

대역폭 감소를 위한 QUIC-DASH 기반 360VR 영상 전송 시스템 개발

*송민정 *유성근 *박상일

서울과학기술대학교

*mi0_0ng@naver.com, {orcogre, sangilparkmail}@gmail.com

A Development of QUIC-DASH-based 360VR Video Transmission System for Bandwidth Reduction

*Song, Minjeong *Yoo, Sung-geun *Park, Sang-il

Seoul National University of Science and Technology

요약

최근 시청자에게 실제적인 몰입감을 제공하는 실감미디어가 발전하고 있다. 이러한 실감미디어 중 가장 접근성이 뛰어난 것은 VR으로, 현재 다양한 서비스에 응용되고 있는 상황이다. 하지만 360VR은 높은 비트 전송률과 고해상도의 특성을 지니고 있기 때문에 대역폭 비용 및 시청자의 QoE 보장의 불확실성 등의 다양한 문제가 있다. 이러한 문제를 개선시키기 위해 HTTP 기반의 적응적 스트리밍 기법이 발전해나가고 있으며 이 기술의 표준으로 MPEG-DASH가 채택되었다.

MPEG-DASH는 TCP 기반의 전송 프로토콜을 사용하고 있지만 현재 TCP는 고용량의 데이터가 전송되는 웹 기반에서 HTTP의 병목현상을 일으켜 대역폭 효율성을 떨어뜨리는 하나의 원인으로 나타나고 있다. 이러한 문제를 해결하기 위해 본 논문에서는 UDP 기반의 QUIC 프로토콜을 MPEG-DASH에 적용하는 시스템을 고안하고 이를 QUIC-DASH라 칭한다. 고해상도의 360VR 송출 실험으로 QUIC-DASH 시스템과 기존의 MPEG-DASH 시스템을 비교 분석한 결과로 대역폭의 절감이 이루어진 것을 확인하였다.

1. 서론

360 VR(Virtual Reality) 영상은 시청자에게 FoV(Field of View)를 자유롭게 조정할 수 있게끔 하여 실제적인 몰입감을 제공하고 있다. 하지만 360 VR 영상은 높은 비트 전송률과 고해상도의 특성을 지니고 있기 때문에 네트워크 대역폭 비용의 문제가 발생하고 있다. 또한 시청자의 QoE(Quality of Experience) 보장 문제의 해결을 위해 네트워크 사업자들은 HAS(HTTP Adaptive Streaming) 방식을 시스템에 적용시키고 있다[1-2].

MPEG(Motion Picture Experts Group)은 2012년 MPEG-DASH(Dynamic Adaptive Streaming over HTTP)를 개발하였고, 2012년 이는 HAS 기술의 표준이 되었다.(ISO/IEC 23009-1:2012)[3] 표준으로 등록된 후, DASH 기술은 HEVC(High Efficiency Video Coding) 등의 고해상도의 용량이 큰 동영상 전송 방식에 응용되고 있다.

현재 MPEG-DASH는 전송 프로토콜로 TCP(Transmission Control Protocol)를 이용한다. TCP는 현재 HTTP 병목 현상을 일으키는 하나의 원인으로 대두되고 있다. 이는 전송되는 데이터가 고용량이 되고 TCP가 클라이언트와 서버에게 계속적으로 연결을 요청하고 응답을 받기 때문에 나타나는 현상이다. 이러한 점 때문에 대역폭 활용이 비효율적이다. 또한 TCP의 혼잡 제어 메커니즘(Congestion Control Mechanism)으로 인해 새 연결이 발생할 때, 전송하는 속도에

제한이 걸리고 그 결과로 웹 페이지 로드 시간(PLT, Page Load Time) 늘어남다는 단점이 있다[4].

