

---

# HTML5을 활용한 실시간 영상 스트리밍 시스템

반태학\* · 정한길\* · 송현옥\* · 육정수\* · 정희경\*

\*배재대학교 컴퓨터공학과

## A Study on HTML5 using Real-time Video Streaming System

Tae-Hak Ban\* · Han-Gil Jeong\* · Hyun-Ok Song\* · Jung-Soo Yuk\* · Hoe-Kyung Jung\*

\*Department of Computer Engineering, PaiChai University

E-mail : banth@pcu.ac.kr, jhg-0518@daum.net, paperblue21@hanmail.net, kimics1@naver.com,

hkjung@pcu.ac.kr

### 요 약

현재 대다수 스트리밍 서비스들은 특정 S/W나 별도의 프로그램 설치를 통해 실시간 스트리밍 서비스를 지원하는 실정이고, 저장되는 영상에 대해 저장의 종료 전까지는 영상에 대해 편집 및 사용이 불가능 하였다.

이에 본 논문에서는 사용자가 저장하고 있는 영상에 대해 멀티스레드 기법을 적용한 시스템을 기반으로 별도의 플러그인이나 프로그램의 설치 없이 HTML5을 활용하여 웹 콘텐츠를 제작, 웹 브라우저를 이용한 실시간 스트리밍 시스템을 기술한다.

이는 스트리밍 실시간으로 저장되는 영상에 대해 편집 및 사용이 가능하며, 실시간 커뮤니케이션 및 미디어 공유를 필요로 하는 웹 서비스 및 스트리밍 분야에 활용될 것이다.

### ABSTRACT

The majority of current streaming services are particular S/W I a separate setup program supports real-time streaming service a reality, will be stored until the end of the video, save for a video about editing and is not available.

In this paper, allows the user to save and apply the multi threaded techniques for video based on a system without a separate plug-ins, or installation of programs HTML5 by taking advantage of Web content authoring, real-time streaming system using a Web browser.

This saves the video streaming in real time can be used for editing and, on real-time communications and media sharing and streaming Web service you need and will be utilized in the field.

### 키워드

멀티 쓰레드, 실시간 스트리밍, 실시간 영상 저장, HTML5

## 1. 서 론

영상 콘텐츠의 제공자 측면에서 웹 브라우저를 통해 영상을 스트리밍 할 때 기존의 원본영상이 녹화 또는 편집의 단계가 완료된 영상에 대해서만 클라이언트 측에서 재생 및 스트리밍이 가능하다[1,2.] 하지만 현재 실시간으로 저장되고 있는 영상은 저장이 완료되기 전까지는 확장자 및 파일의 정보가 명확하지 않아 웹을 통해 스트리밍 서비스를 제공하지 못하는 실정이다.

이에 본 논문에서는 캡처 프로그램이나 화상카

메라를 통해 사용자가 녹화나 편집을 실행 한 후 현재 녹화나 편집되고 있는 영상에 대해 멀티스레드 기법을 통해 영상의 녹화가 종료되지 않더라도 일정시간 단위의 저장되고 있는 영상을 쓰레드 저장기법을 통해 실시간으로 웹을 통한 영상의 스트리밍 서비스가 가능한 시스템에 대해 제안한다[3,4]. 제안하는 시스템은 2장에서 설명한다. 시스템의 구현은 3장에서 기술한다. 결론 및 향후연구는 4장에 기술한다.

## II. 시스템의 설계

### 2.1 시스템 구성

시스템의 구성에서는 캡처 프로그램이나 가상 카메라 또는 웹캠으로부터 저장이나 편집되는 영상에 대해 웹에서 스트리밍 서비스 제공 및 재생이 가능하도록 서버에서 멀티쓰레드 기법을 통해 분산처리 저장시스템에 대해 설명한다. 그림 1의 시스템 구성도에서는 스트리밍 제공자, 서버, 스트리밍 사용자 3가지로 분류하였다. 스트리밍 제공자는 웹캠이나 기존영상에 대해 실시간으로 저장 및 편집한다. 서버에서는 편집이나 저장되는 영상에 대해 실시간 멀티쓰레드 분산처리 기법을 적용시켜 실시간으로 일정 시간 별 웹 콘텐츠 형식으로 변환한다. 스트리밍 사용자는 웹 브라우저를 통해 스트리밍 제공자로부터 제공되는 영상에 대해 서버로부터 실시간으로 제공받아 재생이 가능하다.

