Research Directions for Network Based Video Streaming with Emphasis for Live Screen Mirroring

SC Gajanayaka[1], DMR Kulasekera[2] and ADAI Gunasekera[3]

[1-3] *General Sir John Kotelawala Defence University, Sri Lanka*
[#] Corresponding author; sinhalokaya@gmail.com

Abstract *— Recent advancements in the field of embedded systems and network speeds in general – particularly the realization of concepts of image processing, live video streaming and Internet of Things have redefined the technology world, as we knew years ago. In terms of video streaming, the rapid increase of speeds of networks around the world have enabled fast and efficient streaming of live content. Embedded systems have made it simpler to build portable devices which can stream live video via a network. This review analyses these developments in detail, outlining the potential left for development. In doing so, the paper looks at techniques and technologies that have been used for streaming videos over networks. Special emphasis is placed on using the available technologies for screen mirroring and live video broadcasting. Development in these concerns will assist in developing lightweight wireless systems for TV stations to personal video uploading in social media.*

Keywords — **wireless video transmission, embedded systems, screen mirroring, live collaboration tools**

I. INTRODUCTION

Video transmission over networks has been a contended topic of research ever since the birth of the Internet. Though many technologies have developed over time, one significant shortcoming which marred the propagation and the popularisation of network video streaming is the practical issue of low bandwidth in networks. In the current sphere, however, most countries have achieved significant network bandwidths with the development of ADSL, 3G and 4G networks around their countries. In such a context, live video streaming has become a crucial area of research as it now involves the added concern of managing millions of connected devices at once. It is predicted that there will be over 25 billion connected devices by the year 2020 ("The Evolution of Mobile Technologies," 2014).

As of 2017, major corporations are focussing strongly on live video streaming. As such, "streaming" is moving into the mainstream. This focus has resulted in apps like Periscope ("Periscope - Live Video - Android Apps on Google Play," 2017), YouTube and Facebook Live which have made it easy for anyone with a smartphone to start streaming. They have normalized the idea of live streaming video and as a result millions of people are now used to this type of content. That growing ubiquity represents an opportunity for other businesses. Live video can be a powerful tool for marketing, sales, training, entertainment, or media, as it is a compelling technology for everyone, in effectively transmitting a message of pages in a few seconds ("Youtube Live vs Facebook Live compared to Online Video Platforms," 2017).

Though this is the status quo, the birth of all these technologies run years behind. This paper thus discusses the development of wireless (online) video streaming technologies and engages in a comparative analysis of the same.

Another important aspect discussed in this paper is the parallel development of live collaboration software tools. These tools have been the artefacts to show how much technology has triumphed in its application. Online collaboration tools have become instrumental in many areas including business and education.

Though technology has developed at an alarming rate, one underlying factor of limitation when it comes to streaming live video is the bandwidth problem. To overcome this, either smaller packets of data containing the video data should be transmitted or more effective algorithms must be used. These however should be done in such a way to ensure that the quality of the video is not diminished.

This paper therefore looks at the above-mentioned aspects and how these technologies can be integrated to deliver higher quality streaming experiences to the end user. The essential hypothesis is to see if such integration is possible within technologies available as of today and to assess potential issues associated with the same. The objective of this research is therefore, to identify the applicability of network based video streaming for live screen mirroring. In general, the paper analyses related aspects, as mentioned above, in detail.

The paper is divided into 3 main sections – literature review – which highlights the research and the journey thus far in above technologies, discussion – analysis of the contents of the literature review, and finally the conclusion – concluding remarks as to what developments may be undertaken by future researchers.

II. LITERATURE REVIEW

A. Video Transmission Techniques

Firstly, literature clearly shows that there has been a constant interest to research about the various video transmission techniques which can be deployed for network based video transmission. Networks function in layers in which a data packet goes through a stack of different layers which perform distinct functions to the data packet. These layers include both hardware and software. Therefore, it could be seen that optimisation for video transmission could be achieved at both a hardware level and at a software level. One of the first approaches is seen in (Cherkassky et al., 2002), where the project focuses on the issues in wireless transmission of image and video data – video compression and network prioritization which are the key factors to consider when making efficient use of the limited bandwidth. In (Shan, 2005), a network layer-based video transmission technique is discussed which uses equal-sized radio link protocol (RLP) packets.

