像数据,从而在传输过程中消耗大量网络带宽.

为了缓解带宽压力,研究者提出各种方法.早期,用来处理服务器屏幕数据的是一些相对简单的算法,如 Raw,RRE,Hextile,这些方法基本不涉及数据的压缩,从而使得带宽的消耗很大.随后,TridiaVNC[7]实现了两种被称为 Zlib 和 ZlibHex 的算法,它们都采用了标准的“zlib”库来实现对 Raw 和 Hextile 编码的数据进行进一步的数据压缩,从而使得需要传输的数据量有所减少.然而,带宽问题仍然十分突出.

为进一步解决问题,研究者提出了一些更复杂的方法来压缩和传输图形屏幕数据.Tight[7]是一种新的有效的编码方法,该方法通过采用数据分析器来判断输入数据的统计特性,从而决定采用哪种预处理过滤器来处理数据.在用 Zlib 压缩前,不同的数据过滤器被用来预处理不同类型的图像数据.这种方法试图找到一些特殊图像(如单色图像、大面积色块图像)以减少最终传输的压缩数据.这种方法对一些有许多空白区域的简单图像是有效的,但对复杂图像,效率不高.

OLI(optimal linear interpolation)[8]通过采用优化线性插值提出了一种新的基于像素的屏幕更新编码算法.服务器只发送一小部分像素样点来表示屏幕的更新,通过像素样点,客户端使用分段线性插值来恢复更新的全部区域.基于 OLI,产生了一种 2-D 无损线性插值(2DLI)算法.然而,这种方法仍然只是单独地将每个更新区域作为静态图像来处理.

文献[9]提出了一种 FCE(fast content expression)的屏幕编码方法,这种方法在给定方形区域内构造一个唯一像素值表,并将区域中的每个值转换为一个索引,从而实现屏幕数据的压缩.这种方法仍然没有考虑在不同时间更新不同区域的时间冗余.

此外,还有一些应用于特殊场合的特殊方法.如,对于嵌入式平台,文献[10]提出了一种应用于简单屏幕处理的方法——μVNC.不过,它只适合一些非常简单的屏幕处理,例如 VCR(video cassette recorder)中的控制板.

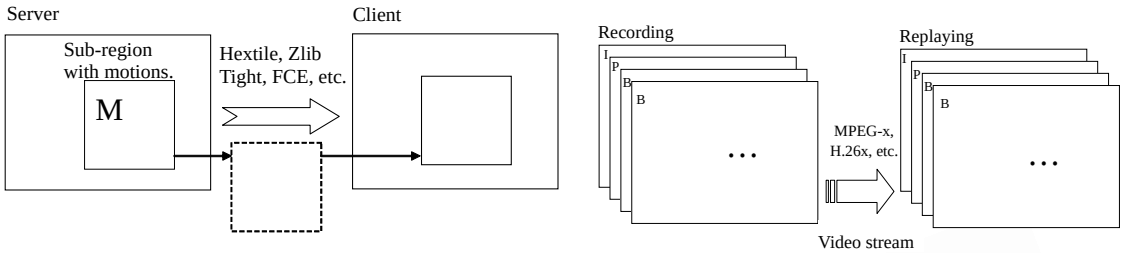
综上所述,我们发现,目前处理服务器屏幕通常采用的都是静态图像压缩方式.这些方法仅仅考虑如何减少单帧图像内的空间冗余,使得传输的图像数据的量依然很大,影响传输及显示效果.与此同时,随着无线计算技术的飞速发展,在普适环境下基于服务器计算的应用需求越来越大.人们希望通过各种无线设备以 SBC 的方式远程访问远端服务器的资源.然而,通常情况下,在无线环境中网络的带宽更窄、更不稳定,这给屏幕数据的传输带来了更大的瓶颈.

另一方面,由于要传输的图像是一系列具有用户界面风格的屏幕持续改变的区域,在两个连续屏幕图像之间通常存在大量的时间冗余.尤其在计算机图形界面中,平移是一种经常发生的动作,这也是产生时间冗余的一个重要因素.因此,可以应用这些潜在的时间冗余来进一步减少图像数据的传输.

为了进一步减少带宽消耗,本文提出了一种新的减少时空冗余的屏幕处理方法 ASTR(the association of spatio-temporal redundancy reductions).该方法考虑了当前屏幕变化区域与前一帧相应区域之间的时间冗余,并基于此,提出了一种新的远程屏幕同步机制.在时间冗余计算中,采用了一种改进的 SSC-APDS 算法[11]以提高块运动估计(BME)的速度.

