

스마트폰 어플리케이션 기반의 컨퍼런스 중계방송 시스템 개발

*김영애 **양지희 **박구만

서울과학기술대학교 *일반대학원 미디어IT공학과, **나노IT디자인융합대학원
정보통신미디어공학전공

*young5636@seoultech.ac.kr

Development of Conference Broadcasting System based on Smart-phone Application

*Kim, Young-Ae **Yang, Ji-Hee **Park, Goo-Man

Seoul National University of Science and Technology

요약

스마트폰 기반의 모바일 플랫폼이 발달하면서 관련 서비스에 대한 수요가 급증하고 시공간의 제한 없이 누구든지 방송 콘텐츠를 제작 및 송수신 가능하게 되었다. 이에 본 논문에서는 학회 또는 세미나와 같은 컨퍼런스에서도 모바일 기반의 스트리밍 플랫폼 서비스를 제공하고자 웹캠을 통해 영상을 획득하고, 획득한 영상을 사용자 개인의 모바일 어플리케이션을 통해 실시간 스트리밍 시청 및 VOD 서비스를 제공하는 시스템을 개발하였다. 실시간 스트리밍 시청, VOD 서비스 등 다양한 기능을 탑재함으로써 기존의 컨퍼런스 관련 모바일 어플리케이션보다 더 많은 정보 제공 및 사용자 편의성 향상이 가능하다. 이 시스템을 제공함으로써 사용자는 컨퍼런스 발표 현장에 있지 않더라도 개인 모바일 어플리케이션을 통해 정보를 보다 쉽게 제공할 수 있을 것으로 기대된다.

1. 서론

스마트폰과 태블릿 PC의 보급이 빠르게 확산하면서 모바일 영상 서비스 및 콘텐츠 사업의 성장이 급속도로 이루어지고 있다. 모바일 기반의 콘텐츠를 제공하기 위한 방송 플랫폼은 시공간의 제약 없이 방송 콘텐츠의 제작, 송수신이 가능하다는 점에서 하나의 1인 미디어 서비스로 활용되고 있다.[1,2] 또한 이는 개인의 스마트폰을 사용하기 때문에 기존의 방식과는 달리 비싼 장비를 추가적으로 구입하여 스튜디오를 구성할 필요가 없기 때문에 호응도가 높다. 뿐만 아니라 실내를 벗어나 실외 환경을 활용함으로써 더욱 다양한 주제의 콘텐츠를 제공할 수 있다는 장점이 있다.[3]

이러한 모바일 플랫폼을 기반으로 하는 방송의 적용 분야는 개인 방송 시스템뿐만 아니라 화상회의 시스템[4] 등 다양해지고 있지만 학회 또는 세미나와 같은 컨퍼런스에서의 적용은 간단한 정보만을 제공하고, 실시간 중계 영상은 제공하지 않는다는 한계점이 있다. 이에 따라 본 논문에서는 세미나 또는 학회와 같은 컨퍼런스 행사에서 활용 가능한 스마트폰 어플리케이션 기반의 컨퍼런스 중계방송 시스템을 개발 및 구현하였다. 각 컨퍼런스 행사장에 설치된 카메라를 통해 영상을 획득하고 이를 스트리밍 서버를 통해 전송함으로써 개인 모바일 디바이스 또는 노트북 PC를 통해 실시간 스트리밍 및 VOD 서비스뿐만 아니라 추가적인 관련 정보에 대해 제공받을 수 있다는 장점을 가진다.

본 논문의 2장에서는 시스템에 대한 전체적인 구성 및 스트리밍 서버 구축, 어플리케이션 구현과 실험 결과에 대하여 설명한다. 끝으로 3장에서는 결론 및 향후 연구 계획에 대해 논한다.

2. 본론

2-1. 시스템 전체 구성

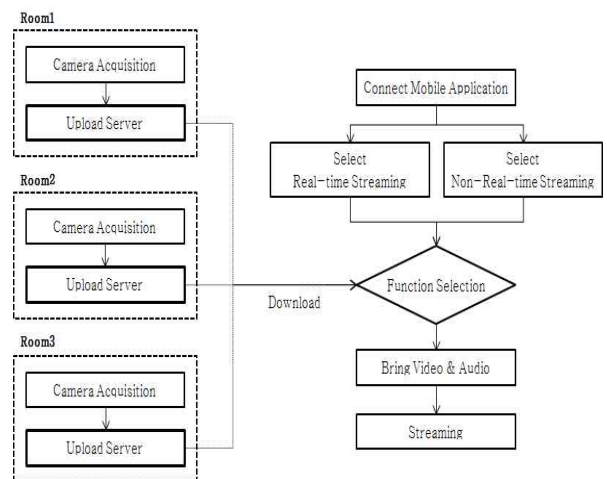


그림 1. 시스템 전체 구성도

그림 1은 본 논문에서 제안하는 시스템의 전체 구성도이다. 각 컨퍼런스 장소에 설치된 카메라를 통해 영상을 획득하여 가상의 서버로 업로드하는 송신부와 서버로부터 컨퍼런스 참가자의 모바일 디바이스 또는 PC를 통해 스트리밍 하는 수신부로 크게 나누어 시스템의 모든 수행과 작동이 이루어진다.