Quite an extensive analysis and a mechanism into "optimizing" video streaming for wireless transmission is discussed in (Lu, 2009), where a method described as "PROTAR" has been used. This again is a layer-based approach. A packet scheduling algorithm for wireless video streaming has been discussed in (Kang and Zakhor, 2002), which is based on unequal deadline thresholding. First, the proposed video packet scheduling algorithm is efficient in achieving unequal loss rate between video packets with different importance. Second, the motion-texture discrimination is more efficient for large motion clips and small quantization step, i.e., large size of texture fields. Though a key assumption in developing this algorithm has been fixed round trip time, it is a safe assumption to make in today's context of high speed networks. Wireless network modes are analysed in depth in (Ketkar et al., n.d.), where they claim that WiMAX could be more efficient to transmit videos over networks as opposed to Wi-Fi and ADSL. A voice and video over Wireless LAN technique is discussed in (Iyer et al., 2013) and the results show how a WLAN is in fact more efficient than using a LAN for intra-organization communication of video and voice. In (Ramya et al., 2015), a system has been devised to act as a framework to stream cloud based video streams to a "CCMN" – Cloud-Centric Media Network. This is capable of a multi-screen application but what it does it telecasting "one screen" to many other screens and not vice versa.

Another popular protocol proposed for video transmission is Transmission Control Protocol. TCP is widely used by commercial video streaming systems. When a packet has not arrived by its playback time, a typical practice in these commercial systems is that the client simply "stops and waits" for this packet, and then resumes playback. This stop-and-wait playout strategy is easy to implement. However, stopping playout due to late packet arrivals renders the viewing experience unsatisfactory. A continuous playout strategy, i.e., continuing playout regardless of late packet arrivals, also leads to unsatisfactory viewing experience, since late packet arrivals cause glitches in the playback. The performance of both the stop-and-wait and the continuous playout strategies therefore depends on the frequency of late packet arrivals during the playback of the video. In (Wang et al., 2003), where a model is proposed overcome these issues, discrete-time Markov models have been developed to evaluate the performance of live and stored video streaming using TCP. Based on these models, guidelines have been provided as to "when" using TCP for streaming leads to satisfactory performance. A crucial finding in this research has thus been that the fraction of late packets is similar for long videos in constrained streaming while it decreases with the video length (after a short duration of increasing trend at the beginning of the playback) in unconstrained streaming.

A more recent development has been the RTS (Real Time Streaming) Protocol. The RTSP protocol is based on the HTTP protocol, and is commonly used to manage media content streaming. This protocol doesn't directly deal with the streaming content, but uses the RTP protocol to handle content transmissions. (Peltotalo et al., 2010) discusses a peer-to-peer streaming application using this protocol. The effective real-time P2P streaming system for the mobile environment presented in this paper is an alternative solution to traditional client-server-based streaming applications. However, it also highlights that more advanced laboratory tests with different latencies and throughputs between peers are still needed to highlight system bottlenecks and usability issues.

RTMP (Real-Time Messaging Protocol) is also a widely-used protocol nowadays for live video transmission. RTMP is the Adobe's network protocol used to transmit audio, video and data between its Flash platforms (Adobe Incorporated, 2011). RTMP consists of two important structures, namely Message and Chunk. RTMP belongs to the application-level protocol, and usually TCP is accompanied with it as the transport-level protocol. The basic unit of the RTMP transmitting information is the "Message". During transmission, for consideration of multiplexing and packetizing multimedia streams, each Message will be split into some "Chunks" (Lei et al., 2012).

Another example of P2P (Peer to Peer) streaming is seen in (Tran et al., 2003), where first, an end-to-end transport architecture for multimedia streaming over the Internet is presented. Second, a new multimedia streaming TCPfriendly protocol (MSTFP), which combines forward estimation of network conditions with information feedback control to optimally track the network conditions, is discussed. As the said protocol is TCP based, it would still have the inherent issues which TCP communication has always had.