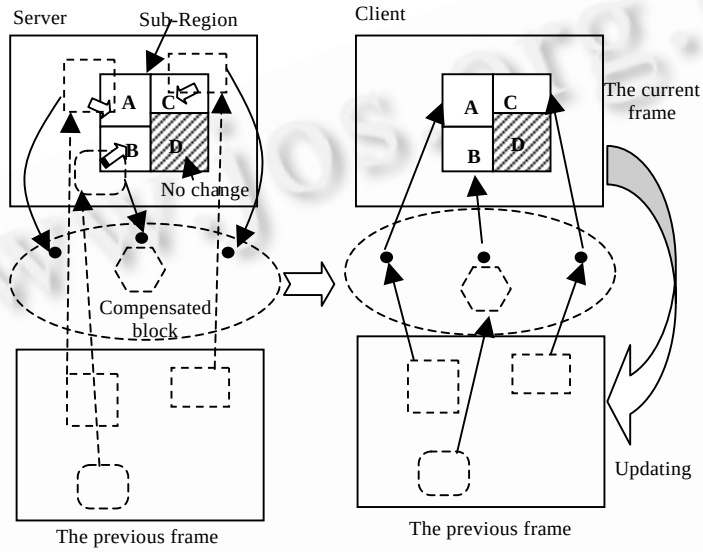
1 远程屏幕同步机制

在讨论新的桌面同步机制之前,来看一下传统远程屏幕同步中屏幕的处理方法:如图 1(a)所示,当服务器屏幕的某些区域(如图中所示 M)由于某些原因,如用户拖动滚动条、应用程序之间进行切换、连续播放动画,而发生变化时,系统捕获这些变化区域的图像,并采用 Hextile,Tight 或者 FCE 等方法进行压缩,然后按顺序一帧一帧发送到客户端.客户端接收到数据后进行相应的解码,最后在本地屏幕上显示.如果屏幕频繁发生变化,压缩后的静态图像数据量将会非常大,导致很大的传输开销.因此进一步减少数据的传输量是十分必要的,特别是在普适环境下.



(a) Static image process (b) Traditional video compression

(a) 静态图像处理方法 (b) 传统视频压缩方法



(c) ASTR approach

(c) ASTR 方法

Fig.1 Some approaches for screen process

图 1 几种屏幕处理机制

前面已经提到过在用户图形界面风格的屏幕变化过程中存在大量的时间冗余,因此,可以利用这一冗余特点来进一步降低屏幕变化过程中产生的压缩数据.事实上,减少时间冗余已经应用在许多传统的视频压缩方法中,例如 MPEG-x,H.26x.同时,这些传统方法也能够被用来处理计算机变化的屏幕.一个最典型的应用是屏幕录像(如图 1(b)所示),即将一段时间屏幕的变化记录为一个流媒体文件,便于以后重放,可以用在计算机教学、内容展示等诸多场合.然而,这些方法不仅极大地增加了服务器的开销,而且产生了更多传输数据,因为这些方法每一帧都是对整个屏幕进行压缩,且压缩帧率固定.因此,该类方法不适合具有实时特性的远程屏幕同步.

这里,为适应远程屏幕同步的特点,并进一步减少传输数据量,提出了一种新的远程屏幕同步机制——ASTR(如图 1(c)所示).首先,在某一时刻获取服务器屏幕上一个区域(称为子域,sub-region).这些区域经常不是整个屏幕,而是屏幕中经常发生变化的某些局部区域.这是 RSS 的第一步,在传统的同步方法中已经提到,这一步可以看成是空间冗余的处理.然而,在 ASTR 中,不是将这个子域作为静态图像来压缩,而是与前一帧的相应子域进行比较,从而将子域划分为一些不同类型的子区(sub-area),以达到进一步减少时间冗余的目的:

(a) 无变化子区(如图 1(c)中的 *D*).由于我们用一个矩形来涵盖包含运动的区域,与前一帧相应的子区比较可能存在一些没有任何变化的子矩形区域,由于这些子矩形区域和前一帧对应区域的内容完全一样,因此完全没有必要压缩和发送.但在传统的方法中,这些区域也一起被处理了;

(b) 平移运动子区(如图 1(c)中的 *A* 和 *C*).这意味着这个子区能够由前一帧中相应子区仅通过平移运动得

到.即可以用前一帧子区的矩形位置及到当前位置的运动向量来表征.在传统的方法中,这类子区也作为静态图片进行压缩并发送.而在 ASTR 中,仅需要发送前子区的位置和运动向量的元数据,而不是整个图像区域,这将极大地减少传输的数据量.在许多情况下,对于图形用户界面,用户倾向于进行滚动条拖动,窗口移动等等操作,从而产生大量平移运动,所以通过元数据来表征这种区域的运动很有意义;