2-2. 송신부 - 영상 획득 및 서버 업로드

본 시스템의 송신부에서는 컨퍼런스가 개최되는 장소마다의 영상을 획득하기 위해 각각 다른 장소에 카메라를 1대씩 설치한다. 카메라는 발표자와 스크린이 한 화면에 보이도록 위치와 각도를 설정하여 촬영한다. 획득된 영상은 실시간 스트리밍과 동시에 자동으로 저장함으로써 이후에 VOD 서비스로 제공 가능하도록 가상의 서버에 업로드시킨다. 가상의 서버는 auto scaling 기법을 적용하여 유동적으로 서버 저장소의 크기를 확장 또는 축소함으로써 그 성능을 유지하고 비용을 절감할 수 있도록 구성한다. 그림 2는 실험을 진행하기 위해 설치한 카메라를 통해 획득한 입력 영상 결과를 나타낸다.



그림 2. 컨퍼런스 장소 영상 획득 예시

2-3. 수신부 - 서버 다운로드 및 스트리밍

본 시스템의 수신부에서는 가상의 서버에 업로드 된 영상을 사용자의 모바일 디바이스를 통해 제공하기 위해 어플리케이션을 제작하였다. 어플리케이션을 통해서 실시간 스트리밍 영상 시청뿐만 아니라 컨퍼런스를 주최하는 기관 정보, 컨퍼런스 일정 및 장소 정보, 컨퍼런스 프로그램 정보 등 간단한 정보를 함께 제공 받을 수 있다.



그림 3. 모바일 어플리케이션 구성 결과 화면

2-4. 실험 및 결과

컨퍼런스 장소의 영상 획득은 웹캠을 이용하여 총 3군데 각각 다른 교내 강의실에서 발표하는 촬영하였으며, 가상 클라우드 서버의 개발 환경은 AWS(Amazon Web Services)를 활용하여 Linux Ubuntu 16.04 기반으로 구축하였다. 사용자가 제공 받을 어플리케이션은 JAVA 언어를 기반으로 Android SDK와 Red5 Pro 라이브러리를 사용하여 Android Studio 2.3.3에서 구현하여 성능 실험을 진행하였다.

그림 4는 구현된 어플리케이션을 실행한 결과 모습이다. 실험을 통해 총 10명의 사용자가 동시에 접속하여 실시간 스트리밍 영상을 시청하였음에도 불구하고 1초 이내의 지연시간으로 실시간 영상 제공이 가능한 것을 확인하였다. 사용자는 영상뿐만 아니라 웹캠을 통해 실

간 음성 정보도 획득 가능하였으며, 어플리케이션의 다른 메뉴를 통해 다양한 정보도 제공 가능하였다.

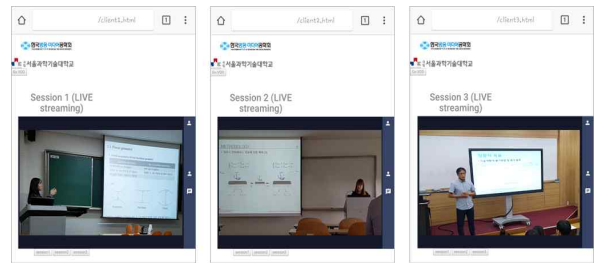


그림 4. 시스템 실험 결과 화면

3. 결론

본 논문에서는 컨퍼런스 발표 영상을 개인 모바일 어플리케이션을 통해 실시간 스트리밍 또는 VOD 서비스를 제공받는 방법을 제시 및 구현하였다. 기존 컨퍼런스 어플리케이션과 다르게 단순 전달 정보만 제공하는 것이 아닌 사용자가 직접 스트리밍 받을 컨퍼런스 영상을 선택하고, 저장된 VOD도 제공 받으므로써 사용자 편의성 제공 및 컨퍼런스 참여도 향상의 효과를 기대한다. 또한 기존 모바일 기반의 방송 플랫폼과는 다르게 360도 카메라, AR/VR 등과 융합하여 응용 가능 분야에 대한 확장성이 좋다는 점에서 차별성이 있다.

향후 연구로는 본 논문에서 제안한 개인 모바일 어플리케이션을 통해 실시간 스트리밍 영상을 시청하면서 질의응답이 가능하도록 채팅창 기능을 추가할 것이다. 더 나아가 설치된 카메라 이외에도 서로 간 학회 주변 영상 등을 촬영하여 업로드하고 공유함으로써 사용자 간의 소통 기능을 향상시키기 위한 연구를 진행할 것이다.

감사의 글

이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No.2016-0-00099, 제작편리성과 실감 시청 체험 극대화를 위한 개인방송 제작 기술 개발)

참고문헌

- [1] 임제완, 박병호, “모바일 개인방송 시청 요인 분석 : HMSAM 모델을 중심으로”, Information Systems Review 제18권 제4호, pp.89-106, 2016
- [2] 윤영석, 이현우, “개인 방송 플랫폼 기술”, 한국통신학회지(정보와 통신) 제33권 제4호, pp.56-63, 2016
- [3] 송민정, “글로벌 5대 MCN 미디어 기업들의 비즈니스 모델 연구-과파적 혁신 이론을 토대로”, 방송통신연구 제96권, pp.38-67, 2016
- [4] Osuagwu, O. E., Ndigwe, C., Ihedigbo, C., Suleiman, U., Babatunde, O., “Video conferencing: most effective technology to run assemblies and meetings for large audience dispersed in distant locations: is it feasible to deploy in Nigeria?”, West African Journal of Industrial and Academic Research, Vol.17, No.1, pp.73-81, 2017