IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, July 2026
Impact Factor: 8.2

VirtualCon: A WebRTC-Based Peer-to-Peer Architecture for Real-Time Video Conferencing with Collaborative Tools

Vednidhi[1] and Dr. Priyanka[2]
Student, Department of Computer Science and Engineering[1]
Assistant Professor, Department of Computer Science and Engineering[2]
Ganga Institute of Technology & Management, Kablana
shannukumar7988@gmail.com[1], Priyachillar30@gmail.com2

Abstract: *Today, the use of real-time video conferencing is considered the backbone of working from home, learning at a distance, and providing healthcare services, among other things, but in many cases, the mainstream platforms are based on centralized media servers that involve expenses, complexities, and privacy compromises. In this paper, we propose VirtualCon, an end-to-end browser-based, peer-to-peer (P2P) video conferencing architecture that does not rely on the server-side media routing for small group video conferences and is completely implemented on WebRTC. VirtualCon comes with a small node.js/ socket.IO signaling layer is used for session bootstrapping only, and media and collaborative data (such as shared whiteboard, real-time chat and screen sharing) are directly communicated between the peers via encrypted SCTP data channels. VirtualCon under controlled network emulation is tested with a headless set-up of browser clusters, and connection set-up time, end-to-end latency, packet loss and resource consumption are measured as the number of groups can scale from 2 to 10 participants. Results demonstrate that VirtualCon is able to maintain sub-80ms latency and fast connection establishment for groups of 4 or less, but performance declines dramatically for groups much greater than 5 because of the quadratic growth of mesh connections. These results reveal the actual limit of the number of nodes that can perform pure P2P mesh conferencing and encourage the development of a hybrid approach between P2P mesh and SFU for a higher number of nodes. VirtualCon shows that the end-to-end encrypted conferencing and its collaboration features are possible in a decentralized manner for privacy-focused small group scenarios like telemedicine consultations, secret team discussions, and more.*

Keywords: Skin cancer detection, CNN-LSTM, Grad-CAM, explainable AI, HAM10000, dermoscopy, EfficientNetB3.

I. INTRODUCTION

A. Background

Overall, the rise of remote work, digital education, and telemedicine is transforming the nature of professional and academic interactions with a real-time, video-conferencing paradigm. This change has created a new set of limitations in the traditional client-server methods of communication and thus has prompted the need for web-native, low latency architectures. An important new development for real-time peer-to-peer communication is the introduction of WebRTC, which has become the de-facto standard in using the browser to exchange media and data without relying on intermediate servers in the case of base-level communication [1]. Full production of WebRTC, however, involves a number of choices and compromises between various media topologies. Peer to peer mesh networks provide private communication without the need for any infrastructure or operators, but are limited by bandwidth and processing power, which increase with the number of users in the network [4]. Centralized solutions like Multipoint Control Units and Selective Forwarding Units can decrease load at the clients, but have a cost in infrastructure, complexity in deployment, and vulnerabilities in encryption at the server [12].