(c) 复杂运动子区(如图 1(c)中的 *B*).这时我们不能在前一帧中找到一个与当前子区完全相同的对应区域.我们需要通过补偿块和运动向量等参数一起来描述这种类型的子区.当然,我们应当在前一帧中当前子区对应位置的附近寻找一个最相似的子区来简化补偿块.补偿块越简单,压缩的数据就越少.然后采用传统静态图像压缩方法来压缩补偿块.

通过上述 3 种类型子区的划分,当前帧与前一帧之间的时间冗余得到了充分考虑.在获得这些子区后,我们应该将相关的数据发送到客户端.对于(a),显然无须传送任何数据;对于(b),将表征子区原位置及运动向量的元数据发送到客户端(图 1(c)中用黑点表示);对于(c),需要发送更多的数据,除了元数据外,需要将补偿块(图 1(c)中用六边形表示)压缩并发送.通过上述分析,可明显看到,客户端和服务器均需要分别保存屏幕的前一帧信息.当客户端接收到待更新子区的相关信息时,它会根据位置信息在前一帧搜索得到相关子区,并综合所有信息,重构相应区域.对于(a),无须做任何事情.对于(b),只需根据运动向量将相应子区复制到当前位置.对于(c),问题稍微复杂一些,需要解压补偿块,并结合前一帧相应子区来产生当前子区.

从上面提出的方法,可以看出不仅传输的数据量会明显减少,而且客户端的开销也会在一定程度上降低,这主要是由于需要解压缩的数据量减少了.然而,由于需要花费时间来搜索不同类型的子区,会导致服务器开销有所增加.幸运地是,在许多远程控制和协作的应用中,服务器通常比客户端有更强的计算能力,通过增加部分服务器开销来减少带宽消耗和客户端开销是值得的.此外,基于一种改进的 SSC-APDS 算法,我们有效地提高了子区分类中 BME 的速度,使得服务器的工作效率更高.

2 改进的 SSC-APDS 算法

为了识别不同类型的子区,当前变化的子域被划分为一系列标准的宏块(通常是 16×16).在这里,BME 用来完成宏块的运动估计和绝对误差和 SAE 的计算.图 2 给出了 BME 基本处理过程:设 *A* 为子域中的一个宏块,以 *A* 为中心,在前一帧对应得一定搜索范围内寻找一个最合适的宏块 A_p,使其与 *A* 最为相近,也就是与 *A* 的 SAE 最小.由 A_p 到 *A* 的位移即为 *A* 的运动向量.在求出每一个宏块的运动向量和位移以后,便可以据此来完成子域中不同类型子区的分类.

由于 BME 是一个费时的过程,研究者已经提出了诸多快速算法[11−13].SSC-APDS 是最近提出的高效方法之一,它对宏块的候选位移搜索点和宏块内的像素点同时进行抽样.该算法精度高、速度快.然而,由于该算法搜索总是从搜索区域的中心点开始(如图 3(a)所示),不够灵活,有时会带来较大计算量.而在本文的应用环境中,存在大量因滚动条的运动、窗口的移动而导致的平移运动,因而许多相邻宏块具有相同或相似的运动向量的概率很大.如果对于每个宏块,我们都从搜索区域中心开始估计它的运动,而一些宏块的最佳运动点远离中心,搜索算法将花费很长时间才能到达最佳点,因为在到达该点之前需要沿着螺旋扫描路径需要进行多次误差和的比较.