따라서 본 논문에서는 구글(Google)에서 개발한 UDP(User Datagram Protocol) 기반의 QUIC(Quick UDP Internet Connection)을 MPEG-DASH에 적용하여 기존 TCP 전송 프로토콜의 단점을 개선하고 사용자에게 하여금 개선된 대역폭 활용을 제공하여 QoE를 보장할 수 있는 360VR을 시청할 수 있도록 하는 전송 시스템을 개발하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로써 MPEG-DASH와 구글의 QUIC에 대해 탐색한다. 3장에서는 QUIC-DASH 시스템과 이를 어떤 방식으로 360VR 시스템에 적용하였는지 설명한다. 4장에서는 Node.js 기반 웹 서버의 MPEG-DASH를 이용하여 360VR을 송출한 시스템과 QUIC 서버를 이용한 QUIC-DASH 송출 시스템의 전송 속도 및 대역폭 등을 비교하여 분석하였으며 마지막으로 5장에서 결론을 맺는다.

2. 관련 연구

(1) MPEG-DASH

MPEG-DASH는 HAS의 국제 표준 기술로, 시청자가 사용하는 각각의 단말기와 네트워크의 상황에 맞추어 해당 시청자에게 최적화된 콘텐츠를 제공하고 있다. MPEG-DASH는 설정된 서버에 여러 개

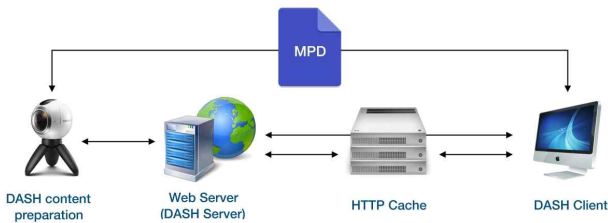


그림 1. MPEG-DASH의 360VR 전송의 개념 구조도

의 품질로 인코딩된 콘텐츠를 저장하고 사용자에게 전달함으로써 끊임 없는 서비스를 경험할 수 있게 한다[1]. 이러한 MPEG-DASH는 현재 Youtube 등의 다양한 OTT(Over The Top) 서비스에 적용되며 실 생활에 밀접하게 존재한다.

그림 1과 같이 MPEG-DASH는 Gear 360, RicohTheta 등의 360 카메라로 촬영된 VR 영상을 MPD(Media Presentation Description) 기반으로 콘텐츠 세그먼트를 선택한 후 검색 및 렌더링을 수행한다. MPD는 전송 할 콘텐츠의 정보를 제공하는 XML(Extensible Markup Language) 형식의 파일이고, 세그먼트는 스트리밍 서비스를 위해 시간과 해상도별로 분할이 된 영상 파일이다. 콘텐츠를 전송 받는 클라이언트는 수신 당시의 네트워크 상황 또는 버퍼 등을 고려하여 비디오의 비트레이트(bit-rate)를 결정한다[5]

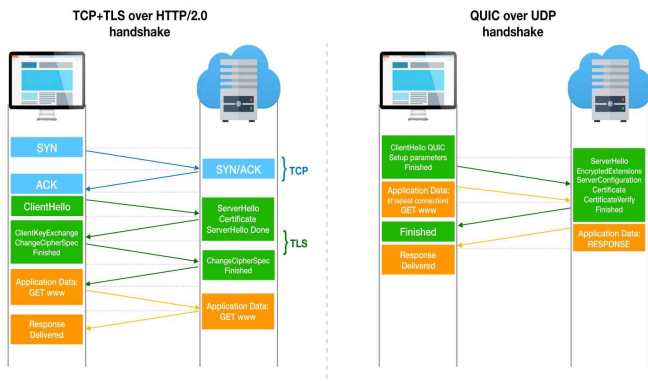


그림 2. TCP+TLS와 QUIC의 핸드셰이크 수 비교

(2) QUIC

QUIC 프로토콜은 HTTP/2.0을 이용함으로써 웹 페이지의 트래픽 성능을 향상시키고 새로운 전송 솔루션으로의 발돋움을 하고 있다. 전통적인 TCP가 아닌 UDP를 이용하도록 개발된 QUIC은 다양한 응용 프로그램의 일부로 배포가 용이해졌다. 기존의 UDP는 보안성에 취약했으나 QUIC은 TLS를 상위 계층으로 설계되어 있기 때문에 패킷이 인증되고 암호화되어 수정을 방지할 수 있다는 장점이 있다[6].

