NTU Journal of Pure Sceinces (2024) 3 (2) : 22-27
DOI: https://doi.org/10.56286/s0h8bg27

IRAQI
Academic Scientific Journals

ISSN: 2789-1089 EISSN: 2789-1097
NTU Journal of Pure Sciences
Available online at: https://journals.ntu.edu.iq/index.php/NTU-JPS/index

Real Time Video Streaming Over Homogeneous Systems

Hesham Hashim Mohammed [1]
1. Northern Technical University, Mosul, Iraq

Article Informations

Received: 06-03- 2024,
Accepted: 02-04-2024,
Published online: 14-07-2024

Corresponding author:
Name: Hesham Hashim Mohammed
Affiliation: Northern Technical University, Mosul, Iraq
Email: hesham@ntu.edu.iq

Key Words:
Video Streaming,
Real Time Protocol,
User Datagram Protocol (UDP),
Frame Drop,
Video streaming jitter

ABSTRACT

Streaming videos from one device to another become more important due to the spread of mobile devices and social media applications. The process of sending video from one device to another have a set of challenges. The video must be delivered in time, otherwise the service quality will decrease. Packet drop and lip synchronization is another challenge that streaming video will encounter. Real Time Protocol (RTP) is a simple, lightweight protocol that used to perform video streaming with low burden. RTP uses User Datagram Protocol (UDP) in video transferring. This paper use streaming video through RTP over homogeneous systems and measure the Packet Loss Rate, Jitter, Out-of-Order Packets, Datagram Size, Packet Delivery Ratio, latency, and throughput. Four experiments were performed with (15, 30, 45, 60) seconds to check the effect on the previous measures. The results show that video jitter increases as streaming time increases, it also shows that latency and throughput doesn't depend on the streaming period.

Introduction

Streaming video file from one device to another device or devices is reffered as video streaming and it has greatly grouth in the last two dedcates. Videos like news, coking, fishing, sport, music, movies and other activities are now avalible on the Internet. Video streaming adds another exctra burder on the Internet due to the fact that fideo files usually have more size than audios or images [1].

Internet Engineering Task Force or (IETF) group has developed a protocol that resopnsible about end to end real time user communication for delevaring audio and video over IP services, this protocol is named RTP. RTP stands for real time protocol and is reponsible about end to end transmission for real time applications, but it doesnt garantee the quality of service [2], [3].

Straming videos has many benfits, on of them is that they allow users to watch videos without the need for downloading them, another benfit it that it can provide the same video with different resolutions up to 4K. As well as the affortability. Nevertheless, video streaming still has a set of challenges such as the limiteations imposed by client device (reciver) as well as slow connection. In the former case the client device may be old or run a set of processes which caues errors in stream receving, this may cause low Quality of Service [4], [5].

This paper studies the effect of transfering video usind RTP in the context of the following measures:-

1. **Packet Loss Rate**: - packet loss is defined as the loss of one or more packets from the original message, this occure due to many causes such as bad chanel connection or network status like congestion [6], [7].
2. **Jitter**: - jitter can be defiend as the loss of transmitted data while hopping via network devices. Lossing packets may cause to uncorectlly reassemble the packets [8], [9].
3. **Latency**: - is one of the important measures for videos streaming. Latency defiend as the difference in time between the instant of when the live streaming occure and the instant when it starts at the reciver screen [10], [11], [12].
4. **Out of order packets**: - as the name implies out of order measure means that the packets reachs the distination in sequence differs from the sequence the send from the source. This may happened if the network has some troubles and it defentlly affects the the video quality [13], [14].
5. **Packet Deleivary ratio**: - The ratio between the number of recived packets at destnation to the number of actually sent packets from the source node is called PDR [15], [16].
6. **Throughput**: - the rate of transmitting video from source to distnation successfuly, it measured by (bps), (Kbps), (Mbps) which are bit per second, Kilo bit per second, Mega bit per second respectivly [1], [17].

The rest of this paper is orginized as follow: Section two views a set of related works about real time video streaming. Section three describes the methodology of the proposed work, the results are disscussed in section four, section five views challenges and feuter works.

Related Works

This section will view a set of related works, their chllenges and benfits.