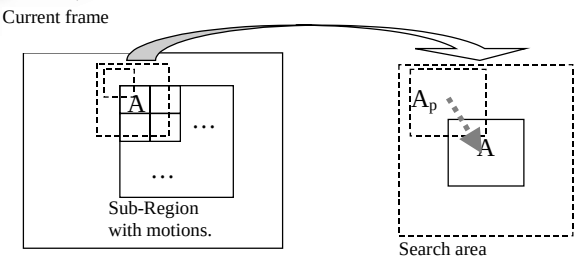
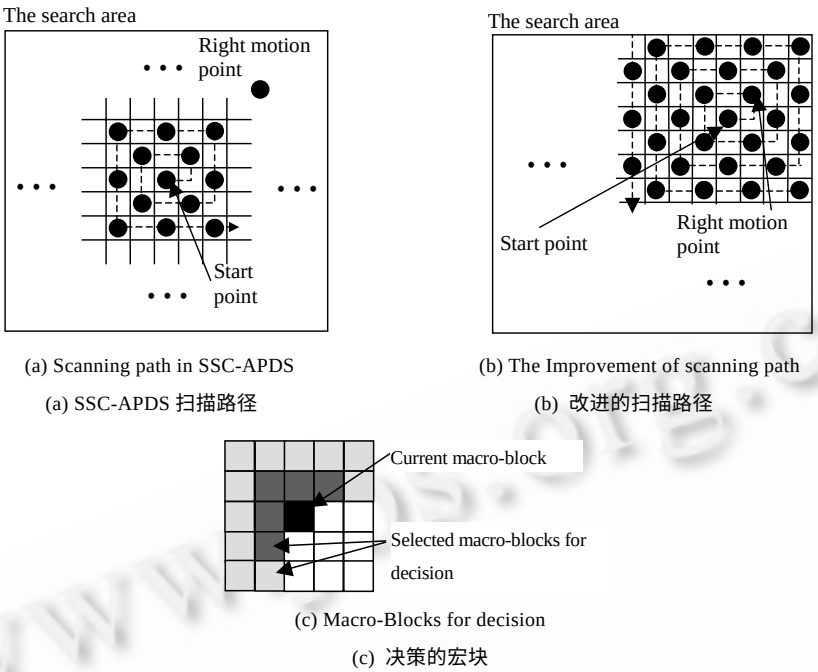


Fig.2 The process of BME

图 2 BME 处理过程



(a) Scanning path in SSC-APDS

(a) SSC-APDS 扫描路径

(b) The Improvement of scanning path

(b) 改进的扫描路径

(c) Macro-Blocks for decision

(c) 决策的宏块

Fig.3 Improvement of the SSC-APDS

图 3 改进的 SSC-APDS 算法

为了提高 BME 的速度并使宏块更加适合合并,我们对 SSC-APDS 算法进行了改进,将搜索起点从中心移动到一个最合适的位置,如图 3(c)所示.当前宏块周边的一些宏块的运动被用来作当前宏块最适合起点的参考(如图 3(c)所示).最适合的起点计算如下:

$$\vec{V}=\frac{1}{1+\lambda}\left(\frac{1}{n}\sum_{i=1}^{n}\vec{V}_{1i}+\lambda\frac{1}{m}\sum_{j=1}^{m}\vec{V}_{2j}\right) \tag{1}$$

其中,$\vec{V}$ 是最合适的搜索起点(如图 3(b)所示),$\vec{V}_{1i}$ 是最靠近当前宏块的周边各宏块的运动向量,$\vec{V}_{1j}$ 是次靠近的周边宏块的运动向量.n,m 是分别是 $\vec{V}_{1i}$ 和 $\vec{V}_{1j}$ 的数目.λ是影响因子,在这里λ的值设为 0.5.同时那些运动向量尚未被计算的宏块不予考虑.

通过使用改进的 SSC-APDS 算法,可以获得所有宏块的运动向量和 SAE.

3 子区识别和客户端重构

3.1 子区识别

通过考察宏块的运动向量和 SAE,我们合并具有相同特征的相邻宏块作为子区.如图 4(a)所示,子区是一些互相连通的区域.白色代表的是没有发生任何改变的宏块.具有相同运动向量,且只发生平移的宏块用点标记(即 SAE 为 0,并且运动方向相同).用栅格标记的块的 SAE 不为 0.因为这样得到的每个子区通常是一个很难描述的不规则区域(图 4(b)所示),所以有必要将其划分为一些规则的矩形,便于处理.图 4(c)说明了这种划分方法.基本规则如下:

(1) 选择子区的左上宏块,将它作为初始矩形;

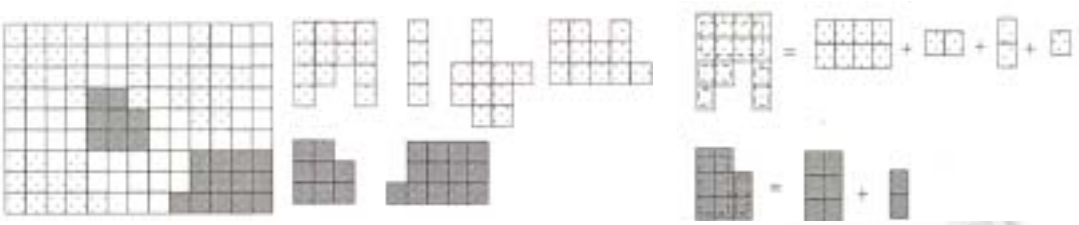
(2) 做水平延伸,搜索所有最右边宏块的相邻宏块,如果这些相邻块均存在,将它们添加到当前矩形中;