QUIC의 주요 기능 중 하나는 고속 연결 설정이다. 그림 2는 TCP+TLS와 QUIC의 핸드셰이크 수의 차이를 나타낸다. 기존의 TCP+TLS의 경우에는 데이터가 전송될 때 TCP를 통해 전송이 된 후 TLS를 거쳐 암호화된 HTTP 연결을 만들어내기 때문에 최소 2번에서 3번 사이의 패킷 송수신이 이루어진다. 이는 웹 페이지의 대기 시간을 증가시켜 효율성이 떨어진다. 하지만 QUIC의 경우에는 UDP를 전

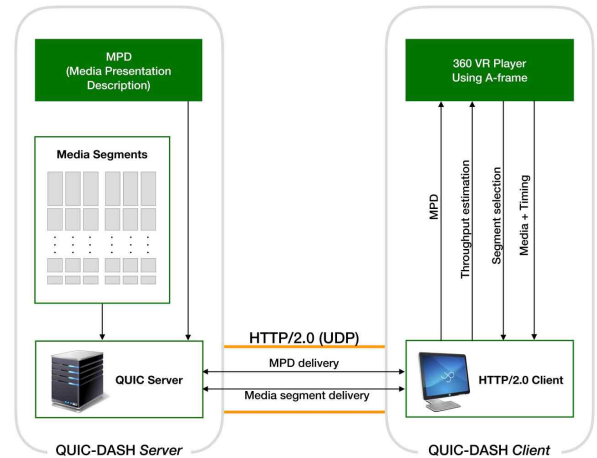


그림 3. QUIC-DASH 시스템 구조도

송 프로토콜로 사용하기 때문에 패킷을 전송받을 때 클라이언트 측에서 응답을 수신하지 않더라도 상황을 분석하여 프로그램 자체에서 해당 프로세스를 처리한다. 이와 같은 구조로 QUIC은 최대 0 왕복시간(RTT, Round Trip Time) 핸드셰이크를 이룰 수 있고 사용자의 웹 페이지 대기 시간을 개선시킬 수 있게 되었다[7].

3. 제안 시스템

(1) QUIC-DASH

현재 MPEG-DASH에 관련한 연구들은 기존의 시스템과 동일하게 HTTP/1.1을 사용하여 진행되고 있다. MPEG에서는 해당 시스템의 웹 페이지 대기 시간 및 지연 발생의 감소를 위해 HTTP/2.0의 서버-푸시(Server-Push) 기능과 웹 소켓(Web-socket)을 활용한 새로운 MPEG-DASH를 개발하고 있는 상태이다[8].

본 논문에서 제안하는 시스템의 구조는 그림 3과 같다. 현재의 MPEG-DASH의 시스템과 유사하지만 기존 MPEG-DASH는 HTTP 서버 및 HTTP/1.1과 TCP 전송프로토콜을 사용하지만 제안 시스템은 MPEG에서 연구하고 있는 것과 마찬가지로 HTTP/2.0 응용 계층(Layer)을 사용하고 HTTP/2.0 클라이언트를 적용시켰다. 또한 QUIC 서버를 시스템에 응용하고 QUIC 프로토콜의 구조로 구성되어있는 UDP 프로토콜을 사용하였다. 따라서, 본 시스템은 QUIC 기반의 MPEG-DASH 기술로써, QUIC-DASH라고 칭하였다.

본 시스템에서는 가장 먼저 기어 360으로 촬영한 비디오를 MPD 파일로 생성하고 전송할 360VR 영상의 세그먼트들을 QUIC 서버에 저장한다. 이 QUIC 서버는 HTTP/2.0 및 UDP를 통해 HTTP/2.0 클라이언트와 통신한다. 이 과정에서 MPD와 세그먼트들이 클라이언트 측으로 전송된다. HTTP/2.0 클라이언트는 웹 페이지의 VR 플레이어에 MPD와 throughput estimation에 대해 전송한다. 또한 웹 페이지는 사용자의 네트워크 상황을 고려하여 VR 영상에 대한 세그먼트를 선택하고 영상의 타이밍에 대해 클라이언트 측으로 정보를 전송한다.