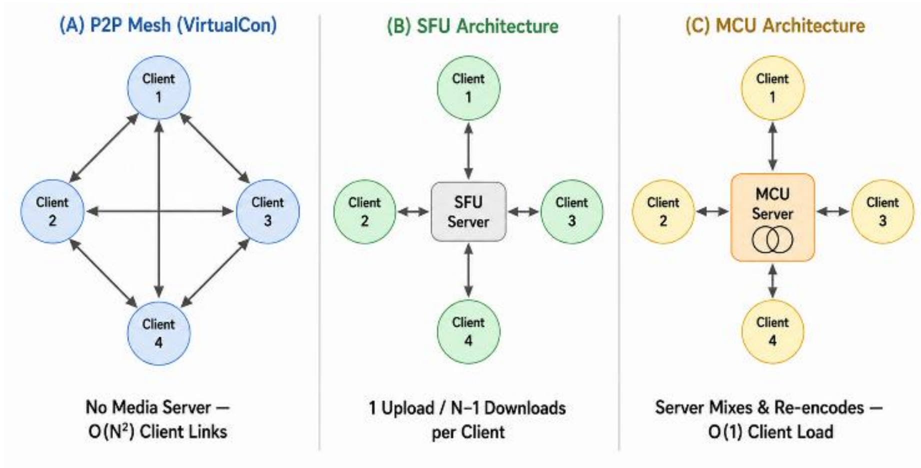

Fig 1. VirtualCon Mesh Topology vs. Centralized Conferencing Architectures

B. Problem Statement and objectives

This research aims at designing, implementing and testing VirtualCon which is a completely decentralized WebRTC based conferencing system which does not require any external plugin and works within web browser itself. The goals are three-fold: designing a lightweight, server-less P2P-mesh system to coordinate multiparty sessions using an efficient signaling protocol; making screen sharing, messaging and collaborative whiteboarding features a first-class citizen in peer connections using WebRTC SCTP data channels [9] and testing the performance of the P2P-mesh under simulated network constraints to determine the scalability boundaries due to client hardware and bandwidth as a function of group size.

C. Significance and Contributions

But enterprise platforms use expensive media routers for large meetings, and there's a definite need for lightweight, self-hosted systems that would work for smaller, more privacy-oriented meetings. VirtualCon fills this void with a multi-channel, decentralized architecture that enables the synchronization of media alongside collaborative data channels, a dual-mode whiteboard synchronisation protocol which provides a balance of reliability and low latency and an empirically based scalability analysis that determines the point at which the hybrid mesh-to-SFU escalation is needed [10].

II. LITERATURE REVIEW

A. Literature Review

WebRTC is a hybrid of many protocols that allow you to send media and data in real time in a secure manner between browser sandboxes. Most of the terminals are located behind NAT and firewall devices and cannot be reached directly from end to end; to find out which route is possible, WebRTC has introduced Interactive Connectivity Establishment [11]. ICE uses two extra technologies to enable NAT traversal: Session Traversal Utilities for NAT, which lets a client know its public address, and Traversal Using Relays around NAT, which lets a client act as a relay when symmetric NAT or strict firewalls prohibit direct UDP routing. Traversal statistics collected from reports have shown that STUN is successful for the vast majority of sessions, with a smaller but stable number of sessions relying on TURN relays to stay connected [11]. Once a session path is set up, the media streams are encrypted with Secure Real-time Transport

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, July 2026 Impact Factor: 8.2

Protocol using Datagram Transport Layer Security and the data exchange is done through WebRTC data channels, also encapsulated in DTLS, in order to provide low-latency and encrypted transport of data [9] [23].

Table I: Summary of Related Literature.

Ref no.	Study Title	Authors	Study Year	Key Findings
[2]	An Empirical Study of WebRTC Congestion Control under Variable Network Conditions	A. Czerwiec et al.	2022	Analyzed WebRTC's Google Congestion Control behavior across fluctuating bandwidth and loss conditions, showing that delay-based rate adaptation stabilizes throughput but reacts slowly to sudden network shifts.
[10]	Decentralized Video Architectures: Exploring WebRTC P2P and XDN Scalability	R. Davidson and C. Wang	2023	Compared P2P mesh and Experience Delivery Network topologies, finding that pure mesh architectures scale poorly beyond small groups while maintaining superior end-to-end encryption guarantees.
[11]	NAT Traversal and Session ICE Candidate Gathering Performance under Restrictive Firewalls	S. Holdobin and V. Hrunovych	2022	Evaluated ICE/STUN/TURN performance under restrictive enterprise firewalls, reporting that a notable share of sessions require TURN relaying when symmetric NAT blocks direct UDP paths.
[12]	Comparative Performance of Mesh, SFU, and MCU Video Conferencing Architectures	A. Birke and T. Han	2024	Benchmarked the three major conferencing topologies, confirming that mesh networks impose quadratic client-side bandwidth growth while SFU and MCU shift load to centralized infrastructure.
[23]	SCTP User Message Interleaving Integration and Validation	F. Weinrank, I. Rüngeler, M. Tüxen, and E. Rathgeb	2021	Validated SCTP-based message interleaving for WebRTC data channels, demonstrating reduced head-of-line blocking for concurrent low-latency data streams.
[24]	eXpressSFU: Toward Super-Scalable Video Conferencing with SmartNICs	T. Tran et al.	2026	Proposed SmartNIC-offloaded SFU media routing to reduce server-side CPU overhead, enabling conferencing systems to scale to substantially larger participant counts than software-only SFUs.