In (Seeling et al., 2004), a discussion is available on how to evaluate the network performance for single-layer and two-layer encoded video using "traces". As (Zhu and Girod, 2007) notes, in network video transmission, cross-layer information exchange is required, so that video source rates can adapt to the time-varying wireless link capacities. They note that many problems remain, particularly in the context of wireless mesh networks – for instance, it is still unclear whether the stringent latency constraint (usually less than a second) for video streaming can be met when packets need to be delivered over multiple hops of time-varying wireless links in a mesh network. Typically, the wireless network is shared by both video streaming and other applications such as file downloading. The problem remains to be addressed as how to optimally allocate network resources among heterogeneous traffic types, each bearing a different performance metric (e.g., completion time for file downloading versus video quality for streaming). For example, in a case of where a wireless projecting device is connected to a machine, there is a possibility that the network being used to be used for various other purposes too (Zhang et al., 2001), while broadcasting the screen to the projector. Such a situation can surely limit the bandwidth required to broadcast the screen at a higher frame rate of more than 25-30 fps. Even more problematic is the situation where "many" clients would be trying to connect to the projector for multi-screen sharing. This problem presents a totally new research problem of not only having to stream the videos fast, but also having to combine them in an effective manner. Back in 2001, as per (McCrohon et al., 2001), it was predicted that it would be possible to stream live lectures, "with expected future developments in networking technology, the quality of streamed video will soon be of even higher quality ensuring video streaming a promising role in the delivery of online education.". However, even with technologies available 15 odd years later, it is still doubtful if a "real-time" output can be achieved – to transmit the video feeds of many device and combine them in one device.

Dynamic Adaptive Streaming over HTTP (DASH) (also known as MPEG-DASH) is quite a modern technology which is now used for high quality streaming of media content over the Internet delivered from conventional HTTP web servers. Like Apple's HTTP Live Streaming (HLS) solution, MPEG-DASH works by breaking the content into a sequence of small HTTP-based file segments, each segment containing a short interval of playback time of content that is potentially many hours in duration, such as a movie or the live broadcast of a sports event. This is probably the best solution that is available for live streaming of content at the moment. The MPEG DASH standard was published by the ISO in April 2012 (Andy Salo, 2012).

A comparative evaluation of most of these techniques is seen in (Aloman et al., 2015). They have conducted a comparative performance evaluation of MPEG DASH, RTSP, and RTMP streaming protocols over 4G and Wi-Fi (IEEE 802.11g/n) real networks in terms of QoE (Quality of Experience), tested both video on demand and live video streaming. Results in this analysis has suggested that RTSP is more efficient than MPEG DASH for starting the video playback, but at the expense of decreasing QoE due to packet losses. In addition, the long pre-loading time interval needed by MPEG DASH or RTMP permits to alleviate the impact of the packet losses which take place during the transmission, as revealed by a lower number of re-buffering events for these two protocols. MPEG-DASH surely the best available protocol currently, for streaming content such as TV series and / or live events. This is for requirements that demand both high quality and speed at the same time. However, this obviously requires a network with high bandwidth. Therefore, for other lesser needs, specifically for purposes of screen mirroring and such, RTMP is a lucrative option.

B. IoT Based Embedded Systems for Video Transmission

Secondly, we should look at the developments of the IoT sphere as well. Transmitting video over the web using embedded devices has surely been an area of crucial significance for the development of the IoT ecosystem. Most of the integrations among various technologies have been possible due to the rapid accession and research on development boards such as Raspberry Pi and Arduino.

A classic integration of embedded systems with IoT and image processing is seen in (Kulkarni et al., 2014). Here a surveillance robot has been developed which is capable to transmit a video stream over the web. To be able to do this on a simple chipset such as Arduino, clearly shows that complex systems could now be implemented using available technologies. Image processing has also become more viable on a Pi as it contains both a Graphics Processing Unit (GPU) along with an internal memory. This aspect has been explored in (Ujjainiya and Chakravarthi, 2006),where the system is used to detect the objects in front of a vehicle by using a camera module. The camera detects the picture and operation is performed in OpenCV to detect the edges of the detected picture. Another image processing example is seen in (Manasa et al., 2015), where the paper proposes the design and implementation of object counting algorithm based on image processing by using the RPi on real time basis. RPis are also ideal for the construction of surveillance systems both in-house and remote. This is explored in (Chuimurkar and Bagdi, 2016), where a video stream is transmitted via the web to a mobile device.

A recent device which professes IoT capabilities is the Intel® Compute Stick shown in Figure 1 (Pete Carey, 2015), which came to the market as late as 2016. This is a complete computer in a "stick" with a HDMI port. This too, would be ideal for streaming video via a wireless method, but the only drawback of this device is that it is quite costly compared to the RPi. Research done using this device is yet to be seen and as thus, the reliability and performance is yet to be assessed.