이러한 QUIC-DASH를 이용해 고해상도 및 고용량의 360VR을 전송하면, 전송 대역폭의 효율성을 보완하고 보다 발전된 QoE를 보장하여 사용자에게 한층 더 발전된 경험을 할 수 있게 한다.

4. 실험 결과

(1) 실험 환경

제안하려는 시스템을 구현하기 위하여 먼저 HTTP/1.1을 지원하는 웹 서버와 QUIC을 지원하는 서버를 설치하였다. 물리 서버는 3.4Ghz의 클럭(clock)으로 동작하는 Intel I7-6700 CPU가 장착되어 있고 8GB 메모리를 장착하였다. 물리 서버의 운영체제는 Windows 2016 Datacenter이고, 각각 서버는 VMware Workstation 14버전의 가상머신 위에 설치하였다. 서버 당 2개의 코어와 2GB의 메모리를 할당하였고 가상머신에는 Ubuntu 18.04를 설치하였다. 가상머신은 Bridged Network로 연결되어 게이트웨이로부터 각각 사설 IP주소를 할당받았다. HTTP/1.1서버는 Node.js v.4.8.6을 사용하여 구현되었다. QUIC을 지원하는 서버는 오픈소스로 공개된 Google의 proto-quick 소스를 빌드하여 서버를 구현하였다. 사용된 QUIC 프로토콜의 버전은 VERSION39이다. 실험하려는 웹페이지는 MPEG-DASH 레퍼런스 클라이언트인 Dash.js와 Javascript 프레임워크인 jQuery 3.13을 포함하였다. 또한, 4K 360VR 비디오[9]를 x264를 이용해 키 프레임 간격을 96 프레임으로 설정하고 초당 24프레임으로 인코딩하였다. 그 후 생성된 파일을 MP4Box를 이용해 4000ms 청크(Chunk)로 분리하여 동영상 세그먼트 및 MPD 파일을 생성하였다. 현재 사용자가 360VR을 가장 많이 시청하는 환경인 모바일에서의 콘텐츠 로드를 시뮬레이션하기 위하여 3Mbps로 전송 대역폭을 제한하였다. 이와 같은 웹 페이지를 HTTP/1.1 서버와 QUIC 서버에 동일하게 업로드 하여 총 8개의 MPEG-DASH 세그먼트가 로드될 때까지 문서 오브젝트 모델 콘텐츠 로드(DOM Contents Loaded)에 걸리는 시간과 PLT를 측정하였다.

(2) 실험 결과

그림 4와 같이 HTTP/1.1로 구현된 서버는 문서 오브젝트 모델 콘텐츠 로드에서 총 1.86초가 소요되었다. 문서에 추가적인 엘리먼트(Element)가 없기 때문에 문서 로드에도 같은 시간이 걸렸다. 8개의 영상 세그먼트를 로드하는 데에는 총 38.89초가 소요되었고 총 23개의 리퀘스트를 통하여 총 14.1MB의 데이터가 전송되었다. QUIC을 이용한 서버는 그림 5와 같이 문서 오브젝트 모델 콘텐츠 로드에서 총 1.85초가 소요되었고 문서 로드에도 같은 시간이 걸렸다. 또한, 8개의 영상 세그먼트를 로드하는 데에는 총 38.79초가 소요되었고 총 23개의 리퀘스트를 통하여 총 14MB의 데이터가 전송되었다.

다음과 같은 결과를 통해 QUIC을 사용한 서버가 약 0.1초 가량 전송이 빠른 것을 볼 수 있었다. 이러한 실험 결과에서 주목할 만한 점은 총 리퀘스트의 수가 23개로 많지 않음에도 불구하고, 800Kb의 데이터가 절약 되었다는 것이다. 이와 같은 결과가 도출된 것은 QUIC이 HPACK을 이용한 자체 헤더 압축과 UDP기반의 전송을 사용하기 때문으로, 같은 데이터를 전송하더라도 대역폭을 감소할 수 있는 것으로 확인되었다.

