

文章编号: 1007-757X(2014)2-0053-03

一种基于 WebRTC 的网络遥操作系统

倪 智

摘 要: 遥操作系统能够使人类在危险或者复杂的工业场景中进行实时操作。在当前的遥操作系统中, 操作端的视频质量对于操作者能否正常完成工作起着决定性的作用, 而实时的视频质量又会受到带宽受限的网络环境的影响, 且不能实现便捷的移动设备的访问。为此, 设计了一种基于 WebRTC (Web with Real Time Communications) 的网络遥操作系统, 利用 WebRTC 技术来实现远端到操作端的实时音视频传输, 并且利用 Websocket 来传输操作端的控制指令, 并且实现了远端移动平台的运动规划。最后的实验表明该系统能够有效降低传输速率, 提高视频质量。

关键词: 遥操作系统; WebRTC; Websocket; 视频质量

中图分类号: TP249

文献标志码: B

A Network Teleoperation System Based on WebRTC

Ni Zhi

(Department of Automation and key laboratory of System Control and Information Processing, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract: Teleoperation systems enable people to do real-time work in hazardous or dangerous industrial situations. In the current teleoperation systems, the video quality on the control side plays a decisive role whether the controller could finish the work smoothly. The real-time video quality is affected by the network environment such as limited bandwidth, and portable devices could not have access to the teleoperation system. To overcome this drawback, a network teleoperation system based on WebRTC is proposed in this paper. The system uses WebRTC to transmit real-time video and audio from the remote side to the control side, and uses Websocket to transmit control command from the control side, and the motion planning is realized in the remote motion platform. The final experiment shows that the system can reduce transmission rate and improve video quality.

Key words: Teleoperation System; WebRTC; Websocket; Video Quality

0 引言

随着工业生产的发展和科技的进步, 应用于复杂工业现场的遥操作系统逐渐成为当今学术界研究的一个热点^[1]。通常情况下由于远端处于移动状态, 相机捕获的视频流运动剧烈, 传输速率大; 远端和操作端的网络带宽也时时在发生变化, 而传输速率过大就会带来不同程度的视频丢包, 影响操作端的视频质量。另外, 由于操作者经常处于工业现场, 这也使得研究基于移动设备的遥操作系统成为了一种趋势^[2]。如何能够更加便捷的实现远端监控, 而且能够降低传输速率, 保证视频质量就成为了一项重要而且艰巨的任务, 也成为网络遥操作系统需要解决的一个突出问题。WebRTC^[3]技术作为一种浏览器内部的实时音视频交互技术, 采用比较高效的音视频编解码算法和网络自适应算法, 支持跨平台的调用, 成为了新兴高效的流媒体编解码和传输方法; 而 Websocket^[4]则是下一代的实时通信协议, 效率高且占用资源较小。WebRTC 和 Websocket 的出现, 使得解决上述问题成为了可能。综上所述, 本文设计了一种基于 WebRTC 的网络遥操作系统, 传输远端的实时音视频信息; 利用 Websocket 传输操作端的控制指令, 并实现了远端移动平台的运动规划。最后的传输速率测试和视频 MOS (Mean Opinion Score) 评分测试证明了该系统能够有效降低传输速率, 提高操作端的视频质量。

1 系统结构

如图 1 所示:

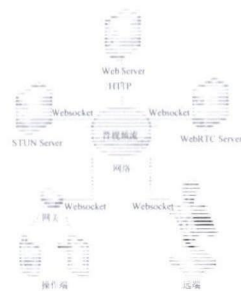


图 1 基于 WebRTC 的遥操作系统结构图

为基于 WebRTC 的网络遥操作系统的结构框图。整个系统分为六个部分: 操作端、远端移动平台、网络、STUN Server、Web Server 和 WebRTC Server。Web Server 主要提供网络中的 Internet 服务, 使得操作端和远端可以使用 HTTP 协议从 Web Server 上下载网页。WebRTC Server 主要完成音视频流的初始化工作, 操作端和远端首先向 WebRTC Server 发送基于 Websocket 协议的消息, 其后 WebRTC Server 便向两端转发对端的 ID 信息, 这样便能完成音视频流的 P2P 传输。STUN Server 主要目的是使用户数据报协议能够穿越 NAT 防火墙, 使得不同网络结构的终端可以进行通话。通

作者简介: 倪智 (1987-), 男, 上海交通大学自动化系, 硕士研究生, 研究方向: 网络遥在机器人, 上海, 200240

过 STUN Server 可以了解公网地址和网络类型。对于控制指令而言,利用 Websocket 协议传输控制指令,在远端首先进行 Websocket 的握手,然后实时解析控制指令,并根据远端环境信息进行运动规划。由于 android 和 IOS 系统的 Chrome 浏览器具备 WebRTC 功能,所以该遥操作系统能够兼容手机等移动设备,基于 WebRTC 的遥操作系统结构图如图 1 所示。

