

基于改进 UDT 协议的实时视频传输*

Real-time Video Transmission Based on Improved UDT Protocol

章 超 周家建 (浙江理工大学, 浙江 杭州 310018)

王 勤 李晨曦 (国网浙江温岭市供电公司, 浙江 温岭 317500)

王月敏 (浙江奇汇电子提花机有限公司, 浙江 杭州 310051)

摘 要

提出了一种基于改进 UDT 协议实时视频传输的协议算法, 研究了使用该种改进协议进行实时视频传输的效果及性能, 并与使用 UDP 协议和 UDT 协议进行实时视频传输进行了对比, 发现使用 UDP 协议传输视频质量较差, 会出现花屏的现象, 使用 UDT 协议传输延时较大, 对实时视频传输来说是不可接受的。因此需要对 UDT 协议做出改进, 使得其在传输实时视频的时候可以有一个较好的表现。

关键词: UDP, UDT, 实时视频传输, 传输延时

Abstract

This paper presents a method of real-time video transmission based on improved UDT protocol and studied its effect and performance. And compared with transmitting real-time video by UDP protocol, the quality of video by using UDP protocol to transmit real-time video was a little bad, it may appear blurred screen. And compared with UDT protocol itself, the delay of transmission was too large. In order to adjust to transmitting real-time video, the UDT protocol should be improved.

Keywords: UDP, UDT, real-time video transmission, transmission delay

使用 UDP^[1] 协议进行视频数据的传输, 具有传输开销小、速度快、效率高的优点; 但是 UDP 协议是面向非连接的, 使用其进行视频数据传输并不可靠, 会造成视频帧的丢失^[2], 在接收播放端可能就会出现花屏乃至黑屏的现象。UDT^[3-4] 是一个定义良好的、改进的、基于 UDP 的应用层协议, 是一种面向连接的协议, 该协议包含了可靠控制和拥塞控制策略, 是一种可靠的传输扩展协议。使用 UDT 协议进行视频数据的实时传输, 可以有效提升视频传输的质量。但是与 UDP 协议相比, 使用 UDT 协议进行视频传输延时较大, 这对实时视频传输来说是不可以接受的。

UDT 协议对网络丢包的重传是造成视频传输延时的主要原因, 网络拥塞越严重, 延时也会越大。同时在实时视频的传输中, I 帧能否准确到达接收播放端, 对传输视频的质量也起着至关重要的影响。综合上诉影响实时视频传输质量的因素, 本文提出了一种基于改进的 UDT 协议进行实时视频传输的协议算法。该改进协议针对视频编解码及实时视频传输的特点, 修改 UDT 协议底层的发送端和接收端, 不发送已经超时的数据包, 使得其在传输实时视频的时候, 视频质量能得到一定保证的情况下, 同时能够有效地控制传输延时。

1 基于改进的 UDT 协议实时视频传输

由于实时视频传输数据量较大, 所以在传输之前, 需要对采集到的视频数据进行压缩, x264 编码库可以很好的完成这个任务^[5]。一般经过 x264 编码会产生 I 帧、P 帧和 B 帧, 而这三种帧类型在解码端的重要性是不一样的, 在一个 GOP (画面组, 指的是一组连续的画面) 序列中, I 帧是不依赖别的帧数据就可以直接在解码端解码成功的, 如果没有 I 帧, 属于这个 GOP 的 P 帧和 B 帧是无法解码的。P 帧是前向预测帧, B 帧是双向内插帧^[6]。P 帧或者 B 帧的丢失, 会造成该 GOP 中以其为参考帧的数据帧解码错误, 并造成扩散效果。

UDT 协议是一种可靠的传输协议, 对于在传输过程中丢失的数据包, UDT 协议会对其进行重传, 其重传算法保证了数据的可靠传输。UDT 重传算法如图 1 所示:

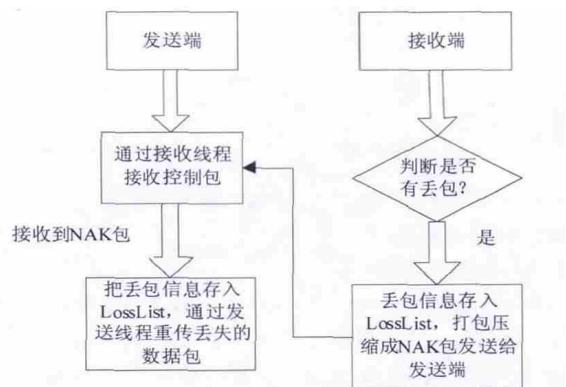


图 1 UDT 协议数据重传流程

UDT 发送端的数据发送线程在发送数据的时候首先会去查找是否有需要重发的数据包, 如果有需要重发的数据包, 发送端就会从数据发送缓冲区找到相应的数据包进行重发, 否则就直接从数据发送缓冲区取出下一个需要发送的数据进行发送。但是从前文所述可知在网络状况不是很好的情况下, 对所有数据都进行重发, 会造成较大的延时, 这对实时视频传输来说是一件不可以接受的事情。因此需要对 UDT 协议数据的发送和重发策略进行改进。通过引入数据包生存期阈值, 可以有效控制收发两端的延时。改进的 UDT 协议在发送或者重传数据的时候, 需要判断数据包的生存期是否超过了生存期阈值, 如果超出了范围, 就不再发送或者重传这个数据包。

然而在同一 GOP 序列中, 如果其中一帧数据出现包的丢

* 杭州市企业高新技术研发中心项目 (20111131P16) 资助

失,就会造成后续的视频帧无法正确解码,因此即使后续需要发送的数据包生存期在阈值范围内,再去发送这些数据是没有意义的。所以在 UDT 协议的应用层需要给 UDT 底层传递一个 I、P、B 帧的标志,这样改进的 UDT 协议在因为数据包超时而主动仍包后,可以通过这个数据包帧类型的标志不去发送这个 GOP 后续的数据包,从而可以减少一些不必要的发送开销。