그림 6과 같이 총 20번의 반복을 통하여 문서와 영상 세그먼트 8개가 로드되는 총 시간을 측정하였다. HTTP/1.1의 경우 38초 후반대의 시간을 기록하였으며, QUIC의 경우는 36.5~37초 초반의 시간을 기록하였다. 평균적으로 QUIC이 1초 이상 빠른 것을 확인 할 수 있었으며 이와 같은 결과는 영상의 길이가 더 길거나 특히 세그먼트의 길이가 긴 영상에서 더 좋은 결과를 보여줄 것으로 예상된다. 하나의 세그

먼트가 로드되고 재생되고 나야 그 다음 세그먼트를 전송하는 MPEG-DASH의 특성으로 인하여 QUIC의 효율이 다소 적게 나타났으나, 앞에 재생할 콘텐츠를 미리 로드하는 새로운 방식이 적용된다면 더욱 더 효율적인 HAS기법이 QUIC을 기반으로 구현될 것으로 전망한다.

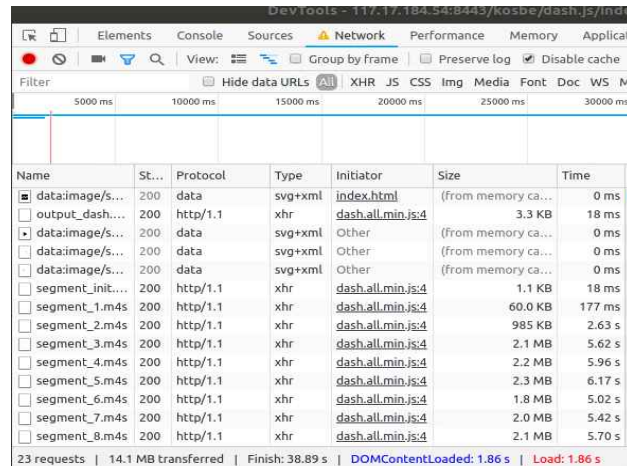


그림 4. MPEG-DASH 기반의 HTTP/1.1 문서 로드 결과

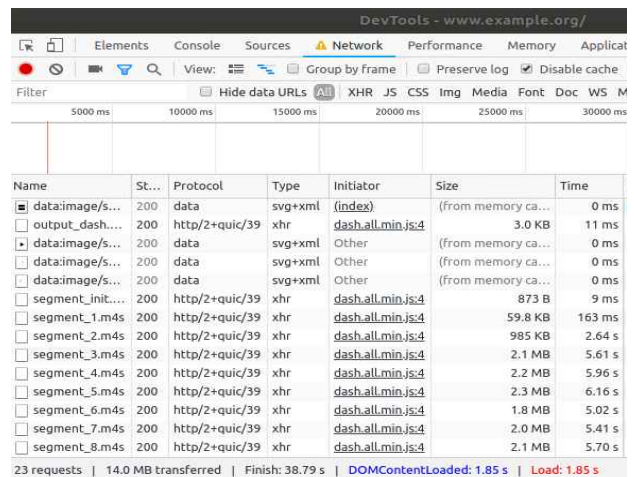


그림 5. QUIC-DASH 기반의 HTTP/2.0 문서 로드 결과

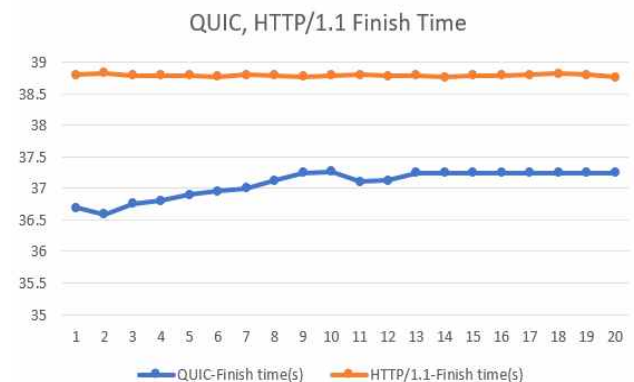


그림 6. 두 시스템의 Finish time 비교

4. 결론

현재 실감미디어의 발전에 따라 다양한 스트리밍 기술이 개발되고 있다. 이 기술의 중심을 이루는 HAS는 다양한 OTT 서비스 및 방송 시스템에 적용되고 있으며 기술 표준으로 MPEG-DASH가 채택되었다. 또한, 구글에서는 HTTP/2.0의 서버 푸시 기능 및 UDP 기반의 QUIC 프로토콜을 개발함으로써 서버와 클라이언트의 핸드셰이크 및 네트워크의 전송 속도를 개선할 수 있게 되었다.

본 논문에서는 QUIC 프로토콜을 MPEG-DASH에 적용시킨 새로운 QUIC-DASH 시스템을 제안하였다. 이 시스템은 기존 MPEG-DASH와 달리 HTTP/1.1 서버 및 클라이언트를 사용하지 않고 QUIC 서버와 HTTP/2.0 클라이언트를 통해 360VR 영상의 MPD 및 세그먼트들을 전송할 수 있다.

기존 HTTP/1.1 기반의 MPEG-DASH와 본 논문에서 제안한 HTTP/2.0 기반의 QUIC-DASH를 비교 분석한 결과, 23 request를 수신 받았을 때, 전송 속도의 큰 차이는 존재하지 않지만 800Kb의 데이터가 절약되는 것을 확인하였다. 이러한 결과를 통해 같은 데이터를 전송하더라도 대역폭을 감소할 수 있는 것으로 확인할 수 있었고, 세그먼트의 길이가 긴 360VR을 전송하는 경우 더 좋은 결과를 보여줄 것으로 예상할 수 있었다.

