3차원 홀로포테이션 서비스를 위한 중계서버 시스템 구축

김명균*, 김유라, 이승호, 김성제**, 김제우
전자부품연구원 지능형영상처리센터
e-mail : foulpg99@keti.re.kr*, sungjei.kim@keti.re.kr**

Relay Server System Construction for 3D Holoportation Service

Myeong-kyun Kim*, Yura Kim, Seungho Lee,
Sungjei Kim**, Je woo Kim
Korea Electronics Technology Institute (KETI)
Intelligent Image Processing Research Center

요 약

본 논문은 3차원으로 복원된 데이터를 원격지에 있는 홀로렌즈에 실시간으로 전달하기 위해 개발된 중계서버 시스템을 제안한다. 원격지에 있는 홀로렌즈를 위한 실시간 전송 시스템 구축을 위해서는 복원된 3차원 데이터의 압축 스트림을 복호화 해주는 중계서버 시스템이 필요하다. 이와 같은 3차원 데이터의 구조와 원격지에서 데이터를 전달 받는 홀로렌즈 환경을 고려하여 중계서버 시스템을 설계하고 구축 했다.

1. 서론

최근 기어VR, 오큘러스 퀘스트, HTC 바이브, 홀로렌즈, 매직리프 등 VR(Virtual Reality), AR(Augmented Reality) 그리고 MR(Mixed Reality)까지 분야를 가리지 않고 다양한 HMD(Head-mounted display) 제품이 출시되며 3차원 데이터가 주목받고 있다.

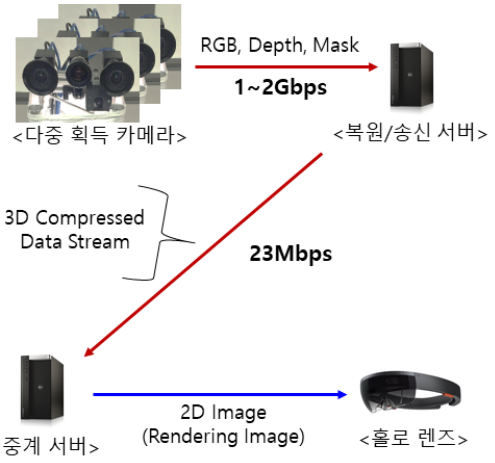
3차원 데이터를 생성하기 위해서는 각 정점의 기하 정보(Geometry), 연결 정보(Connectivity), 그리고 색상 정보(Color)와 법선 벡터(Normal Vector) 등 많은 정보가 필요하다. 3차원 데이터를 홀로렌즈[1]와 같은 HMD로 전달하는 시나리오는 두 개의 원격 캡쳐 장비가 직접 연결을 하여 가상공간과 물리적 공간이 일대일로 대응되는 일대일(1:1) 시나리오와 단일 캡쳐 장비가 실시간으로 여러 수신자에게 브로드 캐스트 해주는 일대다(1:N) 응용 프로그램으로 구성된다. 이 두 가지 시나리오와 같은 시스템을 홀로포테이션(Holoportation)[1] 시스템이라 하는데, 제한된 대역폭 내에서 대용량의 데이터를 전송해야 되기 때문에 일반적으로 지연 시간이 적은 일대일 홀로포테이션 시스템을 이용하게 된다.

일대다 방송 시나리오는 대용량의 이미지가 수많은 수신자에게 전달되어야 되기 때문에 복원/송신 서버에서 3차원 데이터를 부호화된 스트림으로 전송하게 된다. 이 데이터를 홀로렌즈에서 전달받아 렌더링하기 위해서는 복원/송신 서버와 홀로렌즈 사이에 데이터를 복호화해주는 중계서버 시스템이 필요하다.

본 논문에서는 실시간으로 복원되는 스포츠 경기, 음악 공연과 같은 대규모 일대다 방송 시나리오에 적합한 홀로포테이션 시스템에 사용되는 중계서버 시스템을 제안한다. 이 중계서버는 실시간으로 획득하여 부호화 된 3차원 데이터 정보를 전달받아 복호화하여 홀로렌즈에서 중계서버로 전송 요청이 들어오면 요청된 데이터 타입의 3차원 데이터를 전송하게 된다.