1. GROOT: - In 2020, Kyungjin Lee and et. Al present GROOT which is a video streaming system that can stream real time and on demand videos. The former is live video streaming while the latter is streaming previuosly recorded videos. In either the cases GROOT allow to adapt user video viwing continuously. GROOT works on 3D video which is a great challenge because the size of file will increase, the authors also work on algorithm to filter and clarify the 3D points outside of a user's view. The results shows that GROOT achieves stable and faster frame rates compared to any previous method to stream and visualize volumetric videos on mobile devices [18].
2. In 2021, Yu and Chung propose analyzing the content of live video streaming to extract information about the content of video streaming. The authors explain that this information will be very nessesary to manege the process of streaming live videos, this is duo to the fact that controling on live videos is very nessesary. The challenge hear is that information extracted from live videos are usually not ordered and mixed most of the times [19].
3. Yang, et. al propose in 2022 an object detection in real time live video streaming that will help in autonomous driving. The authors develope an algorithm for streaming perception. Furthermore, the system supplied with novel Dual- Flow Perception module (DFP) which capture moving trend that helps in prediction.the system applied on Argoverse-HD dataset and improves the AP by 4.9% compared to the strong baseline [20].
4. In 2023, Foo, et. al propose Systemstatus- aware Adaptive Network (SAN) that can provide high quality and low latency system status predection. The preduction of the system states considering real time instantaneous variables is very helpful to manege efficiency and robustness to fluctuations of the system status. The authors also propose a Meta Self-supervised Adaptation (MSA) method to adaptivly configure

new hardware at test time. This is due to the fact that in some times any change in hardware will affect on the system behaviour and the proposed MSA will make the deployment easier even with unseen devices [21].

Methodology

This section describes the algorithm used to transmet the video over the network using real time protocol. This section splited to two subsections, the first one describes the algorithm used on sender device, i.e source, while the other one describes the algorithm on the reciver device or the destination. Its worth mensioning that his application done using python pycharm community edition 3023 3.3. the sender and reciver devices have following properties.

Table 1. Sender and Reciver Devices Properties.

Property	Sender Device	Reciver Device
Device type	VAIO sony VPCEL 17FX	HP Paviilion DV6
Processor	AMD E-350 Processor 1.60 GHz	Intel Core i5 M 460 2.53GHz
RAM Size	4.00 GB	4.00 GB
OS	Windows 10	Windows 10
System Architecture	64-bit operating system, x64-based processor	64-bit operating system, x64-based processor

A. Sender Device

At the sender side, the sender first initilize the buffer size to be 65536 bytes, then initilize the socket that will send the data from. The socket here represent the IP address along with the port address. The former represent the IP of the sender device while the latter represent the prosess to be used at both sender and reciver sides which in our case equals to 9999.

At this point the connection from sender side is completed and the device ready to stream the video to the other side, which is the reciver side. After that the sender must initilize the variables and buffers that will be used for results measurment. Some variables are measuerd like number of lossed packets are scaler, while other neds array to measuer the value for each packet such as frame jutter.

Its worth mensioning that at this time there are a set of operations performed at the reviver side, these steps are very nessessary to complete the streaming operation. These steps will be mensioned at the next subsection.