B. Conferencing Architectures

More than one participant can be connected in a single real-time session, but there are three different networking topologies: peer-to-peer mesh, Selective Forwarding Units or Multipoint Control Units, each with its own bandwidth allocation, server load and latency benefits and drawbacks [12]. Each participant in a mesh will have a dedicated peer connection with each other participant; media servers will not be needed, but there will be N-1 encoders in each client and N-1 decoders in each client and the bandwidth demand is quadratic in the number of peers N in the session [4]. A Selective Forwarding Unit, on the other hand, functions as a media router: each peer uploads one stream to the server, which forwards without decoding or mixing to all other participants, thus saving a constant upload cost, but needing a linearly increasing download cost per number of participants [24]. A Multipoint Control Unit can do even more - decode, mix and re-encode all incoming streams into one single consolidated stream per user, and reduce the bandwidth and CPUs required on the client side - but create a considerable amount of processing overhead on the server side and

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, July 2026 Impact Factor: 8.2

add latency [12]. These architectural choices immediately lead to the scaling limits which are measured further along this paper.

C. Related Systems and Gaps

Scalability to hundreds of concurrent users is achieved by enterprise platforms like Zoom, Google Meet, Microsoft Teams, and Jitsi using Selective Forwarding Unit or hybrid architectures. However, this model has a number of obvious drawbacks on the side of the light-weight and privacy-sensitive application: first, there is a need for continued computation and operation to run the centralized media servers, secondary, collaborative tools have to be transmitted on top of a separate signaling channel, thus risking mismatch with media streams [7] and thirdly, centralized media routers have to decrypt the transmission packets in order to perform transcoding or recording operations, thus breaking the end-to-end encryption guarantees [10]. VirtualCon overcomes these constraints by using a decentralized architecture that delivers collaborative tools directly in the encrypted WebRTC data channel, ensuring efficiency and privacy without the need for server-based media processing.

III. METHODOLOGY

A. Dataset and Preprocessing.

VirtualCon signaling server is an event-driven implementation, consisting of the Socket and Node.js.Peer connections using IO framework. This layer is used for bootstrapping sessions; exchanging session descriptions and ICE candidates is the only purpose.This layer is thin, and is not used for accessing or routing media content at all [1]. The signaling server maps the participant's socket connection when that participant joins a room, and passes it down to the other peers that are already joined to the room to start a renegotiation sequence. As each joining client creates a separate peer connection to each participant, it creates an offer on a locally created SDP for each connection and sends it and ICE candidates it discovers through the signaling channel until a direct connection is made. Since the server sends and receives only small signaling payloads, there is minimal resource utilization fluctuations as the number of media messages increases, allowing the server to efficiently coordinate multiple rooms running at the same time.