(3) 做垂直延伸,搜索所有最下方宏块的相邻宏块,如果这些相邻块均存在,将它们添加到当前矩形中;

(4) 交替重复上述步骤(2)和(3)直到不能继续为止;

(5) 从子区中移去当前矩形;

(6) 重新开始搜索,重复步骤(1)~(5)直到子区为空.



(a) The caught area (a) 当前子域 (b) Sub-Areas found (b) 提取的子区 (c) Division of sub-areas into regular rectangles (c) 将子区划分为规则矩形

Fig.4 The process of finding sub-areas

图 4 子区的获取过程

在进行上述处理后,当前获得的区域被划分为一系列规则矩形.这些矩形可分为只有平移(SAE 为 0),或有复杂运动(SAE 不为 0)两种类型.分别地,对于这两种类型,下面的数据会被发送至客户端.

(1) 只有平移的矩形:(运动,位置).

(2) 复杂运动矩形:(各宏块的运动向量,补偿矩形).

通过每个宏块的运动向量和服务器上保存的前一帧信息,补偿矩形很容易得到.

3.2 客户端重构

对于客户端而言,根据服务器发送的数据容易重构待更新的区域(如图 5 所示).对于只有平移运动的矩形,仅需要做的是根据运动向量和位置信息将相应的矩形从前一帧复制到目的位置.对于复杂运动的矩形,处理过程稍微要复杂一些.首先根据每个宏块的运动在前一帧中找到相应预测宏块,并合并形成预测矩形,然后将接收到的补偿矩形和预测矩形进行重叠从而形成最终的更新矩形.

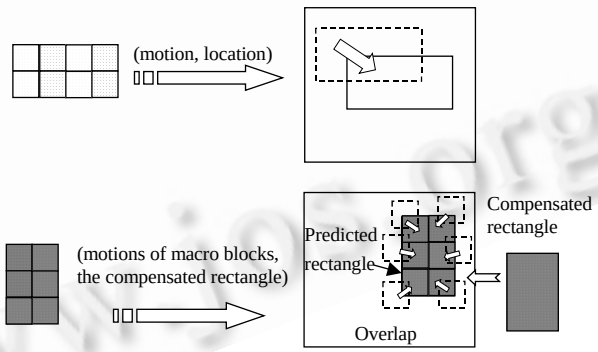


Fig.5 Reconstruction of client

图 5 客户端重构

补偿矩形的压缩方式决定了客户端更新的屏幕是无损还是有损的.如果采用无损压缩方式,例如应用 Zlib,则更新后的屏幕是无损的.否则,便是有损的.在以前的一些方法中,为了满足某些窄带宽环境的需要,有损方式经常被作为首选.不过,在本文的方法中,由于发送的数据量已经有了明显地下降,因此无损方式可以优先考虑,这将能有效地保证客户端屏幕的清晰度.

当服务器当前屏幕变化区域和前一帧的相应区域非常相似时,本文提出的方法将非常高效.然而,如果前者很大且与后者的区别很大,将会消耗大量时间来进行子区的分类,因为这时 BME 会带来大量的计算量.在这里,采用一个折中的方法来缓解这个问题.当我们开始处理变化子域时,首先采用改进的 SSC-APDS 算法分析少量的散布宏块.从它们的运动和绝对误差和,我们能够判断当前获取的子域和前一帧子域的相似程度.如果差别很

大,则采用传统的方法(如 Tight)而不是 ASTR 来压缩这个区域.当然,这些宏块应该尽量分散以能够反映整体的情况.

在实际情况中,有时由于某些随机因素会导致数据包的丢失.在以前的方法中,这种意外被忽略,因为下一幅图像会迅速覆盖丢失数据包的相应区域.然而,本文提出的方法很多时候仅仅发送的是补偿图像,并且必须利用客户端的前一帧来更新变化的区域.如果没有采用其他措施,丢失的数据包将可能较长时间影响客户端屏幕.幸运的是,在许多实际应用中,数据传输是不连续的.当用户思考、讲话、休息时有大量空闲时间,这时服务器可以把整个屏幕作为静态图像发送到客户端来更新屏幕从而解决上述问题.下一步,本研究将考虑设计一个更有效的丢包重传协议来使得本文的方法在一些不稳定的环境中应用得更有效.