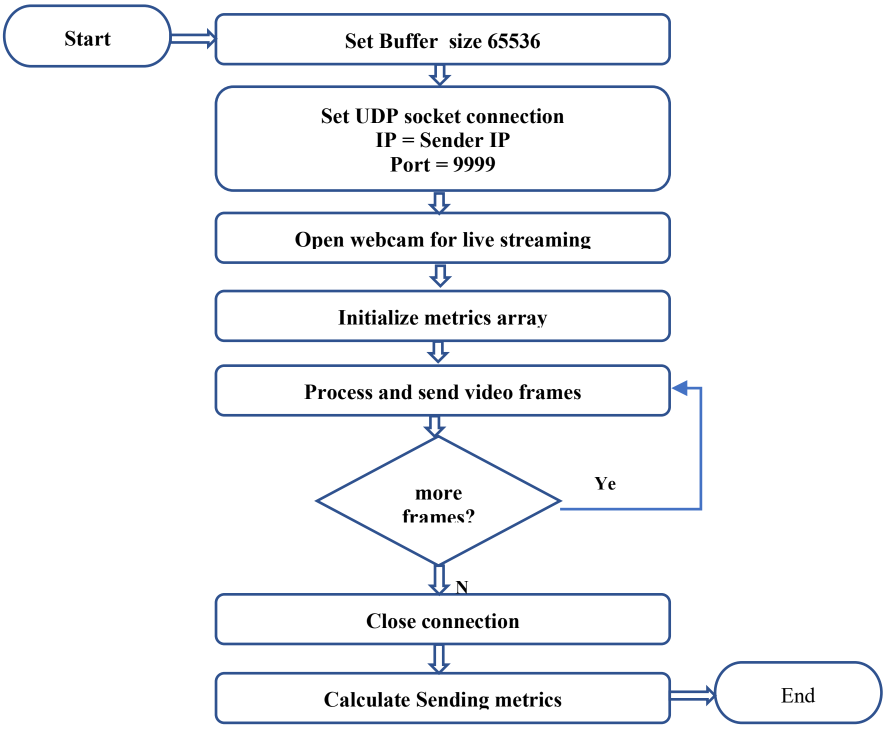

Start
Set Buffer size 65536
Set UDP socket connection
IP = Sender IP
Port = 9999
Open webcam for live streaming
Initialize metrics array
Process and send video frames
more frames?
Ye
N
Close connection
Calculate Sending metrics
End

Figure 1. Sender side flowchart.

At this time, the sender starts to prosecc each video frame to make it ready to encapsulated inside UDP packet and then sent to the reciver side using real time protocol RTP. This prosess will continue until all video packets sent.when there are no more frames, the sender will close the connection and calclate the sending metrics, See figure 1.

B. Reciver Device

At the reciver side, the reciver first initilize the buffer size to be 65536 bytes, then initilize the socket that it will listen to it. The socket here represent the IP address of the sender and the port which is 9999.

At this time the connection with sender side is completed and the device ready to recive video stream from the other side, which is the sender side. At this time the reciver listening to the port continuasly and process the recived packets to extract frames and display them in order. Meanwhile the reciver still recive and prosess packets continuaslly until the connection ended. When connection ended the reciver calclate the result metrics

Its worth mensioning that during reciving process, the reciver may receive frames out of their order, this may happend due to the network status. The reciver machine will re arrange them to be inorder.also the reciver may recive coropted packets and this will effect the QoS and other metrics, See figure 2.

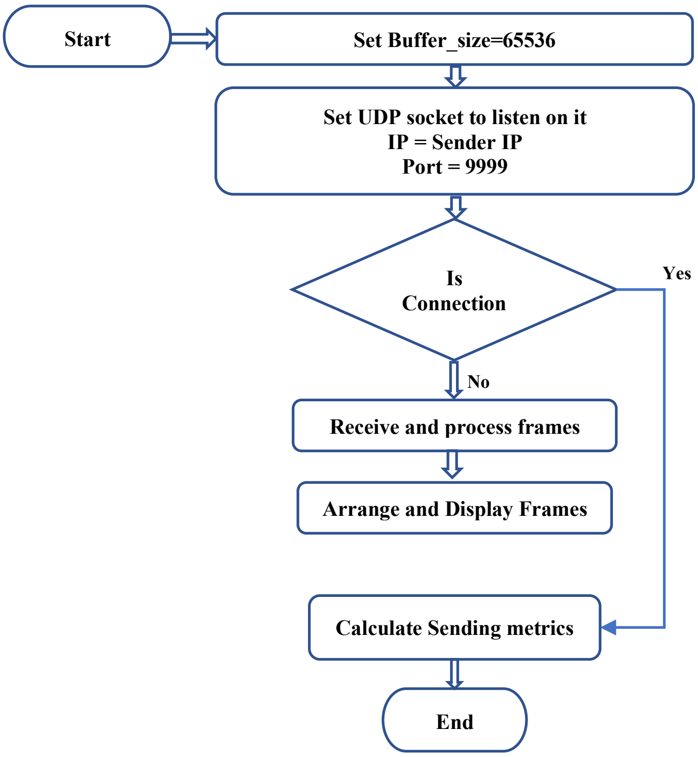

Start
Set Buffer_size=65536
Set UDP socket to listen on it
IP = Sender IP
Port = 9999
Is
Connection
Yes
No
Receive and process frames
Arrange and Display Frames
Calculate Sending metrics
End

Figure 2. Reciver side flowchart.

