

사물인터넷 환경을 위한 스트리밍 프로토콜의 성능 분석과 스마트 낚시 루어의 구현

황제민¹⁾, 신예진²⁾, 설순욱³⁾*

Performance Analysis of Streaming Protocols for IoT Environment and an Implementation of Smart Fishing Lure

Jemin Hwang¹⁾, Yejin Shin²⁾, Soonuk Seol³⁾*

요 약

본 논문에서는 전원효율성과 실시간 서비스가 요구되는 사물인터넷 환경에 적합한 스트리밍 프로토콜의 성능을 비교하기 위한 실험을 진행하고, 그 결과에 따른 스트리밍 프로토콜을 스마트 낚시 루어에 적용하여 결과를 확인한다. 최근 사물인터넷 응용 기술이 발달하며, 이에 따라 영상기반 서비스의 활용도 높아지는 추세이다. 사물인터넷 장치들은 소형화로 인해 성능이 기존 장치들에 비해 경량화되어 실시간 스트리밍 기능에 여러 가지 제약을 가지게 된다. 사물인터넷 기기와 같이 전원이나 네트워크 대역이 제한되는 경우 한정적인 자원의 효율적인 활용이 필요하다. 하지만 스트리밍 프로토콜의 성능을 비교한 기존 연구들은 이더넷이나 WiMAX 환경에서 QoE를 비교하거나 패킷을 분석하여 지연 성능을 비교하는 실험을 진행하였다. 우리는 실험을 통해 무선네트워크 성능이 낮거나 전원이 제한적인 환경에서 요구되는 성능 지표와 실시간 스트리밍에서 요구되는 품질 지표를 이용해 프로토콜의 성능을 비교하여 가장 적절한 스트리밍 프로토콜을 찾는다. 이를 통해 효율적인 전력 사용이 요구되고 네트워크 환경이 제한적인 어플리케이션에서 기존 제품에 비해 향상된 성능과 효율적인 라이브 스트리밍 서비스가 가능함을 보인다.

핵심어: 사물인터넷, 실시간 스트리밍, 무선통신, 저전력, 자원제한

Abstract

Received(July 5, 2019), Review Result(July 28, 2019)

Accepted(September 5, 2019), Published(September 30, 2019)

¹Student MA Course, 31253, Department of Electrical, Electronics & Communication Engineering, KOREATECH, Byeongcheon-myeon, Dongnam-gu, Cheonan, Korea

E-mail: zpdltusl@koreatech.ac.kr

²Student Ph.D. Course, 31253, Department of Electrical, Electronics & Communication Engineering, KOREATECH, Byeongcheon-myeon, Dongnam-gu, Cheonan, Korea

E-mail: yepp1252@koreatech.ac.kr

³(Corresponding Author) Associate Professor, 31253, School of Electrical, Electronics & Communication Engineering, KOREATECH, Byeongcheon-myeon, Dongnam-gu, Cheonan, Korea

E-mail: suseol@koreatech.ac.kr

* 본 연구는 2017년도 중소벤처기업부의 기술개발사업 [C0564746], 2019학년도 한국기술교육대학교 교수교육연구진흥과제 연구비 지원에 의하여 연구되었음

In this paper, we conduct experiments to compare performance of streaming protocols suitable for IoT environment where the power efficiency and real-time service is required, and examine the results by applying the streaming protocol to a smart fishing lure. Recently, IoT technology has been developed, and the use of video-based services is increasing. As IoT devices become smaller, their performance is lighter and there are various limitations on the real-time streaming service. When power or network bandwidth is limited, such as IoT devices, efficient use of limited resources is required. However, Previous studies that compare the performance of streaming protocols have focused on comparing QoE (Quality of Experience) in Ethernet or WiMAX environment or delay by analyzing packets. we compare the performance of the streaming protocols in terms of the performance indicators required in poor wireless network, limited-power, and real-time streaming environment to find the most appropriate streaming protocol. We confirm that improved performance and efficient live streaming services are possible compared to the existing products in applications where efficient power usage is required and network environment is limited.