2 软件设计

2.1 音视频交互软件设计

对于音视频交互软件来讲,主要功能就是向操作端实时传输远端的环境信息。该软件主要采用 WebRTC 进行设计。在支持 WebRTC 功能的 Chrome 浏览器内部,已经集成了众多与音视频编解码、音视频传输有关的 API,使用 C++ 实现,Web 开发者仅需通过上层的 Javascript 调用,便可以实现浏览器内部的实时音视频交互。如图 2 所示:

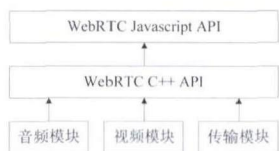


图 2 基于 WebRTC 的遥在系统结构图

基于此,本系统采用如下软件流程实现音视频交互。该软件主要分为客户端软件和 WebRTC Server 软件两部分。对于客户端软件来讲,首先通过 `getUserMedia` 方法捕捉本地的音视频流,其次,通过 `webkitPeerConnection` 方法与 WebRTC Server 建立连接并向 STUN Server 请求 NAT 服务。连接建立成功后,向 WebRTC Server 传输本地音视频流初始化信息,并等待接收 WebRTC Server 传回的 ID 信息。而 WebRTC Server 软件主要监听服务器某个特定端口,一旦检测到两端的连接请求便进行 ID 信息的转发。音视频交互软件流程图和操作端为手机时的操控界面分别如图 3 和图 4 所示:



图 3 音视频交互软件流程图



图 4 操作端界面

2.2 控制软件设计

对于控制软件来讲,主要功能就是实现控制指令的传输以及机器人的运动规划。控制指令传输之前,首先要进行的是 Websocket 的握手。远端首先检测操作端的连接请求,并且解析其中的密钥,经过安全的哈希算法和 base64 编码后,得到解密密钥并向操作端发送,操作端判断解密密钥是否正确,如果正确则传输控制指令,远端接收控制指令并根据环境信息进行运动规划,若不正确则通信关闭。握手阶段的软件流程如图 5 所示:



图 5 握手阶段的软件流程图

对于机器人的运动规划算法,采用的是势场法的避障算法^[5],其原理图如图 6 所示:

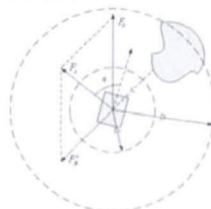


图 6 势场法避障原理图

认为环境和操作端的操作对远端的移动平台以力的形式产生作用,表现为目标点对于远端移动平台的引力和障碍物对于移动平台的斥力。在原理图中, F_G 为引力, F_R^i 为第 i 个距离传感器检测到障碍物产生的斥力, F_c 为合力, θ 为合力 F_c 与移动平台正方向的夹角, r 为第 i 个距离传感器检测到障碍物到移动平台的距离, $[d, D]$ 为障碍物有效影响距离区间。其算法流程图如图 7 所示:

认为环境和操作端的操作对远端的移动平台以力的形式产生作用,表现为目标点对于远端移动平台的引力和障碍物对于移动平台的斥力。在原理图中, F_G 为引力, F_R^i 为第 i 个距离传感器检测到障碍物产生的斥力, F_c 为合力, θ 为合力 F_c 与移动平台正方向的夹角, r 为第 i 个距离传感器检测到障碍物到移动平台的距离, $[d, D]$ 为障碍物有效影响距离区间。其算法流程图如图 7 所示:

认为环境和操作端的操作对远端的移动平台以力的形式产生作用,表现为目标点对于远端移动平台的引力和障碍物对于移动平台的斥力。在原理图中, F_G 为引力, F_R^i 为第 i 个距离传感器检测到障碍物产生的斥力, F_c 为合力, θ 为合力 F_c 与移动平台正方向的夹角, r 为第 i 个距离传感器检测到障碍物到移动平台的距离, $[d, D]$ 为障碍物有效影响距离区间。其算法流程图如图 7 所示:

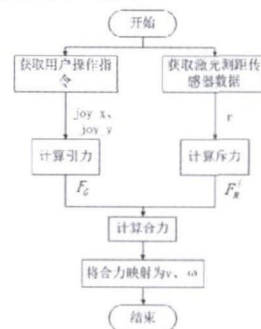


图 7 运动规划控制算法

3 实验分析

为了验证基于 WebRTC 的网络遥操作系统能够有效降低传输速率,并能够提高操作端的视频质量,现分别设计传输速率测试实验和视频质量测试实验来验证上述结论。远端的人形移动平台如图 8 所示:



图 8 远端人形移动平台

其中,头部装有 Logitech 720p 摄像机,用户可以控制摄像机达到期望的位置;距离传感器感知角度为机器人正前方 270° ,感知范围为 3m。其底层移动平台最大速度为 1.1m/s。对于视频质量的评价标准,目前广泛采用的是国际电信联盟 (ITU) 建议的“平均评估分值”法(MOS)^[6],它精确的描述了用户的主观感受。评价标准如表 1 所示:

表 1 视频 MOS 评分标准

评分	视频质量等级	视频失真等级
5	优秀	不能察觉
4	良好	可以察觉,但不讨厌
3	一般	明显察觉,但可以容忍
2	差	明显察觉,让人讨厌
1	很差	非常明显,非常讨厌

本次实验选择了网络中广泛使用的 FLASH 的流媒体传输方法^[7]作为对比,下面,将分别对两个实验进行介绍。

3.1 传输速率测试实验

在传输速率测试实验中,远端移动平台处于实验室内环境中,操作者操控远端移动平台,并接收远端传输的音视频流,记录操作端接收的流媒体传输速率,一天内三个不同的时间段的测试结果如图 9 所示:

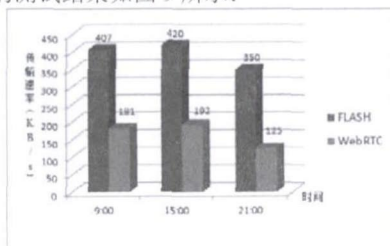


图 9 传输速率测试结果

其中,机器人端的视频流参数为 640×480 ,帧率为 30fps。

由图 9 可以看到,在相同的远端场景和相同的视频流条件下,WebRTC 的传输速率要远低于 FLASH 的传输速率,表明基于 WebRTC 的方法能够比基于 FLASH 的方法效率高且带宽占用少,在相同的受限带宽下能够保证视频流的流畅传输。

3.2 视频质量测试实验

在视频质量测试实验中,远端移动平台处于实验室内环境中,选择 5 名操作者(4 男 1 女)操控远端移动平台,操作者根据表 1 所示的视频质量评分标准进行打分,统计操作

者的打分情况并取均值,采用不同的视频分辨率传输的结果如图 10 所示:

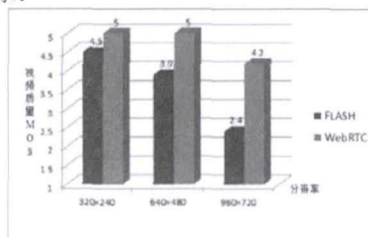


图 10 视频质量测试结果

从图 10 中可以看到,在相同的远端场景下,随着视频分辨率的升高,基于 FLASH 的视频传输方法用户的 MOS 评分逐渐降低,特别是在 960×720 的分辨率条件下,FLASH 的传输方法已经出现了较大程度的卡顿和延迟,表明了 FLASH 的传输方法对于编解码标清以上分辨率的视频流存在卡顿和延迟;相比较而言,WebRTC 的视频传输方法在任何的视频流分辨率下,MOS 评分均在 4 分以上,且视频流畅,图像清晰,可以保证操作端的视频质量。

4 结论

本文首先讨论了遥操作系统的主要问题,并据此提出了一种基于 WebRTC 的网络遥操作系统,笔者详细阐述了该系统的结构和软件设计,并通过对比实验验证了该系统能够有效降低传输速率,提高视频质量。此外,该系统还能实现操作端的跨平台调用,实现了更加便捷的远端监控,在实际应用中收到了良好的效果。

参考文献

- [1] Ross B, LaRose D, Wilde, etc. High performance teleoperation for industrial work robots[C]. Applied Robotics for the Power Industry (CARPI), Oct, 2010: 1-5.
- [2] Park J H, Song T H, Jeon J H P A J W. Usability Analysis of a PDA-based User Interface for Mobile Robot Teleoperation[C]. The IEEE International Conference on Industrial Informatics, July, 2008: 1487-1491.
- [3] Bertin, Cubaud E, Tuffin S, et al. WebRTC, the day after: What's next for conversational services?[C]. Intelligence in Next Generation Networks (ICIN), Oct, 2013: 46-52.
- [4] 徐漫江, 陈天皓. 实时 Web 工业控制系统原型的设计与实现[J]. 化工自动化及仪表, 2013, 40(8): 971-974.
- [5] 吴镜开, 黄远灿, 王世兴. 基于势场法的移动机器人避障路径规划[J]. 微计算机信息, 2007, 23(5).
- [6] 仇栋才, 赵国锋, 薛飞. IPTV 测试[J]. 电信工程技术与标准化, 2007, 20(2).
- [7] Mohsin Khan, Jason Gu. Web based teleoperation architecture and h.264 video encoder[C]. Electrical & Computer Engineering (CCECE), April 2012: 1-4.

(收稿日期: 2014.01.09)