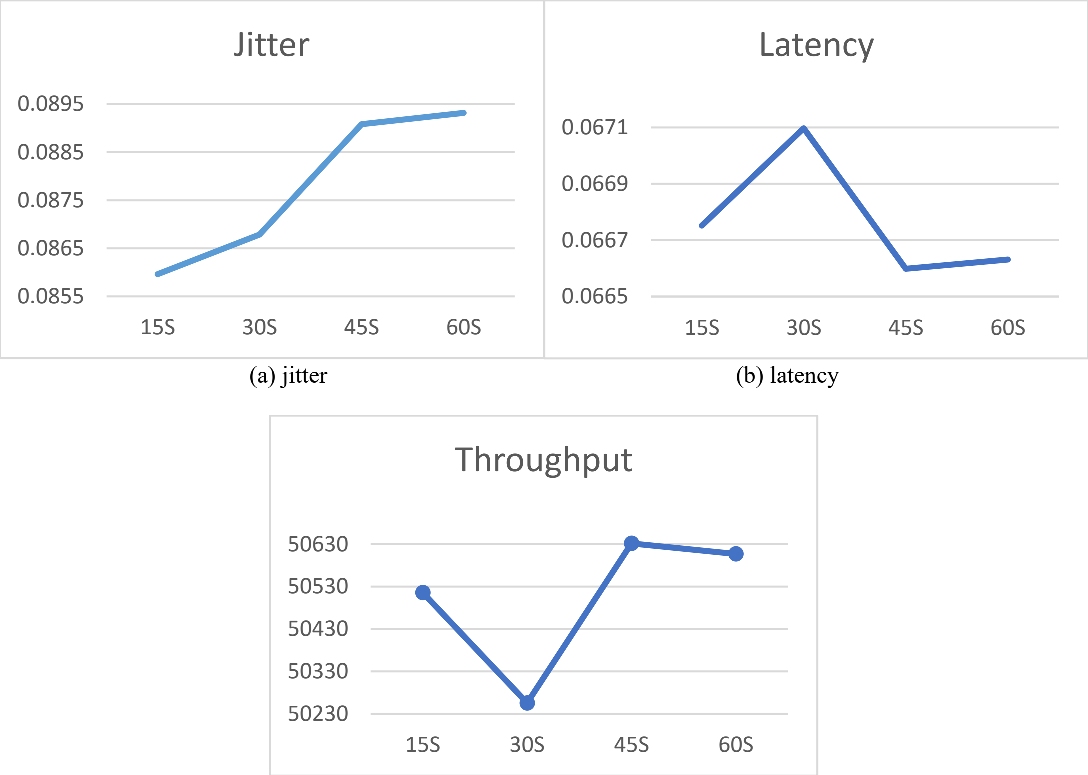
Results

This paper measure the results in four experments with time equals to (15, 30, 45, 60) seconds respectivly. Table 2 shows the jitter, latency, and throughput in the four experments.

Table 2. Experments result.

Time	Jitter	Latency	Throughput
15S	0.085962	0.06675194587	50515.38
30S	0.086789	0.06709719375	50255.45
45S	0.089085	0.0665985330	50631.74
60S	0.089317	0.06663094813	50607.11

Table 2 shows that the jitter increas as the streaming time increases, this is because the jitter represent the mean jitter for all the frames in the video during streaming time. Figure 3 (A) illustrates the curve of jitter.



(a) jitter

(b) latency

(c) throughput

Figure 3. video streaming metrics.

From the previous table you can see that the latency and throughput are not depent on the streaming interval. see figure 3.b and 3.c respectivly. All experiments shows that Packet Loss Rate =0, Out-of-Order Packets=0, Datagram Size = 3372.0, Packet Delivery Ratio =100%. This is logical because there is no congestion on the network.

Conclosion

Video streamig is widly spreaded in multiple applications, streaming from source to destnation depends on many elements. One of these important elemnts is the network status during streaming. Jitter directly proportional with the streaming period. Latency and throughput are not related to the streaming period. When packet loos equals to zero this mean that delevary ration will be 100%.

Futuer works

This work can be expanded to measure the effect of add lode on the network during streaming process. It also can be expanded by changing the network to ad-hoc network and test the affect of nodes movig during streaming process. It also can be expanded to test the affect of streaming progressive video over such networks.

References

[1] Zhang, X., & Hassanein, H. (2010). Video on-demand streaming on the Internet — A survey. *2010 25th Biennial Symposium on Communications*, 88–91. https://doi.org/10.1109/BSC.2010.5472998.