Keywords: IoT, real-time streaming, wireless communication, low-power, resource-constrained

1. 서론

사물인터넷 (Internet of Things, IoT) 기술이 발전함에 따라 인터넷에 연결되는 기기를 쉽게 접할 수 있는 시대가 다가오고 있다. 사물인터넷의 여러 응용들 중에는 가정 내의 전자제품들을 제어하거나 현재 상태를 확인할 수 있는 스마트 홈 시스템 등이 있으며, 최근에는 웹캠이나 로봇청소기에 부착된 카메라 등을 통해 주변을 촬영한 영상을 사용자에게 제공하는 영상기반 실시간 모니터링 기능의 활용도 높아지는 추세이다. 하지만 이동성 및 소형화가 중요한 요소인 사물인터넷 장치들은 컴퓨팅 성능이나 통신 환경, 배터리 등 자원이 제한됨에 따라 데이터량과 연산량이 많은 실시간 스트리밍 기술을 적용하기에는 많은 제약이 따른다. 또한 실시간 스트리밍 기능을 제공하는 기존 제품[1-2]의 경우, 스트리밍 가능한 유효거리가 20~30미터 이내로 한정되어 있어 실질적인 활용에 어려움이 있다. 사물인터넷에서의 전력 효율을 개선하기 위해 LoRa (Long Range)와 BLE (Bluetooth Low Energy)같은 기술이 등장했지만, 낮은 네트워크 속도로 인해 실시간 스트리밍 서비스를 제공하기는 부족한 성능을 가진다. 이와 같이 사물인터넷 환경에서 스트리밍 기능을 위한 여러 시도가 있었지만, 자원제한적인 요인으로 인해 한계가 존재한다. 실시간 스트리밍 기술의 성능에 대한 기존연구[3-5]가 진행되었지만 QoE (Quality of Experience)를 평가하거나[3-4], 영상의 실시간성을 평가[5]하는 등 네트워크 속도가 보장되는 환경에서 어떤 기술이 가장 좋은 성능을 나타내는지에 대해서만 연구되어왔다.

본 논문에서는 제한적인 네트워크 환경에서 배터리로 동작하는 사물인터넷 기기에 적용하기 위한 최적의 스트리밍 기술을 찾는다. 영상을 스트리밍할 때 현존하는 스트리밍 프로토콜들이 사용하는 자원의 효율성을 비교하기 위한 지표를 구성하고, 실험을 통해 사물인터넷 장치에서의 스트리밍 프로토콜의 성능을 분석한다. 또한 기존[6]에 구현한 사물인터넷 기기에 실험 결과를 적용하여, 자원이 제한된 환경에서 향상된 자원효율성을 확인한다. 본 논문은 선행연구[7]를 통해 찾은 결과를 사물인터넷 환경에 실제로 적용을 하여 기존 제품 대비 향상된 성능을 확인한다. 사물인터넷

넷 기술은 환경에 따라 장치의 성능이나 활용가능한 자원에서 차이가 발생한다. 따라서 실험을 통해 스트리밍 기술을 적용하는 사물인터넷 환경마다 적합한 프로토콜을 선정할 수 있도록 한다.

2. 관련 연구

2.1 실시간 스트리밍 프로토콜

실시간 스트리밍 기술은 대표적으로 WebRTC (Web Real-Time Communication)[8], MPEG-DASH (Moving Picture Experts Group-Dynamic Adaptive Streaming over HTTP)[9], RTSP (Real Time Streaming Protocol)[10]가 있다. 이 프로토콜들은 연결수립 방식이나 스트리밍 방식에서 차이점을 지니고 있어, 각각의 장단점에 대해 분석하는 연구가 활발히 진행되고 있다[3-5].

WebRTC는 웹 브라우저 및 모바일 애플리케이션 간에 플러그인의 도움 없이 서로 통신할 수 있도록 설계된 API이다. 즉, 별도의 프로그램을 설치하지 않아도 웹 브라우저만으로도 실시간 멀티미디어 서비스를 제공하는 것을 목적으로 한다. WebRTC는 오픈소스이기 때문에 직접 빌드하여 사용할 수 있으며, 웹 브라우저 뿐 아니라 모바일 애플리케이션에서도 적용이 가능하다는 특징을 가지고 있다.

MPEG-DASH는 ISO 표준 (ISO/IEC 23009-1)으로 개발된 HTTP Adaptive Streaming의 한 종류이다. Adaptive Streaming은 서버에 다양한 화질 및 음질을 가진 짧은 길이의 영상을 스트리밍하며, 플레이어는 짧은 영상들을 연결시켜 연속적으로 재생하는 방식으로 동작한다. 또한 수신측의 네트워크 환경에 따라 해상도 등을 조절하여 적절한 품질의 영상을 전송하는 특징이 있다. I. Sodagar의 연구[11]에 의하면 MPEG-DASH는 영상을 작은 조각단위로 나누어 저장한 파일을 스트리밍하는 방식으로, 영상을 저장하지 않고 스트리밍하는 방식과 차이점을 가진다.