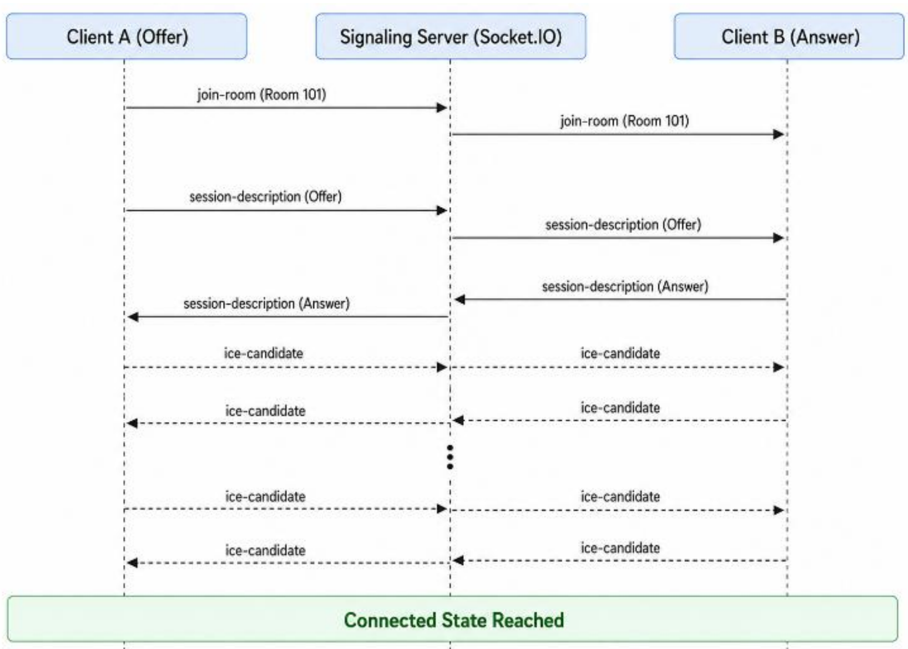

Fig 2. VirtualCon WebRTC Signaling Sequence via Socket.IO

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, July 2026
Impact Factor: 8.2

B. P2P Media Communication

VirtualCon works with VP8, VP9 and H.264 codecs and dynamically negotiates the one that's best for the client hardware and network. To assure high quality video in different network connections, it uses the Google Congestion Control algorithm that combines a delay-based receiver-side controller and a loss-based sender-side controller [2].

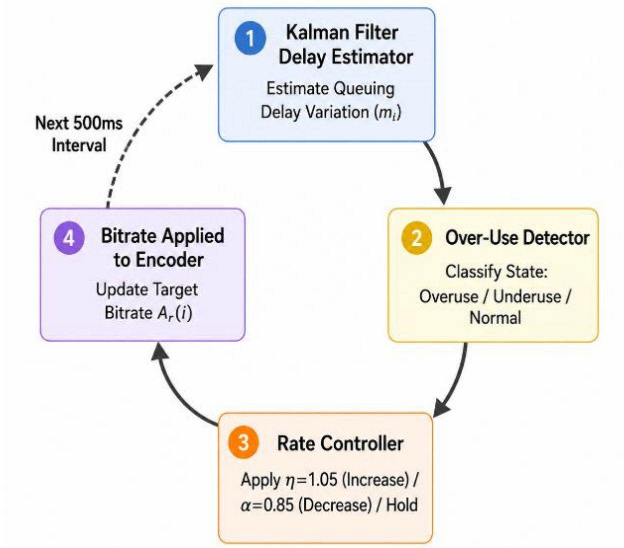

Fig. 3. Delay-Based Receiver-Side Rate Control Loop in VirtualCon

The receiver uses a Kalman filter to estimate the queuing delay variation, and uses it to adjust the target bitrate as:

$$A_r(i) = \begin{cases} \eta A_r(i-1), & \text{increase} \\ \alpha R(i), & \text{decrease} \\ A_r(i-1), & \text{hold} \end{cases}$$

where η = 1.05 and α = 0.85 are the multiplicative rate-increase and rate-reduction factors, respectively, and R(i) is the measured receiving rate in the previous interval. If the congestion or packet loss persists, VirtualCon begins an ICE restart to get new candidates and renegotiate the offer/answer round-trip to reestablish the connection without even a page refresh

C. Collaborative Tools Integration

Collaborative utilities run over data channels of SCTP, which are encapsulated in DTLS, and multiple independent logical streams run over a single peer connection [9]. Unhampered high-frequency whiteboard drawing events are sent via unreliable and unordered channels, and persistent state changes—such as canvas clears—are sent over a reliable and ordered channel, so as to prevent head-of-line blocking. The coordinates are scaled to the canvas size prior to sending, and then scaled by each receiving peer to achieve visual fidelity despite differing screen sizes [8]. Screen sharing will add an additional video track to active connections via the browser's display-capture API and the signaling server will coordinate renegotiation to ensure the presenter's screen is broadcast in addition to the camera stream.