본 논문의 구성은 다음과 같다. 2절에서 홀로포테이션을 이용한 중계서버의 구조를 제안하고, 3절에서는 2절에서 제시한 구조로 구축된 중계서버 시스템과 홀로렌즈의 시연 결과를 보여준다. 그리고 4절에서 본 논문에 대한 결론을 맺는다.

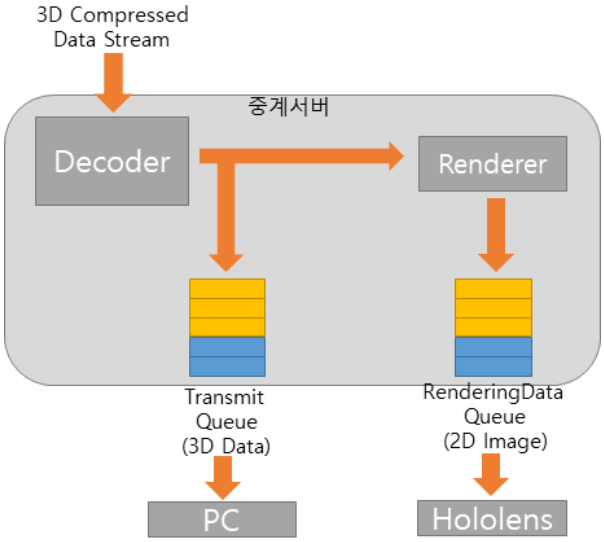
2. 중계서버 구조



(그림 1) 중계서버를 이용한 홀로포테이션 구조

본 논문에서 중계서버를 활용하여 구축한 홀로포테이션 시스템의 구조는 그림 1과 같이 구성된다.

다중 POD로 이루어진 획득 카메라에서 물체가 존재하는 공간의 RGB, Depth, Mask 2차원 정보를 획득하게 된다. 획득된 2차원 정보는 10G 네트워크망[3]을 통해 복원/송신 서버에 전달된다. 전달된 2차원 공간 획득 정보는 복원/송신 서버에서 3차원 데이터로 복원 후 부호화 과정을 거쳐 3차원 데이터(Mesh, Point Cloud)와 선택적으로 2차원 데이터(Texture map)스트림을 생성한다. 복원/송신 서버에서 생성된 데이터 스트림은 무선 네트워크(Wi-Fi)를 통해 중계서버에 전달된다. 전달된 데이터 스트림은 중계서버에서 복호화 과정을 거쳐 중계서버 렌더러에 렌더링을 실시간으로 진행하고, 홀로렌즈에서 중계서버로 데이터 전송 요청 시 홀로렌즈에서 요청한 데이터를 선택적으로 전달한다.



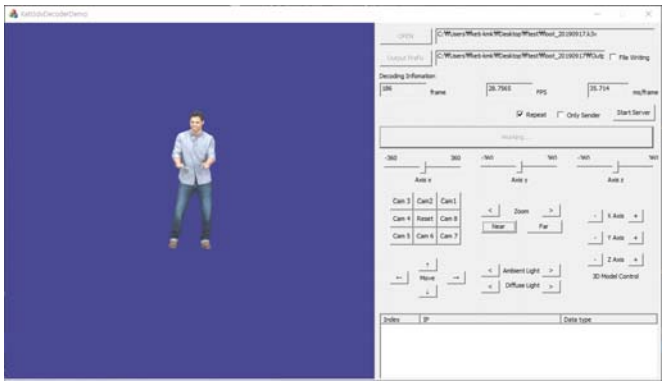
(그림 2) 중계서버 구조도

중계서버의 구조도는 그림 2와 같다. 데이터 스트림의 효율적인 전송을 위해 복원/송신 서버에서 NAL 유닛[4] 형태로 네트워크를 통해 중계서버로 전달한다. 중계서버는 전달받은 데이터를 디코더를 통해 복호화 한다. 복호화 된 데이터는 현재 복호화 되는 데이터의 상태 확인을 위해 중계서버 렌더러에 3차원 모델에 대한 정보(Vertex, Color, Face, Texture)를 전달하고, 홀로렌즈로 전달되는 데이터(Mesh, Point cloud)는 디바이스가 중계서버에 데이터 전송요청이 들어올 때 요청된 데이터를 즉시 보내주기 위해 각각의 큐에 쌓아둔다.