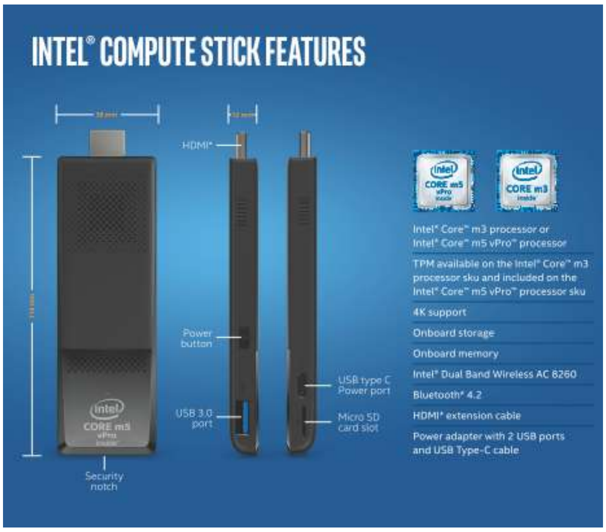

Figure 1: Intel Compute Stick features

C. Screen Mirroring / Projection

Nowadays, screen mirroring devices (McGill et al., 2014) have been built by integrating the wireless video transmission techniques discussed above and using various chipsets. There is also a possibility where projectors will soon become completely wireless, minimizing the many practical restrictions (not being able to project more than one screen, issues setting up wire cords etc.) that traditional projectors undergo. Therefore, a review of these available devices is apt.

We now look at existing projectors out in the market. The existing projectors provide the basic functionality of where one wired device can be projected. In the last 3-4 years, many projectors, however, have been developed with the capabilities of WLAN (BenQ, n.d.) (Epson, n.d.), meaning the projector can connect to a client using a wireless mode. A few of these are in fact capable of screen sharing like most of the Epson® PowerLite versions using the Epson iProjection™ App (Epson, 2015). These algorithms and implementations are proprietary and allow for a limited scope of application.

Apart from these commercial products, there is little literature available on projects conducted at an academic level relating to projectors itself. This is probably due to the nature of the projectors being predominantly a commercial device. However, a couple of research projects is to be noted. In (Chheda et al., 2013), a Raspberry Pi has been used to transmit the video feed from a desktop to the Pi. The mode of transmission is Wi-Fi and this could be identified as a basic implementation of media streaming over a network. No "enhancing" procedure is available in transmitting the frames and this project depicts and predicts (in further research), how the same concept could be used to create an adapter-like device for already available projectors.

When considering the technical aspect, it could be seen that there are many projectors available in the market deploying one or more of the above technologies intended to achieve real-time wireless video transmission. However, most of the companies who have involved themselves in developing projectors of this sort have provided a profound description of the technicalities of the same. Leading projector brands such as BenQ (BenQ, n.d.), Optoma (Optoma, n.d.), Epson (Epson, n.d.), Panasonic (Panasonic, n.d.) and Acer (Katie Scott, 2008) seem to have built wireless projectors as of late. A few of the models offered by these brands have the "multi-screen" split feature (Shown in Figure 2). Many include an app for mobile devices (Android and iOS), and come equipped with proprietary algorithms. The NEC MultiPresenter Stick (NEC, 2016) is a 2016 device with quite sophisticated features but it still requires a NEC compatible projector to work with.

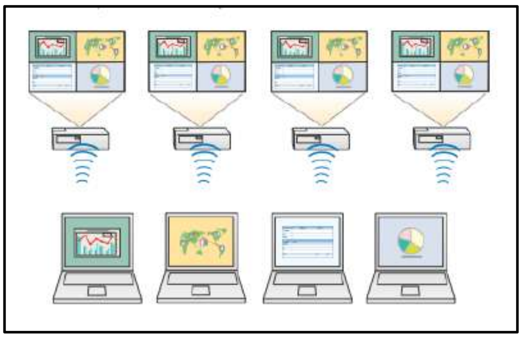

Figure 2: Application of screen-split option in a wireless projector

However, as it may be, there has not yet been a model capable of connecting to the internet to pull a feed from a device not located in the current location. Furthermore, almost all these models use proprietary software meaning they are only compatible with the set list of devices produced by the manufacturer. Also, there is no way that a customer with a projector with no wireless capabilities can enjoy these features unless s/he buys a new projector.