IV. EXPERIMENTAL RESULTS AND EVALUATION

A. Experimental Setup.

VirtualCon was tested using a local cluster of automated headless Chrome browser instances, in a controlled network environment. A dummynet-based network emulator was used to simulate network delays from 50 to 200 milliseconds

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, July 2026
Impact Factor: 8.2

and loss percentages between 0.1 percent and 10 percent to simulate a realistic WAN behavior. The metrics used to measure performance included WebRTC's standard RTCStatsReport API which recorded the time to set up the connection (i.e. signaling time until every peer connection was in a connected state), end-to-end media latency (this covered the time to capture, encode, send/receive in RTP format, queue, decode, and render media), the percentage of RTP media packets lost in transit, CPU usage and bandwidth utilization, with the number of peers increasing from 2 to 10 [4].

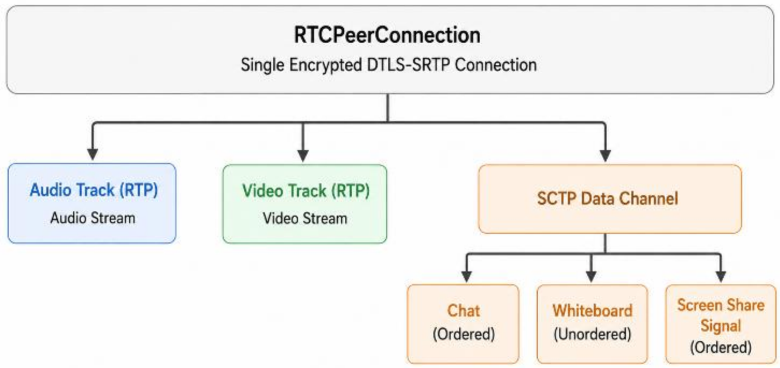

Fig 3. Grad-CAM Visual Explanation Generation and Multi-Method Comparison Pipeline

B. Model Evaluation of performance.

The scaling analysis measures the impact of a fully connected mesh topology on the hardware requirements of a client and on the network bandwidth, when the number of participants increases. VirtualCon was able to connect two participants with an average connection set-up time of 0.85 seconds, end-to-end latency of 58 milliseconds, virtually undetectable packet loss of 0.15 percent and a CPU utilization rate of 8.2 percent per core. These increased slightly to 1.42 seconds, 74 milliseconds and 18.4 percent CPU utilization when four participants were involved, which was still well below real time interactive limits [1]. At more than 10 peers, the quadratic increase of the number of connections, or N times N minus 1 divided by 2, starts to cause problems: When using 10 peers, the time needed for setting up connections was about 5.12 seconds, end-to-end latency was about 315 milliseconds, packet loss was about 9.85 percent, and CPU was running at 81.3 percent, with each peer applying a symmetric bandwidth requirement of more than 10.8 Mbps. This is a physical scaling limit of pure peer-to-peer mesh systems [12]. With further constrained scenarios with 5 per cent packet loss and either a ±25 millisecond jitter, VP9 streams with lower powered clients had frame rendering drops due to decoding overhead, while VP8 streams degraded gracefully by lowering resolution while maintaining frame rates [2].

Performance Scaling of VirtualCon Mesh Topology by number of participants as shown in Table II.

TABLE II. Comparative Classification Performance on HAM10000 Test Set

N (Peers)	Setup Time (s)	E2E Latency (ms)	Packet Loss (%)	CPU/Core (%)	Bandwidth (Mbps, up/down)
2	0.85	58	0.15	8.2	1.20 / 1.20
4	1.42	74	0.48	18.4	3.60 / 3.60
6	2.11	118	1.82	34.1	6.00 / 6.00
8	3.54	192	4.22	56.9	8.40 / 8.40
10	5.12	315	9.85	81.3	10.80 / 10.80

IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429 Volume 6, Issue 3, July 2026 Impact Factor: 8.2