QUIC-DASH 시스템은 현재 하나의 세그먼트의 전송이 완료된 후 다음 세그먼트를 전송하는 MPEG-DASH의 특성으로 인하여 QUIC 프로토콜의 효율이 두드러지지 않았지만, 앞에 재생할 콘텐츠를 예측하는 새로운 적응적 스트리밍 방식이 개발이 된다면, QUIC 프로토콜을 통해 더욱 뛰어난 성능을 보일 수 있을 것이라 전망된다.

감사의 글

2018년 학연연계 사업화 선도모델 지원사업(홍릉 Tech-Biz OPERA 센터 구축을 통한 학연 연계 기반의 산학연 협력체계 구현 과제)의 지원을 받아 수행된 연구임.

참고문헌

- [1] Sun, Liyang, et al. "Multi-path multi-tier 360-degree video streaming in 5G networks." Proceedings of the 9th ACM Multimedia Systems Conference. ACM, 2018.
- [2] 김상욱, 윤두열, 정광수. (2016). DASH 기반의 다시점 비디오 서비스에서 시점전환 지연 최소화를 위한 비디오 전송 기법. 정보과학회논문지, 43(5), 606-612.
- [3] 박기준, 이길복, 이장원, 김규현. (2012). MPEG DASH 기반 service-compatible 3D 콘텐츠 대상 HTTP adaptive streaming 적용 방안. 방송공학회논문지, 17(2), 207-222.
- [4] Megyesi, Péter, Zsolt Krämer, and Sándor Molnár. "How quick is QUIC?." Communications (ICC), 2016 IEEE International Conference on. IEEE, 2016.
- [5] 신정훈, 이윤주, 박우철. (2014). MPEG-DASH를 위한 HTTP 적응적 스트리밍 알고리즘. 대한전자공학회 학술대회, , 550-552.
- [6] QUIC, a multiplexed stream transport over UDP: "https://www.chromium.org/quic"

- [7] Langley, Adam, et al. "The QUIC transport protocol: Design and Internet-scale deployment." Proceedings of the Conference of the ACM Special Interest Group on Data Communication. ACM, 2017.
- [8] MPEG-DASH with Server Push and WebSockets : "https://mpeg.chiariglione.org/standards/mpeg-dash/dash-server-push-and-websockets"
- [9] Surrounded by Wild Elephants in 4k 360 : "https://youtu.be/mlOIXMvMaZo"