D. Online Collaboration Tools

Next is to consider commercial software solutions which address the need of online collaboration. Many tools are available such as GoToMeeting ("GoToMeeting," n.d.), Skype for Business (Skype, n.d.), Join.me (Join.me, n.d.), screenleap.com (screenleap, n.d.) and so forth. A few of these provide multi-user interaction while some provide interfaces for customized viewing such as manipulating PDF files, presentation slides etc. All these solutions, however, limit their usability and reliability on the availability of their proprietary systems. A summary of these systems is seen in Figure 3:

Software Platform	Multi user	Dependant on a web platform?	Image optimisation techniques
Skype for Business	Yes	Yes	No
Join.me	No	Yes	No
Screenleap.com	No	Yes	No
TeamViewer	Yes	Yes	No

Figure 3: Comparison of software solutions for screen sharing

E. Pre-processing Before Streaming

Is it also important to look at research that has gone into pre-processing image frames before being streamed via a network. This means that the streams received by whatever device (from clients) will undergo a stage of pre-processing to "enhance" the frames received. This may vary from changing basic properties such as contrast, brightness, mid-tones and so forth. This could be achieved via existing graphic processing libraries such as OpenCV (open source - ("OpenCV library," n.d.)) and CUDA (proprietary - ("CUDA Zone," 2015)). Both libraries are quite powerful in their own ways.

(Di Salvo and Pino, 2011) has highlighted how applications have been parallelized in various areas using CUDA, to achieve very high-performance in time processing keeping the same performance in terms of accuracy. They further note that the main portion of the available CUDA based approaches deal with the parallelization of generic image processing operations, whereas much more work should be done in the biomedical and video-surveillance fields, where this is mostly used. OpenCV on the other hand is more prominently used for "detection purposes" in videos. This is shown in many research literature including (Pulli et al., 2012), (Shah, 2014) and (Farhadi-Niaki and Mehrvar, n.d.).

III. DISCUSSION

The methodology of this research centres around the performance measures of different video transmission techniques and how each performs for the purpose of live screen mirroring. Such factors include transmission speed, reliability, quality of decoding, platform dependability, scalability and usage of image optimisation techniques.

Firstly, when referring to the status quo of video transmission protocols MPEG-DASH and RTMP seem to be two protocols which are in the forefront when it comes to real-time video transmission.

Another concern is "how" the video frames should be transmitted. Should they be transmitted as "frame by frame" or should it be after a pre-compression technique has been applied such as after converting the stream to a format like "mp4"? What are the already available packages which can do this? "FFmpeg" ("FFmpeg," 2017) is one such strong open source library available for streaming audio / video which supports both above techniques (Wan and Dai, 2016). This easily supports Linux based systems and thus runs fine on embedded chipsets discussed here. Furthermore, this library also supports mobile platforms such as Android. In (Fu et al., 2010) an Android-based codec application is designed and implemented using Java with FFmpeg. Therefore, this would be an ideal solution for anyone planning to implement a video transmission algorithm.

Secondly, what is a possible embedded system which can be optimized for video transmission? In comparison, Arduino has limited support for video streaming. ComputeStick is surely a high-end option which supports video streaming but when looking at other concerns such as the cost and cross platform capabilities, the ideal chipset for mini scale projects, specifically for screen mirroring concerns would be the Raspberry Pi. Another similar concern is what methodology is most appropriate for network connectivity. For this, the most intuitive solution would be to use the technology which offers the "fastest" bandwidth. However, it should be noted that video transmission does not solely depend on the speed of the network, but rather on concerns of priority accorded to video packets. WiMAX + LTE technologies warrant the most optimized connectivity (Iyer et al., 2013) for video streaming.

Thirdly, when looking at the aspect of screen mirroring, as discussed, most projectors are now becoming wireless. It is a matter of time for all our devices to come void of HDMI / VGA ports. More academic research should be undertaken in this area, though, to ensure that the most optimized methods of screen mirroring maybe identified and studies done, as opposed to comparing commercial solutions, due to the restrictions posed via proprietary algorithms. Further research should look at extending its functionality to intelligently analyse the video streams received and provide some basic image corrections (Sukthankar and Mullin, 2000) before it is broadcast.