RTSP는 IETF가 1998년에 개발된 통신 규약으로, RFC 2326[10]에 정의되어 있으며 많은 실시간 스트리밍 응용에 적용되는 기술이다. 영상 데이터를 전송하기 위해 RTP (Real-time Transport Protocol)[12]를 사용하고, RTSP는 영상의 상태를 제어하기 위해 사용된다.

2.2 기존의 성능분석 관련연구

실시간 스트리밍 프로토콜 간 성능을 비교하거나 성능을 분석하기 위해 많은 연구들이 진행되었다. WebRTC와 MPEG-DASH의 성능을 분석하기 위해 영상의 품질에 직접적인 영향을 미치는 모바일 네트워크의 특성에 대해 분석을 하거나[3], MPEG-DASH와 RTSP, RTMP (Real Time Messaging Protocol)를 이용한 실시간 스트리밍 영상의 QoE (Quality of Experience)에 대한 평가 [4]를 진행하였다. WebRTC와 RTSP의 연결수립시간 및 영상지연 성능에 대한 실험을 진행한 연구

[5]도 있다. 위와 같은 기존 연구들은 네트워크 환경이 보장되는 이더넷 환경에서 실험되었으며 실험을 위한 장치는 PC 또는 스마트폰을 사용하였다.

WebRTC와 MPEG-DASH의 성능을 비교한 연구[3]에서는 WiMAX환경에서 차량과 같은 이동수단을 통해 일정한 속도로 이동을 할 때, 다양한 무선 환경에서 비디오 서비스의 품질에 직접적인 영향을 미치는 모바일 네트워크의 특성을 확인하기 위한 실험을 진행하였다. 실험은 이동 중에 무선 네트워크를 통한 비디오 서비스를 송수신하여 진행되었으며, 송수신 데이터를 성능지표를 기준으로 비교를 하였다. 또한 이 실험은 두 프로토콜의 성능 지표를 각각의 설계목적에 맞추어 다르게 도출하였다. MPEG-DASH의 경우, Buffer Status, Bitrate, Segment Download 횟수 등이 있다. WebRTC의 경우, Bitrate, Frame Rate, Frame Size 등이 있다. 영상 지연은 패킷의 RTT (Round Trip Time)를 이용하여 측정을 하였다.

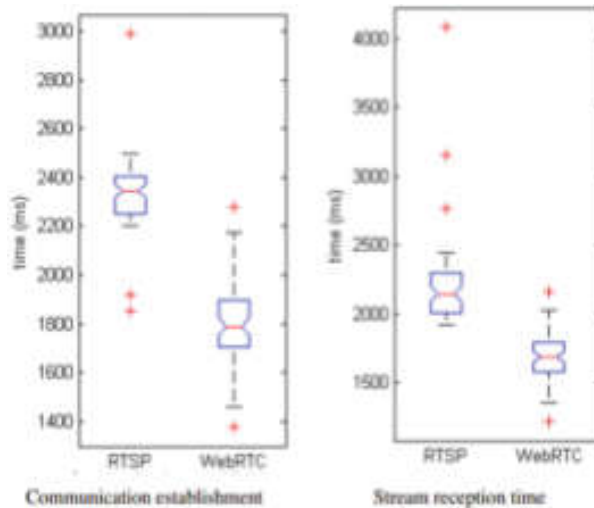
MPEG-DASH와 RTSP, 그리고 RTMP의 성능을 비교한 연구[4]에서는 4G 및 Wi-Fi환경에서 실시간 영상 스트리밍의 성능을 MOS (Mean Opinion Score)를 이용해 측정한 QoE로 평가하였다. 실험에 사용한 지표는 [표 1]과 같다. 실험은 측정 지표를 기준으로 각 조건 별로 진행하였으며, 모바일 네트워크상에서 스트리밍 되는 비디오의 성능을 측정하여 진행하였다. PLR (Packet Loss Rate)와 지연에 따른 네트워크 사용량의 변화에 따른 성능 차이를 측정하였고, 영상 스트리밍이 QoE측면에서 최고의 성능을 나타낼 수 있는 변조방식과 네트워크 환경에 대해 분석하였고, 각 스트리밍 프로토콜별 스트리밍 성능의 차이를 비교하였다.