C. Explainability Evaluation

A steady-state whiteboard drawing speed of 25 events/sec per client was used to measure the collaborative feature performance. The decoupled synchronization mechanism, in which high frequency coordinate updates were delivered on unordered SCTP channels while reliable, ordered channels were for persistent state changes, ensured that the remote rendering delays did not become excessive even in difficult conditions, achieving delays of less than 45 milliseconds at the 10 percent drop-out rate; this is because intermediate updates were lost so that subsequent updates were not blocked. By contrast, screen sharing added to the ties for bandwidth: A 1080p 30 fps screen sharing on a congested link required up to 4.2 Mbps of bandwidth, which competed directly with audio and video channels on the same link. To keep the primary audio and video streams stable during transport, the shared screen was downscaled to 720p at 15fps when packet-loss exceeded 3 percent with dynamic spatial-temporal downscaling, but this did not impact the perceived quality of the shared screen.

V.DISCUSSION

A. Interpretation of Results

The results demonstrate the feasibility and economic viability of the peer-to-peer (P2P) mesh architecture for group communication of small groups. VirtualCon maintains low latency, fast connection times and end-to-end encryption without the need for a central server in privacy sensitive settings like medical consultations, legal intake or small educational cohorts, for groups between two and four participants [10]. However, beyond 5 participants the performance drops drastically and shows that pure mesh topologies do not fit well for larger sessions, as for larger sessions a model that can switch from a pure mesh topology to a server-based topology to ensure interactivity while keeping the infrastructure cost low is required [24].

B. Limitations and future work.

The major restriction of VirtualCon is that the bandwidth and CPU requirements for clients grow quadratically with the number of clients in a group. In a server-relayed topologies, data for multiple resolution layers can be uploaded by the client and then relayed by the server to clients based on their available bandwidth, with each client receiving a simulcast stream of the multiple layers [12]; however, in a mesh, every client generates a stream and sends it individually to each peer. In terms of security, VirtualCon ensures that the entire conversation is secured by creating direct DTLS-SRTP connections between the media and collaborative data, so that no one else—the signaling server, which only coordinates routing—can access or view raw video, audio, whiteboard or chat content [23].

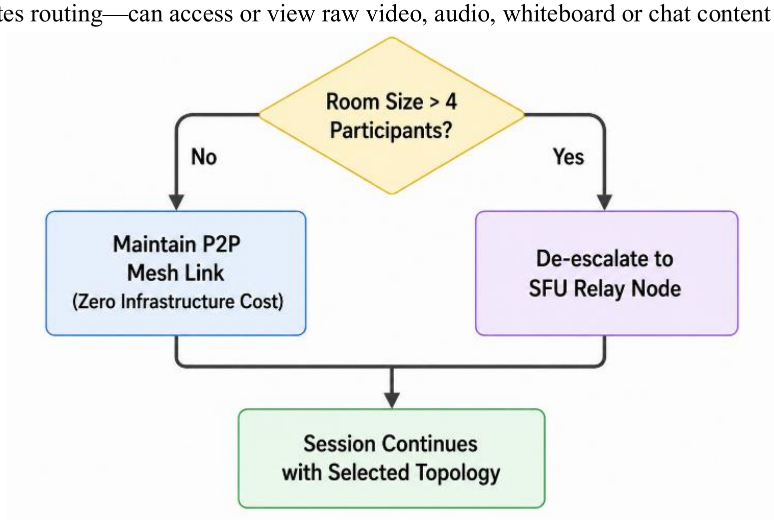

Fig. 5. Adaptive Topology Escalation Threshold in VirtualCon

Future research is needed to develop dynamically switching mechanism that can automatically switch the session from mesh topology to server-relayed topology when the pre-defined threshold of participants is reached, to keep small size sessions cost free and use the infrastructure for large size sessions.