IV. CONCLUSION

This paper discussed, at length, the various advancements in the fields of wireless video transmission techniques, IoT and embedded systems which can facilitate such techniques, screen mirroring techniques and various pre-processing that can be done to livestreams using image processing algorithms. Though one may claim these are distinct research areas, that is not the case in reality. Many real-world applications are being modelled using these aspects. The development of chipsets has now allowed a massive level of data manipulation, graphics data processing and storage abilities. It is important that these chipsets are used to achieve their true potential by researchers and it could be said that much has been already done. However as mentioned, there is room for improvement in integrating all these technologies to build useful embedded systems for day-today use. Such research need to be undertaken with the constant thought in mind that the limitation of network bandwidth should not hinder the performance and speed of the streams being broadcast over a network.

REFERENCES

Adobe Incorporated, 2011. Adobe® Flash® Media Server 4.5.

Aloman, A., Ispas, A.I., Ciotirnae, P., Sanchez-Iborra, R., Cano, M.D., 2015. Performance Evaluation of Video Streaming Using MPEG DASH, RTSP, and RTMP in Mobile Networks. IEEE, pp. 144–151. doi:10.1109/WMNC.2015.12

Andy Salo, 2012. MPEG DASH: A Technical Deep Dive and Look at What's Next.

BenQ, n.d. BenQ W1500 Home Entertainment Wireless Projector [WWW Document]. Cable Free Easy Set-Up. URL http://www.benq.com/product/projector/W1500/features/

Cherkassky, V., He, X., Balasubramanian, P., 2002. Wireless Transmission of Image and Video Data.

Chheda, D., Darde, D., Chitalia, S., 2013. Smart Projectors using Remote Controlled Raspberry Pi. Int. J. Comput. Appl. 82.

Chuimurkar, M.R., Bagdi, V., 2016. Smart Surveillance Monitoring And Automation System Using Raspberry PI and PIR Sensor.

CUDA Zone [WWW Document], 2015. . NVIDIA Dev. URL https://developer.nvidia.com/cuda-zone (accessed 3.26.17).

Di Salvo, R., Pino, C., 2011. Image and video processing on CUDA: state of the art and future directions. MACMESE 11, 60–66.

Epson, 2015. Epson iProjection™ App [WWW Document]. Epson IProjection™ App. URL https://epson.com/wireless-projector-app

Epson, n.d. Epson US [WWW Document]. Epson Wirel. Proj. URL https://epson.com/how-wireless-projectors-work

Farhadi-Niaki, F., Mehrvar, H., n.d. A Review on "Implementation of Real-time Image and Video Processing, and AR on Mobile Devices."

FFmpeg [WWW Document], 2017. URL https://ffmpeg.org/ (accessed 3.26.17).

Fu, X., Wu, X., Song, M., Chen, M., 2010. Research on audio/video codec based on Android, in: Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on. IEEE, pp. 1–4.

GoToMeeting [WWW Document], n.d. . GoToMeeting. URL https://www.gotomeeting.com

Iyer, A., Badgujar, A., Eaga, M., Iyer, R., 2013. Voice and Video over Wireless LAN. Int. J. Sci. Res. Publ. 3, 1–5.

Join.me, n.d. SHARING IS CARING. AND SWAPPING. AND EDITING. [WWW Document]. URL https://www.join.me/screen-sharing

Kang, S.H., Zakhor, A., 2002. Packet scheduling algorithm for wireless video streaming, in: International Packet Video Workshop.

Katie Scott, 2008. Acer P5260i wireless projector.

Ketkar, O., Ninawe, P., Panchal, S., Sonar, Y., n.d. VIDEO TRANSMISSION USING WIRELESS NETWORK.

Kulkarni, C., Grama, S., Suresh, P.G., Krishna, C., Antony, J., 2014. Surveillance Robot Using Arduino Microcontroller, Android APIs and the Internet, in: Proceedings of the 2014 First International Conference on Systems Informatics, Modelling and Simulation. IEEE Computer Society, pp. 107–111.

Lei, X., Jiang, X., Wang, C., 2012. Design and implementation of streaming media processing software based on RTMP, in: Image and Signal Processing (CISP), 2012 5th International Congress on. IEEE, pp. 192–196.

Lu, M.-H., 2009. Optimizing transmission for wireless video streaming. Ozyegin University.

Manasa, J., Pramod, J.T., Jilani, S.A.K., Hussain, M.S.J., 2015. Real Time Object Counting using Raspberry pi.