UDT 协议数据发送端在经过上述改进后,其数据接收端也要进行相应的修改。发送端是根据数据包的生存期主动清除 SndLossList (发送端记录丢包信息的列表),而接收端的 RcvLossList (接收端记录丢包信息的列表)这个时候是不知道发送端不会再发送超时的数据包,因此发送端需要通知接收端同步清除 RcvLossList 中这些数据包的标志,这个可以通过在发送的数据包中加一个是否清除 RcvLossList 的标志 flag 完成。在数据发送端判断出需要发送的数据包生存期超过阈值不发送时,就把下一个可以发送的数据包 flag 置为需要清除 RcvLossList 的值,在接收端接收到这个数据包之后,就知道此数据包之前的数据不会再传,接收端就可以从 RcvLossList 删除掉这个包之前的丢包信息。

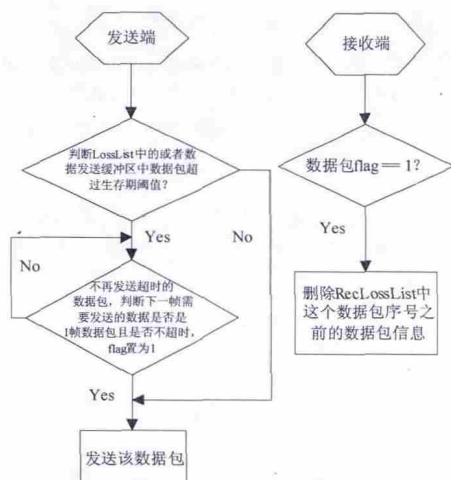


图2 改进的 UDT 协议

2 实验过程与分析

2.1 实验环境及实验过程

视频发送及接收客户端 OS:Android 2.3 操作系统。实验环境由两台 Android 智能手机组成,一台作为视频采集发送端,另一台作为视频接收播放端。实验系统的构成方式是首先接收手机 b 作为客户端连接到视频发送手机 a,然后智能手机 a 使用自带的摄像头采集视频数据并用 x264 进行编码,转发给接收端手机 b,智能手机 b 接收实时视频数据,并进行解码显示。

按照上述流程,实验分成三组进行。第一组:采用 UDP 协议进行实时视频传输;第二组:采用 UDT 协议进行实时视频传输;第三组:采用基于改进的 UDT 协议进行实时视频传输。在三组实验中,分别对客户端的视频质量和传输延时进行测试与分析。

2.2 实验结果及实验分析

在上述不同的协议下进行实时视频传输,对比三组实验的传输延时,分别测试在一次传输过程中,不同时间点的传输延时,实验在局域网下进行,三组实验通过在发送端人为丢包来模

拟 3G 网络易丢包的情况,实验三中数据包生存期阈值设为 2s,测试结果如表 1 所示。

表1 延时时间对比

| | 实验一(s) | 实验二(s) | 实验三(s) |
|------|--------|--------|--------|
| 测试点1 | 0.5 | 2.3 | 0.8 |
| 测试点2 | 0.9 | 8.4 | 2.8 |
| 测试点3 | 1.2 | 11.5 | 1.5 |
| 测试点4 | 0.7 | 5.8 | 3.5 |
| 测试点5 | 1.5 | 13.2 | 1.2 |
| 测试点6 | 1.3 | 22.0 | 1.8 |

对比实验二与实验三的实验结果,可以看出实验二采用 UDT 协议进行实时视频数据的传输,延时较大,延时也会随着视频数据量的变化和丢包率的变化而剧烈波动,由于延时比较大,不符合实时视频传输的要求。而实验三采用改进协议进行实时视频传输,很好地控制了视频传输的延时。

对比实验三与实验一的视频接收播放端的视频质量,实验三的视频传输避免了因为数据包丢失而造成接收解码播放端出现花屏的现象,相较于实验一视频质量更加优秀;实验三与实验二相比较,实验三视频传输虽然在某些时刻会出现画面跳跃的情况,但不会出现像实验二那种长时间停留在一个画面的现象。三组实验视频质量对比如图 3、4、5 所示:



图3 UDP 传输



图4 UDT 传输



图5 改进的 UDT 传输

3 结束语

实验结果表明,相较于使用 UDT 协议进行实时视频传输,采用改进的协议算法进行实时视频传输,很好地控制了传输延时,并且在视频质量上比采用 UDP 协议传输更加优秀。

参考文献

- [1]张威.Linux 网络编程教程[M].北京:北京希望电子出版社,2002:133-147
- [2]郝力.基于 Linux 的实时视频传输技术的研究[D].郑州:中国人民解放军信息工程大学,2006:3-4
- [3]王艳芳.基于 UDP 的多媒体数据实时可靠传输研究[D].湘潭:湘潭大学,2009
- [4]Yunhong Gu, Robert L.Grossman. Supporting Configurable Congestion Control in Data Transport Services[J]. In the proceedings of the 2005 ACM/IEEE SCIO5 Conference, 2005
- [5]孙泉.支持 H264 的实时流媒体服务器的设计与实现[D].北京:北京邮电大学,2010
- [6]牛建民.H.264 视频压缩算法应用研究[D].上海:同济大学,2007

[收稿日期:2013.8.5]

欢迎订阅 投稿

欢迎到当地邮局订阅 2014 年我刊《工业控制计算机》,订阅代号:28-60,月刊,定价:12.00 元/期。

投稿可直接发往邮箱:ipcm_nj@163.com,投稿详细要求请参见我刊网站(www.ipcm.com.cn)“稿件征集”频道。