4 实 验

本文提出的上述方法在一个实际的无线会议协作与投影系统 FreeSpeech[14]中得到了实现和验证.FreeSpeech 是我们实验室自主开发的一个基于远程屏幕共享,工作在 Ad Hoc 网络上的无线会议系统,如图 6 所示,系统由一个 Manager、一个以上的 Provider 和若干 Viewer 组成.演讲者作为 Provider 将自己的屏幕传给 Manager,Manager 通过投影议将其显示出来.同时多个 Provider 可以协同显示和自由切换.进一步地,为兼顾因距离或障碍物影响无法看清投影的其他参会者,Manager 也可将显示内容组播给多个参会者(viewer),进行本地显示.本文提出的方法,主要用于 Provider 到 Manager 端屏幕传输.所有代码均通过 Visual C++ 6.0 实现,且具体测试环境如下:服务器:IBM 笔记本电脑,操作系统:Windows XP,CPU:Pentium M 1.7GHz,内存:512MB;客户端:Asus 笔记本电脑,操作系统:Windows XP,CPU:Pentium M 1.4GHz,内存:256MB;网络:802.11b 无线局域网,带宽:11 Mb/s.

为了能较全面地测试算法的效果,这里共测试了 10 个应用场景,涉及多窗口间的切换、文档的编写操作、PPT 的播放、网页浏览等.在这里着重讨论其中两个典型的代表(如图 7 所示).一个是进行文档操作的场景,另一个是包含一些复杂运动的教学演示场景.为了简便起见,分别命名为 S1,S2.在 S1 中,我们在阅读器中打开一个 PDF 文件并且以 150ms 每行的速度自动滚动.因为在我们操作计算机时,滚动是一个非常普遍的动作,因此比较的结果是很具有代表性的.S2 是一个比较复杂的场景,包括平移,缩放,旋转等各种运动元素.两个场景中 S1 选择的是打开参考文献[9]的 PDF 文件,S2 选择的是执行 Windows XP 系统中的 tour.exe 教学程序.

这里,我们将本文提出的 ASTR 方法与传统的 Hextile,Tight 和 FCE 进行比较.

图 8 给出的是 4 种方法的网络带宽消耗.(1)和(2)分别是 S1 和 S2 的结果.

明显地,ASTR 比其他方法占用更少的带宽.表 1 列出了 4 种方法各自的平均带宽.在 S1 中,Hextile 的平均带宽大约为 865KB/s,比其他方法占用更多带宽,这是因为它只是对屏幕变化子域进行了简单处理,没有采用压缩算法.Tight 和 FCE 采用了较复杂的压缩算法来减少变化子域内的空间冗余,从而使得占用的带宽明显减少.进一步,ASTR 考虑了时间和空间的双重冗余,所以带宽的消耗进一步下降.从 S2 的结果中,可得到相似的结论.

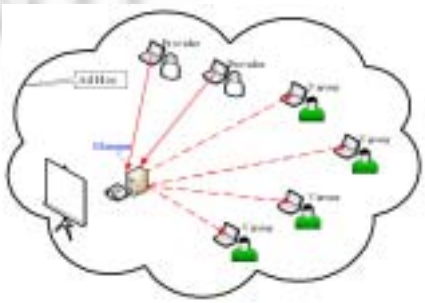


Fig.6 FreeSpeech framework

图 6 FreeSpeech 框架

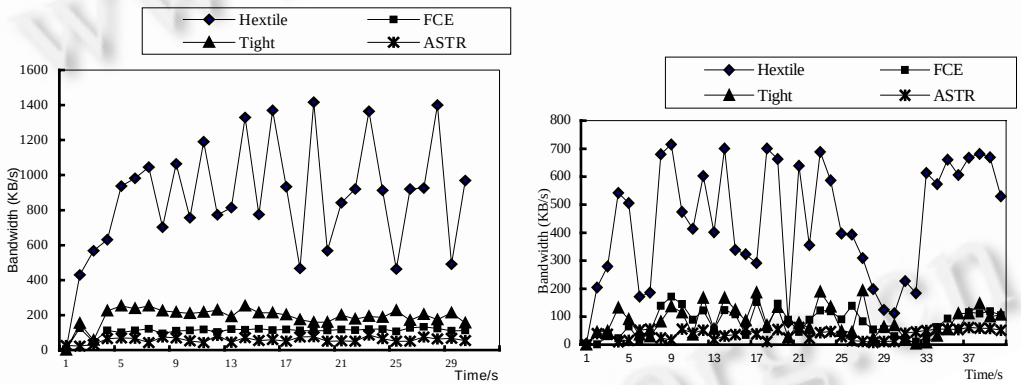
(a) Reading PDF file (b) A flash tour playing