McCrohon, M., Lo, V., Dang, J., Johnston, C., 2001. Video streaming of lectures via the Internet: An experience, in: ASCILITE. pp. 397–405.

McGill, M., Williamson, J., Brewster, S.A., 2014. Mirror, mirror, on the wall: collaborative screen-mirroring for small groups. ACM Press, pp. 87–94. doi:10.1145/2602299.2602319

NEC, 2016. NEC MultiPresenter Stick [WWW Document]. URL http://www.projectorpoint.co.uk/meeting-room-technology/presentation-systems/nec-multipresenter

OpenCV library [WWW Document], n.d. URL http://opencv.org/ (accessed 3.26.17).

Optoma, n.d. Wireless [WWW Document]. URL https://www.optoma.co.uk/projectortechnology/Wireless

Panasonic, n.d. Wireless Projector for iOS [WWW Document]. URL http://panasonic.net/avc/projector/ios/

Peltotalo, J., Harju, J., Väätämöinen, L., Bouazizi, I., Curcio, I.D.D., 2010. RTSP-based Mobile Peer-to-Peer Streaming System. Int. J. Digit. Multimed. Broadcast. 2010, 1–15. doi:10.1155/2010/470813

Periscope - Live Video - Android Apps on Google Play [WWW Document], 2017. URL https://play.google.com/store/apps/details?id=tv.periscope.android&hl=en (accessed 4.28.17).

Pete Carey, 2015. Review: Intel's computer-on-a-stick: Awesome idea, but its day hasn't arrived [WWW Document]. URL https://phys.org/news/2015-06-intel-computer-on-a-stick-awesome-idea-day.html (accessed 3.25.17).

Pulli, K., Baksheev, A., Kornyakov, K., Eruhimov, V., 2012. Real-time computer vision with OpenCV. Commun. ACM 55, 61–69.

Ramya, R., Thendral, G., ME, M.T.K., 2015. EFFICIENT PARALLEL FRAMEWORK FOR ADAPTIVE VIDEO STREAMING IN CLOUD SOCIAL TV. Asia Pac. J. Res. Vol Issue XXIV.

screenleap, n.d. Screenleap [WWW Document]. URL http://www.screenleap.com/

Seeling, P., Reisslein, M., Kulapala, B., 2004. Network performance evaluation using frame size and quality traces of single-layer and two-layer video: A tutorial. IEEE Commun. Surv. Tutor. 6.

Shah, S., 2014. Real-time image processing on low cost embedded computers. Techincal Rep. No UCBEECS-2014–117.

Shan, Y., 2005. Cross-layer techniques for adaptive video streaming over wireless networks. EURASIP J. Adv. Signal Process. 2005, 871928.

Skype, n.d. Skype: Group screen sharing [WWW Document]. Group Screen Shar. URL https://www.skype.com/en/features/group-screen-sharing/

Sukthankar, R., Mullin, M.D., 2000. Automatic keystone correction for camera-assisted presentation interfaces, in: Advances in Multimodal Interfaces—ICMI 2000. Springer, pp. 607–614.

The Evolution of Mobile Technologies, 2014.

Tran, D.A., Hua, K.A., Do, T., 2003. Zigzag: An efficient peer-to-peer scheme for media streaming, in: INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies. IEEE, pp. 1283–1292.

Ujjainiya, L., Chakravarthi, M.K., 2006. Raspberry–Pi Based Cost Effective Vehicle Collision Avoidance System Using Image Processing.

Wan, F., Dai, Y., 2016. The Research of video format conversion based on educational video-on-demand.

Wang, B., Kurose, J., Shenoy, P., Towsley, D., 2003. A model for TCP-based video streaming. Univ. Mass. Tech. Rep.

Youtube Live vs Facebook Live compared to Online Video Platforms [WWW Document], 2017. URL https://www.dacast.com/blog/comparing-youtube-live-and-facebook-live-with-pro-video-streaming/ (accessed 4.28.17).

Zhang, Q., Zhu, W., Zhang, Y.-Q., 2001. Resource allocation for multimedia streaming over the Internet. IEEE Trans. Multimed. 3, 339–355.

Zhu, X., Girod, B., 2007. Video streaming over wireless networks, in: Signal Processing Conference, 2007 15th European. IEEE, pp. 1462–1466.