[표 1] MPEG-DASH, RTSP, RTMP 성능 측정 실험 환경[4]

[Table 1] Cases for experiments of protocols[4]

Tested Parameters	Values
Resolution (px)	(320 x 240); (1280 x 720)
Coding bit-rate (Kbps)	280 and 700 for (320 x 240) 1000 and 2500 for (1280 x 720)
Framerate (FPS)	15; 30
Key Frame Interval (s)	2; 10
PLR (%)	{0; 0.5; 1; 2; 3}
Delay (ms)	{0; 50; 100; 150; 200}

RTSP와 WebRTC의 성능을 비교한 연구[5]에서는 실시간 영상 스트리밍에서 RTSP와 WebRTC의 영상 지연, 스트리밍을 위한 연결수립 시간을 측정하여 성능을 비교하였다. 모바일 클라이언트와 스트리밍 서버인 PC로 환경을 구축하여 동일한 영상에 대해 실시간 스트리밍 프로토콜의 성능을 실험하였으며 실험의 결과는 [그림 1]과 같이 나타났다. 하지만 이 실험의 경우 여러 가지 해상도에 따른 성능비교는 실험되지 않았다.



[그림 1] RTSP와 WebRTC의 성능 측정 결과[5]

[Figure 1] Result of performance measurement[5]

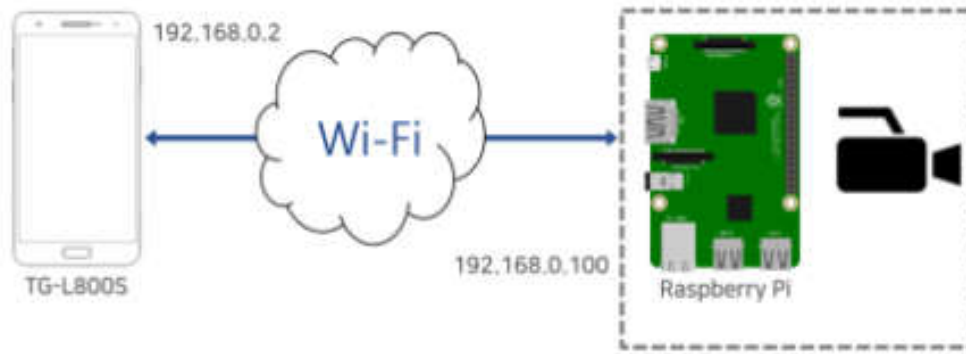
본 논문에서 제안하는 시스템의 환경은 제한된 배터리를 사용하고 불안정한 네트워크 대역을 가진다. 반면, 기존의 스트리밍 성능분석 연구에서는 WiMAX환경이나 LTE, Wi-Fi환경에서 스트리밍 프로토콜의 성능에 대한 비교나 분석이 이루어졌다. 네트워크가 보장되는 상황에서 실시간 스트리밍의 QoE를 비교하거나, 특정 해상도의 영상을 통해 영상의 지연을 비교하였다. 또한 일반적인 모바일 클라이언트 환경에서 측정을 하였기 때문에 스트리밍 서버의 전류소모에 대한 측정은 진행되지 않았다.

하지만 제안하는 시스템의 경우 동작하는 환경의 특성 상 스트리밍 서버의 자원이 제한되어있기 때문에 전류소모량과 컴퓨팅 자원 사용량이 성능 지표에 추가적으로 반영되어 비교가 이루어져야한다. 따라서 사물인터넷 환경을 기준으로 성능을 비교하기 위해 사물인터넷 환경에서 요구되는 추가적인 성능지표를 통해 실험을 진행한다.

3. 스트리밍 프로토콜 성능 분석

3.1 실험 환경

실험을 위해 Raspberry Pi 3 B+[13]를 스트리밍 서버로 사용하고, 스마트폰 (TG-L800S)을 클라이언트로 구현한다. 통신은 802.11n (Wi-Fi)을 활용하며, Raspberry Pi가 AP를 생성하고 스마트폰이 AP에 접속하여 영상을 스트리밍 받는 형태로 구성한다. 실험을 구성한 환경은 [그림 2]에 나타내었다.



[그림 2] 실험 환경 구성
[Figure 2] Experiment environment

각 프로토콜들은 영상의 해상도에 따라 상이한 성능을 가지기 때문에 다양한 해상도와 프레임의 영상으로 실험을 진행한다. 각 실험을 위한 영상 해상도와 초당 프레임을 [표 2]에 나타내었다. MPEG-DASH, WebRTC, RTSP 세 가지의 스트리밍 프로토콜에 대해 실험하며, RTSP는 스트리밍을 위한 전송계층 프로토콜에 대한 권장사항이 없으므로 UDP (User Datagram Protocol)와 TCP (Transmission Control Protocol)를 모두 적용하여 실험한다.

[표 2] 실험을 위한 조건

[Table 2] Cases for experiments

영상 해상도	초당 프레임
60	1280x720
30	1280x720; 640x480; 320x240
15	320x240

각 기술들에 대한 성능 측정 및 비교를 위해 사물인터넷 환경에서 제약이 되는 조건들과 스트리밍 환경에서 우선시 되는 조건들로 지표를 구성하며, 지표 및 측정 기준은 [표 3]에 나타내었다. 자원 제한적 환경에서의 전원효율성과 스트리밍의 실시간성, 안정성을 측정하기 위해 각 측정 대상을 아래와 같이 선정하였다.