(a) PDF 文件阅读 (b) flash 教学程序

Fig.7 Test scenarios

图 7 实验场景

图 9 显示了对于 4 种不同的方法,服务器 CPU 的消耗情况.对于 S2,ASTR 比其他方法花费更多的 CPU 时间,这种情况是在预料之中的,因为在对于子区进行分类时,BME 通常要消耗一定的时间.然而,对于 S1,出乎意料的是,我们发现 ASTR 竟比 Tight 和 FCE 花费更少的 CPU 时间.导致这一结果的主要原因是屏幕滚动时产生了大量的平移运动,这使得 BME 时间明显降低,同时,也导致极少补偿块需要处理,从而在静态图像压缩上花费的时间也明显减少.



(1) S1 Bandwidth consumption (2) S2 Bandwidth consumption

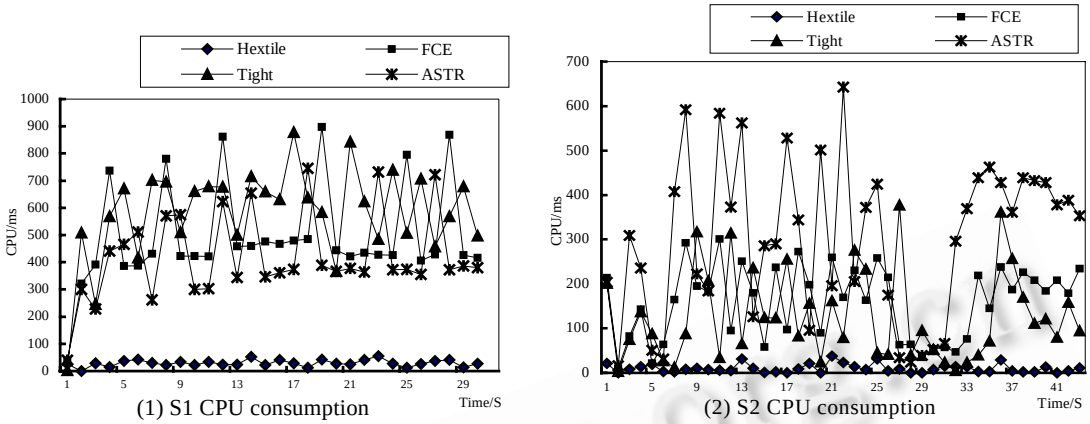
Fig.8 Bandwidth consumption

图 8 带宽的消耗

Table 1 Consumptions of bandwidth (BW) and CPU

表 **1** 带宽与 CPU 消耗

		Hextile	Tight	FCE	ASTR
S1	带宽(KB/s)	865	191	109	58
	服务器(CPU/ms)	29	580	494	421
	客户端(CPU/ms)	11	120	113	54
S2	带宽(KB/s)	302	67	61	23
	服务器(CPU/ms)	8	99	117	202
	客户端(CPU/ms)	3	21	23	14



(1) S1 CPU consumption (2) S2 CPU consumption

Fig.9 CPU consumption of the server

图 9 CPU 的消耗

表 1 给出通过 4 种方法服务器和客户端的 CPU 平均消耗的统计情况.从表中可以发现,在客户端,ASTR 通常比 Tight 和 FCE 占用更少的 CPU 时间,这是因为与其他方法相比,ASTR 需要解压缩的图像数据更少.

最后,我们测试了一种极端的情况.Provider 全屏浏览一组照片,并发送到 manager 端.这时,我们发现 manager 端出现较大时延,带宽占用大、且显示效果不好.这主要是因为,在这种情况下,两帧图像间的时间冗余很小,在进行 BME 计算时,复杂变化子区占绝大多数.因而,在最终处理时,仍然将当前帧作为静态图像压缩并传输,则出现了传统方法存在的数据量大,显示效果差等问题.这也是本文后续研究工作需着重考虑的问题之一.

以上实验表明,在 FreeSpeech 中,绝大多数情况下,本文方法能明显减少了网络带宽的消耗,尤其是在多个 Provider 同时协同显示时.本文的方法同样适合其他应用场合的远程屏幕同步.具有很好的应用前景.表 2 列出了本文提出方法的主要特点.