네트워크 환경 또는 홀로렌즈의 상대적으로 낮은 하드웨어 성능 때문에 3차원 데이터가 홀로렌즈에서 원활하게 렌더링 되지 않는 상황이 발생 할 수 있다. 그래서 중계서버는 위 상황을 고려하여 중계서버 렌더러에 출력된 2차원 화면을 2차원 이미지의 형태로 홀로렌즈에 전달하는 기능을 지원한다. 2차원 이미지를 홀로렌즈로 전달하는 방법은 3차원 데이터를 전달하는 방법보다 데이터의 양이 적어지기 때문에 상대적으로 안정적인 홀로렌즈의 렌더링 성능을 확인 할 수 있다.

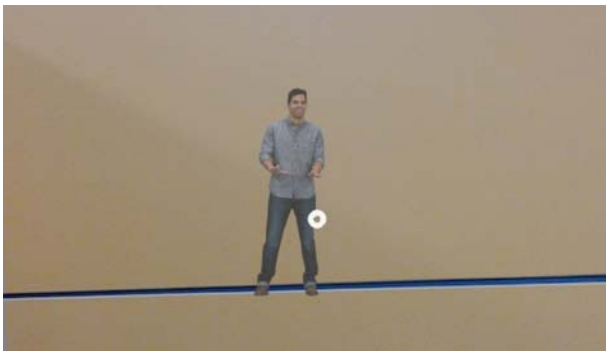
중계서버에 데이터 전송 요청 가능한 홀로렌즈 수는 대역폭에 따라 결정되기 때문에 가변적이다.

3. 결과



(그림 3) 중계서버 프로그램 UI

그림 3은 위에서 언급한 중계서버 시스템을 사용하기 위한 UI다. 좌측에 중계서버 렌더러가 위치하고, 오른쪽에 컨트롤러, 현재 처리되는 데이터의 상태, 그리고 접속된 홀로렌즈의 정보를 나타낸다.



(그림 4) 홀로렌즈 출력 화면

그림 4와 같이 중계서버에서 실시간으로 부호화된 데이터가 홀로렌즈로 전달되고, 재생되는 것을 확인 할 수 있다. 홀로렌즈가 중계서버에서 3차원 데이터를 전달 받을 때 무선 네트워크 환경에서 대용량의 데이터를 실시간으로 전달받기 때문에 홀로렌즈의 부하가 발생할 수 있다.

4. 결론

구축된 중계서버 시스템을 통해 복원된 3차원 데이터를 실시간으로 홀로렌즈를 통해 출력 할 수 있었다. 하지만 상대적으로 낮은 디바이스의 성능과 대용량의 데이터 전달로 인해 디바이스 출력 화면의 부하를 확인할 수 있었다.

앞으로 AR 디바이스의 성능 발전과 좋은 품질의 3차원 데이터를 적은 용량으로 디바이스에 전달하는 방법을

개발한다면 부하를 최소화하고, 높은 품질의 3차원 데이터를 전달하는 중계서버 시스템이 완성 될 것을 전망한다.

Acknowledgement

이 논문은 2019년도 정부(과학기술정보통신부)의 재원으로 '범부처 Giga KOREA 사업'의 지원을 받아 수행된 연구임 (No.GK19P0200, (4D실감-총괄/1세부) 4D 복원 및 동적 변형 거동 모델 기반의 초실감 서비스 기술 개발)

참고문헌

[1] https://docs.microsoft.com/ko-kr/windows/mixed-reality/hololens-hardware-details

[2] Sergio Orts-Escolano, Christoph Rhemann, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip L. Davidson, Sameh Khamis, Mingsong Dou, Vladimir Tankovich, Charles Loop, Qin Cai, Philip A. Chou, Sarah Mennicken, Julien Valentin, Vivek Pradeep, Shenlong Wang, Sing Bing Kang, Pushmeet Kohli, Yuliya Lutchyn, Cem Keskin, Shahram Izadi, "Holoportation: Virtual 3D Teleportation in Real-time", User Interface Software and Technology(UIST), October 2016, Tokyo, Japan.

[3] https://standards.ieee.org/standard/802_3ae-2002.html

[4] Thomas Wiegand, Gary J. Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the H.264 / AVC Video Coding Standard", July 2003.