[표 3] 성능 비교를 위한 지표

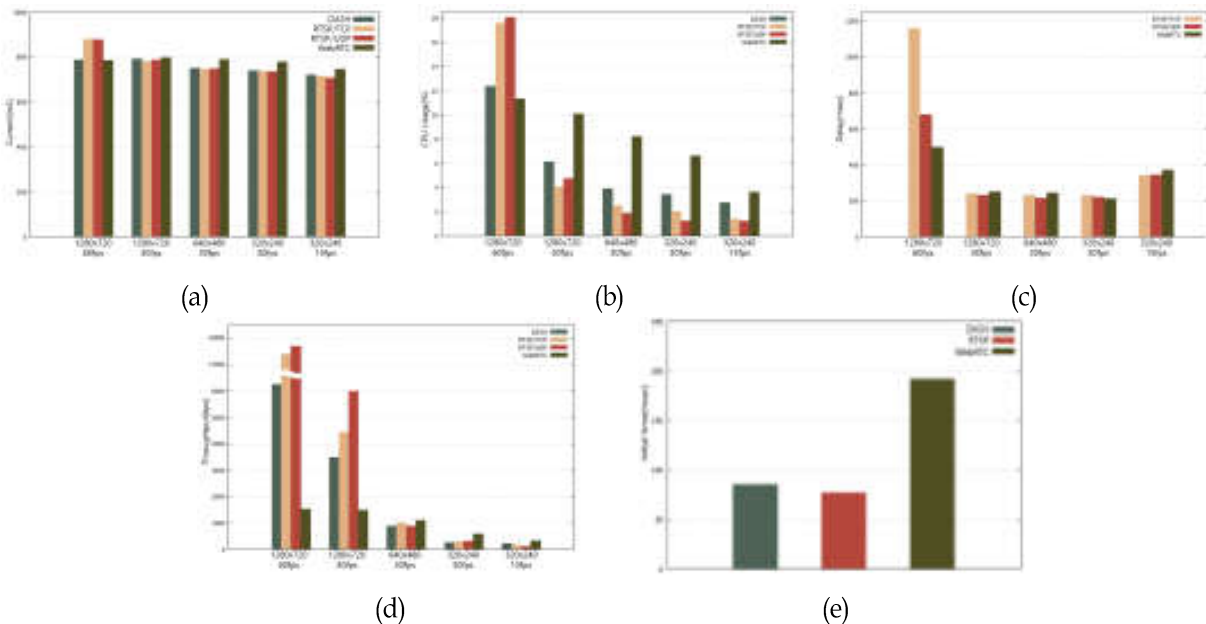
[Table 3] Performance indicators

측정 대상	측정 기준
소모 전류	스트리밍 중 전체 시스템의 소모 전류
CPU 점유율	스트리밍 기능의 CPU 점유율
영상 지연	송수신단에서 영상 프레임의 시간 차이
네트워크 사용량	스트리밍에 사용되는 전체 패킷의 bps
연결수립 시간	스트리밍을 위한 연결에 소요되는 시간

상당수의 사물인터넷 기기는 무선에서 동작하기 때문에 전원공급수단으로 배터리를 사용한다. 따라서 사물인터넷 기기의 전류소모량이 많다는 것은 그만큼 사용시간이 줄어드는 것을 의미하기 때문에 소모전류를 측정하여 전원효율성을 확인한다. CPU 점유율을 통해서 스트리밍 기능이 컴퓨팅 자원을 얼마나 사용하는지를 측정하여, 기기 성능의 경량화 가능성을 확인할 수 있다. 또한 가변의 무선 네트워크 환경에서 안정적인 스트리밍을 제공하기 위해서는 더 높은 자원효율을 가진 방식이 필요한데, 이를 위해 네트워크 사용량과 연결수립 시간, 스트리밍을 통해 영상이 지연된 시간을 측정하고 비교한다. 스트리밍을 할 때 연결수립은 1회만 수행되고 그 이후로는 할 필요가 없지만, 불안정한 네트워크 환경 등으로 인한 이유로 연결이 끊어지는 경우에 다시 연결수립을 진행해야 한다. 따라서 한 번의 연결로 스트리밍이 안정적으로 수행된다면 연결 수립 시간이 msec단위로 차이가 나는 것은 기능적인 측면에서 불편함이 적지만, 연결이 반복해서 일어나는 환경에서는 중요한 요소가 될 수 있다.