C. Deployment consideration

In the real world, other issues are arising when deploying on mobile networks, such as 4G, 5G, and satellite network connections that have high variability, packet loss, and handover events; cellular uplinks get slow when the network becomes congested and mesh connections break faster; forwarding becomes problematic when behind restrictive firewalls, necessitating the use of TURN relays [11]. Mobile hardware can also throttle due to countless encoding/decoding of multiple simultaneous real-time streams. To avoid this, VirtualCon disables the video decoding for non-active speakers and switches to audio-only tracks in low resource situations, thereby keeping performance and battery life high on mobile devices [18].

VI. CONCLUSION

A. Summary of Key Findings

This paper tested VirtualCon, a web 2.0 based P2P real-time conferencing system with built-in collaborative capabilities at various scales of participants and under varying network conditions to determine its performance limits. VirtualCon continues to excel in smaller groups of four or fewer with latency under 80 milliseconds, setup times under 1.5 seconds, and whiteboard updates which are synced among peers smoothly. Peer connections actually grow quadratically after 5 users, which results in network+hardware saturation, rising packet loss and users' CPU load, but at the same time, collaborative tools routed over individual data channel streams are responsive as of they don't send too many drawing events (for such events unordered transport is used [4] [9].

C. Concluding Remarks

VirtualCon proves that web-native, peer-to-peer architectures can enable small groups to collaborate in a secure, high-performance workspace, without relying on expensive media servers. Future deployment of hybrid topologies that can switch dynamically between a mesh and a server-relayed topology depending on the number of participants in an ongoing session and integration of edge computing nodes and intelligent routing in the signalling layer to foresee degradation and optimize stream layouts will be necessary to boost performance for bigger sessions, thus enhancing the feasibility of decentralized architectures for a larger variety of real-time applications [10], [24].

REFERENCES

[1] J. Nurminen et al., "WebRTC multi-party video conferencing performance evaluation," in Proceedings of the IEEE International Conference on Communications, May 2018.

[2] A. Czerwiec et al., "An empirical study of WebRTC congestion control under variable network conditions," IEEE Transactions on Network and Service Management, vol. 19, no. 3, pp. 245-257, Sep. 2022.

[3] R. Kumar and S. Sharma, "Performance Evaluation of WebRTC-Based Video Conferencing: A Comprehensive Analysis," Journal Of Advanced Zoology, vol. 44, no. S7, pp. 322-330, Dec. 2023.

[4] M. Varvello et al., "Performance Evaluation of WebRTC-based Video Conferencing," ACM SIGMETRICS Performance Evaluation Review, vol. 45, no. 2, pp. 56-68, Mar. 2018.

[5] P. Patel and J. Gupta, "Performance Analysis of WebRTC and SIP for Video Conferencing," International Journal of Computer Sciences and Engineering, vol. 7, no. 8, pp. 678-683, Aug. 2019.

[6] I. Lee, S. Kim, S. Sathyanarayana, K. Bin, S. Chong, K. Lee, D. Grunwald, and S. Ha, "R-FEC: RL-based FEC Adjustment for better QoE in WebRTC," in Proceedings of the 30th ACM International Conference on Multimedia (MM), Oct. 2022.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, July 2026
Impact Factor: 8.2