Table 2 Comparison between ASTR and traditional approaches

表 2 ASTR 与传统方法比较

	传统方法	ASTR
冗余考虑	仅空间冗余	时间冗余+空间冗余
是否保留前一帧信息	否	是
网络带宽占用	较大	较小
服务器负载	较小	较大
客户端负载	较大	较小
适合网络环境	有线环境	普适环境

5 结 论

为了进一步减少在 SBC 应用中网络带宽的消耗,本文提出了一种新的远程屏幕同步机制.该机制综合考虑了屏幕变化过程中存在的时间和空间双重冗余.实验结果表明这种方法使得远程屏幕同步比许多传统的方法(如 Hextile,Tight 和 FCE)更有效,主要表现在能节省更多的网络带宽,并且有效地降低客户端的开销.服务器的开销由于耗时的 BME 处理将在一定程度上有所加大.不过,如果应用中存在大量平移运动,服务器的开销会明显减少.总的来说,这种方法非常适合于网络带宽和客户端计算能力均十分有限的普适计算环境.

References:

[1] Richardson T, Stafford-Fraser Q, Wood KR. Andy hopper A: Virtual network computing. Internet Computing, 1998,2(1):33−38.

[2] Guo Z, Moulder JC. An Internet based telemedicine system. In: Proc. of the Information Technology Applications in Biomedicine. 2000. 99−103.

[3] Swamy N, Kuljaca O, Lewis FL. Internet-Based educational control systems lab using NetMeeting. IEEE Trans. on Education, 2002,45(2):145−151.

[4] Leiner R. Tele-Experiments via Internet a new approach for distance education. In: Proc. of the Electrotechnical Conf. 2002. 2002. 538−541.

[5] Talwar V, Basu S, Kumar R. An environment for enabling interactive grids. In: Prof. of the 12th IEEE Int'l Symp. on High Performance Distributed Computing. 2003. 184−193.

[6] Jung DK, Kim KN, Kim GR, Shim DH, Kim MH, Choi BC, Suh DJ. Biosignal monitoring system for mobile telemedicine. In: Proc. of the HEALTHCOM 2005. 2005. 31−36.

[7] Kaplinsky KV. VNC tight encoder-data compression for VNC. In: Proc. of the 7th Int'l Scientific and Practical Conf. of Students, Post-Graduates and Young Scientists. 2001. 155−157.

[8] Li F, Nieh J. Optimal linear interpolation for server-based computing. In: Proc. of the IEEE Int'l Conf. on Communications. New York, 2002. 2542−2546.

[9] Li Q, Li F. FCE: A fast content expression for server-based computing. In: Proc. of the 2004 IEEE Int'l Conf. on Communications. 2004. 1426−1430.

[10] Haraikawa T, Sakamoto T, Hase T, Mizuno T, Togashi A. μVNC: A proposal for Internet connectivity and interconnectivity of home appliances based on remote display framework. IEEE Trans. on Consumer Electronics, 2001,47(3):512−519.

[11] Jiang W, Zhou M. A fast BMA based on combining search candidate subsampling and APDS. In: Proc. of the 2004 IEEE Int'l Conf. on Multimedia and Expo (ICME 2004). 2004. 1115−1118.

[12] Cheung CK, Po LM. Normalized partial distortion search algorithm for block motion estimation. IEEE Trans. on Circuits and System for Video Technology, 2000,10:417−422.

[13] Cheung CH, Po LM. Adjustable partial distortion search algorithm for fast block motion estimation. IEEE Trans. on Circuits and Systems for Video Technology, 2003,13:100−110.

[14] Jiang W, Jin H, Shao Z, Ye Q. FreeSpeech: A novel wireless approach for conference projecting and cooperating. In: Proc. of the UIC-06. LNCS 4159, Heidelberg: Springer-Verlag, 2006. 688−697.

蒋文斌(1975－),男,湖南衡阳人,博士,讲师,主要研究领域为普适计算,P2P 计算,多媒体,网格计算.

金海(1966－),男,博士,教授,博士生导师,主要研究领域为计算机体系结构,集群计算与网格计算,对等计算,普适计算,Web 和网络安全,多媒体技术.

过敏意(1962－),男,教授,博士生导师,主要研究领域为高性能计算,并行编译与算法,普适计算,嵌入式软件的优化编译.

邵志远(1975－),男,讲师,主要研究领域为高性能计算,嵌入式系统.

朱漳楠(1978－),男,硕士生,主要研究领域为普适计算,P2P 计算.