3.2 실시간 스트리밍 성능 측정

측정하는 스트리밍 방식은 MPEG-DASH, RTSP over TCP, RTSP over UDP, WebRTC이며, [표 3]에 나타난 요소들에 대해 각각 측정한다. 모든 실험은 5분간 진행하며, 3회 실험한 결과의 평균값을 사용한다. 소모전류 및 CPU점유율은 100ms주기로 측정하며, 영상 지연은 패킷 딜레이가 아닌 실제 영상의 End-to-end 지연을 측정한다. 또한 네트워크 사용량과 연결수립 시간은 Wireshark[14]를 이용하여 측정한다.



[그림 3] 실험 결과 (a) 전류 측정, (b) CPU 점유율, (c) 영상지연, (d) 네트워크 사용량, (e) 연결수립 시간
[Figure 3] Result of experiment (a) Current, (b) CPU Usage, (c) Delay, (d) Throughput, (e) Initial time

실험을 통해 측정된 결과는 [그림 3]와 같다. 단, MPEG-DASH는 지연 시간이 최소 6초, 최대 20초까지 측정되었으며 [그림 3-(c)]의 그래프에서는 제외하였다.

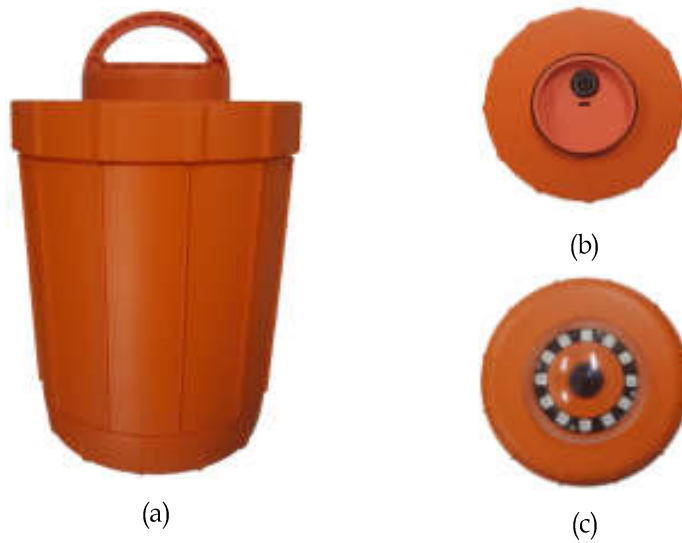
측정 결과를 통해 각 스트리밍 프로토콜의 성능이 해상도에 따라 크게 차이나는 것을 확인할 수 있다. WebRTC의 경우 고화질의 영상을 스트리밍할 때 대부분의 항목에서 가장 높은 성능을 나타내어 고화질의 영상이 요구되는 서비스에서 유리하게 적용할 수 있는 것을 확인할 수 있다. 하지만 고화질의 영상이 아닌 640x480 수준 또는 그 이하의 해상도로 영상을 스트리밍하는 경우 선원효율성이나 네트워크 사용량 등 전체적인 측면에서 RTSP를 이용한 스트리밍이 가장 높은 성능을 나타내었으며, UDP를 통해 영상을 전송할 경우 TCP를 사용해서 전송할 때 보다 조금 더 높은 성능을 나타내는 것을 확인할 수 있다.

4. 실시간 스트리밍 기능 적용

본 장에서는 3장에서 실험한 결과를 사물인터넷 기기에 적용하여 구현한다. 사물인터넷 기기는 선행연구를 통해 개발된 물고기 유인 부표를 활용한다. 물고기 유인 부표는 수면 위에서 동작하며 빛과 주파수로 물고기를 유인하고, 수면아래의 영상을 스트리밍하여 사용자가 확인할 수 있는 기능을 제공하여 낚시를 돕기 위한 사물인터넷 기기이다. 물고기 유인 부표는 원격에서 동작하며 스마트폰 애플리케이션과 End-to-end 통신을 하여 높은 배터리 효율성이 요구되고, 네트워크 환경이 제한적이지만 스트리밍 기능에 고화질의 영상이 요구되지 않는다. 따라서 물고기 유인 부표의 스트리밍 기능에 본 논문에서 실험한 결과를 활용하여 적합한 스트리밍 프로토콜로 RTSP를 선적하여 적용한다. 또한 RTSP프로토콜이 가장 높은 성능을 나타낸 640x480의 해상도와 30fps의 초당 프레임을 가지도록 적용하고 영상을 전송하기 위한 전송계층 프로토콜은 UDP를 사용하였다. 이를 통해 200ms 이내의 영상 지연을 가지는 안정적인 스트리밍이 가능하면서 0.5W 이내의 효율적인 소비전력을 가질 수 있도록 한다.