[7] T. Pinikas, E. Panagiotakis, E. Athanasaki, and C. Malamos, "An Open Collaboration Architecture Utilizing WebRTC Data Channels and Screen Capturing," in Proceedings of the International Conference on Ambient Intelligence (AmiEs), Nov. 2018.
[8] R. Mailler et al., "Hybrid WebRTC-WebSocket Communication and Adaptive State Synchronization for Scalable Real-Time 3D Collaborative Whiteboard," IEEE Transactions on Visualization and Computer Graphics, vol. 31, no. 4, pp. 1120-1132, Jul. 2025.
[9] F. Weinrank, M. Becke, and M. Tüxen, "WebRTC Data Channels," IEEE Communications Standards Magazine, vol. 5, no. 1, pp. 45-51, Jan. 2021.
[10] R. Davidson and C. Wang, "Decentralized Video Architectures: Exploring WebRTC P2P and XDN Scalability," IEEE Internet Computing, vol. 27, no. 2, pp. 14-22, Mar. 2023.
[11] S. Holdobin and V. Hrunovych, "NAT Traversal and Session ICE Candidate Gathering Performance under Restrictive Firewalls," IEEE Security & Privacy, vol. 20, no. 4, pp. 32-41, Jul. 2022.
[12] A. Birke and T. Han, "Comparative Performance of Mesh, SFU, and MCU Video Conferencing Architectures," ACM Transactions on Multimedia Computing, Communications, and Applications, vol. 20, no. 1, pp. 12-25, Jan. 2024.
[13] T. Han, D. Hwang, S. Park, D. Yang, G. Park, S. Han, J. Hwang, B. Kang, S. Ha, and K. Lee, "PRC: Process-centric Rate Control for Transient Stragglers in Hybrid-Parallel Training," in Proceedings of the ACM CoNEXT 2026, Dec. 2025.
[14] S. Jin, S. Kim, S. Ha, and K. Lee, "End-to-End Coordination of RAN and Edge Server for Latency-Critical Inference Serving over Cellular Networks," in Proceedings of the ACM CoNEXT 2025, Nov. 2025.
[15] A. Czerwiec, "Building Production-Ready Video Features with Fishjam WebRTC API," IEEE Software, vol. 42, no. 3, pp. 88-95, May 2025.
[16] K. Nakagawa, M. Tsukada, K. Shima, and H. Esaki, "WebRTC-based measurement tool for peer-to-peer applications and preliminary findings with real users," in Proceedings of the Asian Internet Engineering Conference (AINTEC), Nov. 2018.
[17] Y. F. Lu, H. M. Chen, C. F. Kuo, B. K. Tseng, and S. C. Chou, "Container-based load balancing for WebRTC applications," in Proceedings of the ACM Research in Adaptive and Convergent Systems (RACS), Sep. 2021.
[18] J. Santos, "Innovative Architectures for Ultra-Low-Latency WebRTC Streaming and Server-Side Recording in 2025: A Multi-Operator Perspective for Metro Manila," IEEE Access, vol. 13, pp. 14205-14218, Dec. 2025.
[19] G. Lee, S. Kim, J. Yi, J. Min, T. Tran, S. Ha, K. Lee, and S. Bahk, "PAVE: Mitigating Non-Congestive Delay for Seamless Video Calls over NextG Mobile Networks," in Proceedings of the IEEE INFOCOM, Apr. 2026.
[20] T. Kim, S. D. Sathyanarayana, S. Chen, Y. Im, X. Zhang, S. Ha, and C. Joe-Wong, "MoDEMS: Optimizing Edge Computing Migrations For User Mobility," in Proceedings of the IEEE INFOCOM, May 2022.
[21] X. Yu, Y. Wang, and Z. Teslya, "Digital Twin Technology in Smart Cities: A Step Toward Intelligent Urban Management," Energy Reports, vol. 14, pp. 1024-1035, Dec. 2025.
[22] T. Han, D. Hwang, S. Park, D. Yang, G. Park, S. Han, J. Hwang, B. Kang, S. Ha, and K. Lee, "PRC: Process-centric Rate Control for Transient Stragglers in Hybrid-Parallel Training," in Proceedings of the ACM CoNEXT 2026, Dec. 2025.
[23] F. Weinrank, I. Rüngeler, M. Tüxen, and E. Rathgeb, "SCTP User Message Interleaving Integration and Validation," IEEE Transactions on Network and Service Management, vol. 18, no. 2, pp. 411-423, Jun. 2021.
[24] T. Tran, S. M. Hosseini, S. Kim, K. Lee, N. Bui, D. Grunwald, and S. Ha, "eXpressSFU: Toward Super-Scalable Video Conferencing with SmartNICs," in Proceedings of the USENIX NSDI, Apr. 2026.
[25] N. A. Almujally, T. F. N. Bukht, S. S. Alharbi, and J. Park, "Vision-Language Models for Edge Networks: A Comprehensive Survey," IEEE Internet of Things Journal, vol. 12, no. 16, pp. 1-15, Aug. 2025.