[2] Chu, D., Jiang, C., Hao, Z., & Jiang, W. (2013). The Design and Implementation of Video Surveillance System Based on H.264, SIP, RTP/RTCP and RTSP. *2013 Sixth International Symposium on Computational Intelligence and Design*, *2*, 39–43. https://doi.org/10.1109/ISCID.2013.124.

[3] Kshirsagar, P., & Manoharan, H. (2021). Operational collection strategy for monitoring smart waste management system using shortest path algorithm. *Journal of Environmental Protection and Ecology*, *22*(2), 566–577.

[4] Liu, Y. and G. Y. and L. C. (2008). A survey on peer-to-peer video streaming systems. *Peer-to-Peer Networking and Applications*, *1*, 18–28.

[5] Alaya, B., Khan, R., Moulahi, T., & Khediri, S. El. (2021). Study on QoS management for video streaming in vehicular ad hoc network (VANET). *Wireless Personal Communications*, *118*(4), 2175–2207.

[6] Devi, V. S. and R. T. and P. S. B. (2020). Cluster based data aggregation scheme for latency and packet loss reduction in WSN. *Computer Communications*, *149*, 36–43.

[7] Taha, M., & Ali, A. (2023). Smart algorithm in wireless networks for video streaming based on adaptive quantization. *Concurrency and Computation: Practice and Experience*, *35*(9), e7633.

[8] Urgun, Y., & Kavak, A. (2016). SDN/NFV-based QoS mechanism for Cellular Core Networks. *2016 24th Telecommunications Forum (TELFOR)*, 1–4. https://doi.org/10.1109/TELFOR.2016.7818733.

[9] Ghasemi, C., Yousefi, H., & Zhang, B. (2021). Internet-scale video streaming over NDN. *IEEE Network*, *35*(5), 174–180.

[10] Wei, S., & Swaminathan, V. (2014). Low Latency Live Video Streaming over HTTP 2.0. *Proceedings of Network and Operating System Support on Digital Audio and Video Workshop*, 37–42. https://doi.org/10.1145/2597176.2578277.

[11] Usman, M. A., Usman, M. R., & Shin, S. Y. (2016). A no reference method for detection of dropped video frames in live video streaming. *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, 839–844. https://doi.org/10.1109/ICUFN.2016.7537155.

[12] Uitto, M., & Heikkinen, A. (2021). Evaluation of live video streaming performance for low latency use cases in 5g. *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, 431–436.

[13] Jin, L., Quan, W., Liu, G., Gao, D., Foh, C. H., & Wang, Q. (2020). DPS: A Delay-Programmable Scheduler for the Packet Out-of-Order Mitigation in Heterogeneous Networks. *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, 1–6. https://doi.org/10.1109/ICCWorkshops49005.2020.9145422.

[14] Lim, Y. (2022). Cost-Efficient Framework for Mobile Video Streaming using Multi-Path TCP. KSII Transactions on Internet and Information Systems (TIIS), 16(4), 1249–1265.

[15] Chen, Z., Zhou, W., Wu, S., & Cheng, L. (2020). An Adaptive on-Demand Multipath Routing Protocol With QoS Support for High-Speed MANET. *IEEE Access*, *8*, 44760–44773. https://doi.org/10.1109/ACCESS.2020.2978582.

[16] Xia, W. (2020). Cross-layer optimization technology for wireless network multimedia video. *Компьютерная Оптика*, *44*(4), 582–588.

[17] Chao, M., Stoleru, R., Jin, L., Yao, S., Maurice, M., & Blalock, R. (2020). AMVP: Adaptive CNN-based multitask video processing on mobile stream processing platforms. *2020 IEEE/ACM Symposium on Edge Computing (SEC)*, 96–109.

[18] Lee, K., Yi, J., Lee, Y., Choi, S., & Kim, Y. M. (2020). GROOT: a real-time streaming system of high-fidelity volumetric videos. *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. https://doi.org/10.1145/3372224.3419214 .

[19] YU SHI; Jean-Hun Chung. (2021). A Case Study on Real-time Live Video Streaming Content. *Journal of Digital Convergence*, *19*(7), 251–260.

[20] Jinrong Yang, S. L. Z. L. X. L. J. S. (2022). Real-time Object Detection for Streaming Perception. *ArXiv*.

[21] Lin Geng Foo, J. G. Z. F. J. L. (2023). System-status-aware Adaptive Network for Online Streaming Video Understanding. *ArXiv*, *1*.