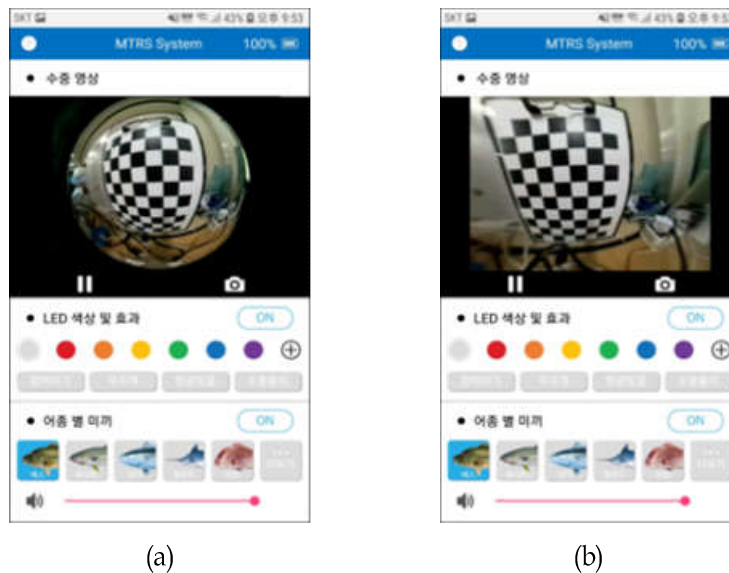
물고기 유인을 위한 부표는 [그림 4]에 나타내었다. LED는 [그림 4(c)]와 같이 부표의 하단에 장착되어 있어 빛을 발생할 경우 수면 아래쪽으로 비추게 된다. 카메라도 하단에 함께 장착되어 있어 수면 아래를 촬영한다. 주파수를 발생하기 위한 진동스피커는 부표의 내부의 벽면에 장착되는데, 주파수를 발생할 경우 진동이 벽면을 통해 물로 전달되도록 구현하였다.

부표는 사용자와 먼 거리에서 동작을 하고 사용자는 스마트폰 애플리케이션을 통해 원격에서 부표를 제어할 수 있다. 물위에 던진 부표와 사용자 간의 거리를 조절하거나, 다시 회수할 수 있도록 [그림 4(a)]와 같이 상단에 줄을 묶을 수 있는 부분을 구성하였다. 또한 상단 마개를 열어 부표를 충전하거나, 전원을 끄고 켤 수 있도록 하였다. 부표는 물 위에서 동작을 하기 때문에 상단 마개뿐 아니라 전체적인 외형에 방수설계가 되었다.



[그림 4] 물고기 유인 부표 (a) 측면부, (b) 상단부, (c) 하단부
 [Figure 4] Smart buoy (a) Side, (b) Top, (c) Bottom

부표 하단에 장착된 카메라는 어안렌즈를 장착한 카메라를 사용하여 사용자가 보다 넓은 화각을 볼 수 있도록 구현하였다. 어안렌즈를 사용하면 넓은 화각을 촬영할 수 있지만, 굴절된 영상이 스트리밍되어 화면을 볼 때 불편함이 생긴다. 따라서 평면의 카메라보다 넓은 화각을 제공하면서 동시에 영상의 변형을 줄이기 위해 왜곡보정 기능을 사용하여 굴절된 영상을 평면영상으로 복원하였다. 왜곡보정의 결과는 [그림 5]에 나타내었다.



[그림 5] 왜곡보정 처리결과 (a) 보정 전, (b) 보정 후
 [Figure 5] Result of distortion correction (a) Before correction, (b) After correction

5. 결론

본 논문에서는 제한적인 네트워크 환경에서 배터리로 동작하는 사물인터넷 기기를 위한 가장 효율적인 실시간 스트리밍 프로토콜을 찾기 위해 실험을 진행하였다. 실험은 MPEG-DASH, WebRTC, RTSP 세 가지 스트리밍 프로토콜에 대해 진행하였으며, 다섯 가지 성능지표를 통해 각 스트리밍 프로토콜의 성능을 분석하였다. 각 스트리밍 방식별로 특징이 존재하여 주어진 상황이나 제약조건에 따라 적합한 스트리밍 방식이 존재한다. MPEG-DASH는 실험에서 영상 지연의 수치가 다른 방식들에 비해 상당히 높게 측정되었다. 이 부분은 MPEG-DASH의 기술적인 특징이며, 영상을 저장하여 스트리밍하는 과정 때문에 생기는 지연이다. 영상 지연 외에 다른 성능에서는 특정 부분에서 높은 성능을 나타내지는 않지만, 다양한 품질의 영상에 대한 자원의 사용량을 비교하였을 경우에는 WebRTC를 이용한 스트리밍 방식보다 높은 자원효율성을 보인다. 따라서 실시간 스트리밍이 아닌 일정한 딜레이를 가진 영상을 유동적인 품질로 스트리밍하는 경우에 적합하다. WebRTC의 경우 스트리밍하는 영상의 품질이 변화에 따라 사용하는 자원의 수치가 다른 스트리밍 방식에 비해서 변화폭이 적어 높은 품질의 영상을 스트리밍할 때 적절하다. 따라서 네트워크 환경이나 컴퓨팅 성능 등 스트리밍을 위한 자원이 풍부한 환경에서 가장 높은 성능을 낼 수 있다. 반면 낮은 품질을 스트리밍할 경우에는 요구되는 자원이 타 스트리밍 방식보다 높아 자원이 일정수준 이하일 경우에는 적용이 어렵다. RTSP의 경우 높은 품질에서는 낮은 성능을 보이지만, 낮은 품질에서는 모든 측면에서 우수한 성능을 나타낸다. 따라서 네트워크 대역폭이 협소하거나 불안정한 상황에서 높은 품질의 영상을 스트리밍 할 필요가 없는 경우에는 RTSP를 이용한 스트리밍을 사용하여 높은 안정성을 제공할 수 있다. 우리는 실험을 통해 얻은 결과를 선행연구에서 구현한 사물인터넷 기기에 적용하였다. 이를 통해 평균 0.5W의 소비전력을 가지는 향상된 배터리 효율을 가질 수 있도록 개선하였고, 기존 제품의 경우 30미터 이상에서 스트리밍 기능이 불안정한 부분을 개선하여 제한된 네트워크 대역에서도 200ms 이내의 지연을 가지는 안정적인 스트리밍을 제공할 수 있도록 하였다. 향후 연구에서는 네트워크 성능에 따라 스트리밍 프로토콜을 가변적으로 바꾸어 제공하는 적응형 시스템을 적용하는 방안에 대해 연구할 예정이다.

References

- [1] <http://www.waterwolfd.com>, February 10 (2019).
- [2] <https://gofishcam.com>, February 10 (2019).
- [3] F. Fund, C. Wang, Y. Liu, T. Korakis, M. Zink, and S. S. Panwar. Performance of DASH and WebRTC video services for mobile users. 2013 20th International Packet Video Workshop. (2013) December 12-13; San Jose, USA
- [4] A. Aloman, A. I. Ispas, P. Ciotirnae, R. Sanchez-Iborra, and M. D. Cano. Performance evaluation of video streaming using MPEG DASH, RTSP, and RTMP in mobile networks. 2015 8th IFIP Wireless and Mobile Networking Conference (WMNC). (2015) October 5-7; Munich, Germany
- [5] I. Santos-González, A. Rivero-García, T. González-Barroso, J. Molina-Gil, and P. Caballero-Gil. Real-Time Streaming: A Comparative Study Between RTSP and WebRTC. In Ubiquitous Computing and Ambient Intelligence. (2016) November 29-December 2; San Bartolomé de Tirajana, Spain
- [6] J. Hwang, J. Jeong and S. Seol, A buoy capable of attracting specific fishes with light and frequency and streaming underwater in real-time, *Asia-pacific Journal of Multimedia Services Convergent with Art, Humanities, and Sociology*, (2019), Vol.9, No.3, pp.279-287.
- [7] J. Hwang, J. Jeong, Y. Shin and S. Seol, Performance Analysis of Streaming Protocols in Resource-Restricted Environments, *Convergence Research Letters*, (2019), Vol.5, No.1, pp.457-461.
- [8] <https://webrtc.org>, November 20 (2018).
- [9] ISO, IEC, Information technology-Dynamic adaptive streaming over HTTP(DASH). 23009-1, (2014).
- [10] H. Schulzrinne, A. Rao, and R. Lanphier, Real time streaming protocol (RTSP). RFC 2326. (1998).
- [11] I. Sodagar, The mpeg-dash standard for multimedia streaming over the internet, *IEEE MultiMedia*, (2011), Vol.4, pp.62-67.
- [12] H. Schulzrinne, S. Casner, R. Fredrick, and V. Jacobson, RTP: a transport protocol for real-time applications. RFC 3550. (2003).
- [13] <https://www.raspberrypi.org>, November 21 (2018).
- [14] <https://www.wireshark.org>, November 21 (2018).