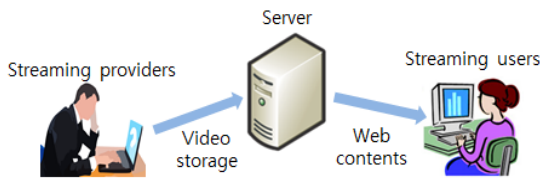


그림 1. 시스템 구성도

## III. 시스템의 구현

본 장에서는 전반적인 시스템의 동작 과정에 대한 설명과 분석을 다룬다.

### 3.1 Upload File Select

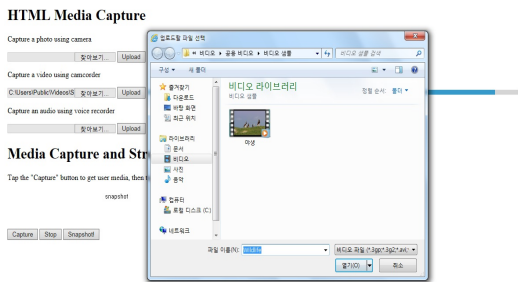


그림 2. HTML5 Web Browser 상에서 Video File 선택

그림 2는 위에서부터 Image, Video, Audio File들을 선택하여 Upload 할 수 있는 것 중 Video File만 찾아 Upload 하는 것을 보여준다. 또한 분산처리 저장시스템을 통해, 한 번에 여러 개의 File 또는 여러 형식의 파일을 한꺼번에 Upload할 수도 있다. 하지만, 한 번에 여러 형식, 여러 개의 File들을 Server로 보내게 될 경우 효율성이 떨어져 진행 속도가 늦어진다.

### 3.2 Video File Playing

Video File을 재생하는데 사용한 영상은 Chrome에

서 기본적으로 제공하는 브라우저에 기본 제공 재생 컨트롤을 표시하도록 지정하였다.

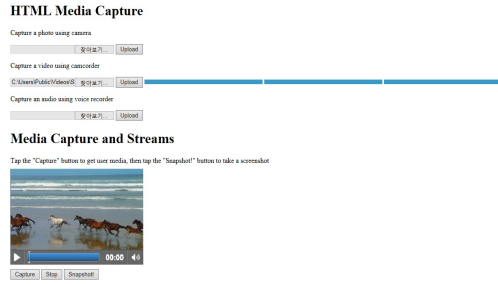


그림 3. File Upload 완료 후 Video 재생

그림 3은 Upload된 3개의 Video 중 하나를 선택하여 Media Capture And Streams 영역에서 재생하는 것을 보여준다. 사용된 영상은 Window의 기본 라이브러리에 포함된 Video File을 mp4 형식으로 변환하여 이용한 것이다. 출력된 영상 아래의 버튼들은, 각각 재생 중인 화면을 찍는 Capture Button과 Capture가 수행 중인 것을 중지시키는 Stop Button, 그리고 Capture된 영상의 한 장면을 아래로 출력시키는 Snapshot Button으로 되어 있다.

```
var video = document.querySelector('video');
var canvas = document.querySelector('canvas');
var ctx = canvas.getContext('2d');
var localMediaStream = null;
```

그림 4. Video Capture를 위한 변수 설정

그림 4는 재생 중인 Video를 Capture하여 Canvas에 출력하기 위한 변수 설정을 나타낸다. 각각 변수별로 Video, Canvas 요소를 지정해주고, Canvas에 출력될 Image를 var ctx = canvas.getContext("2d")를 입력하여 2d로 나타내게 하며 localMediaStream에 Null 값을 넣는다.

```
function snapshot() {
    if (localMediaStream) {
        ctx.drawImage(video, 0, 0); document.querySelector("img").src =
        canvas.toDataURL("image/jpeg");
    }
}
```

그림 5. Video에서 Capture한 image를 JPEG 형식의 Image File로 변환

그림 5는 Canvas상에 표현할 Capture한 image를 제대로 된 Image로 출력하기 위한 설정 방법을 보여준다. Video 상에서의 Capture가 완료되면, Snapshot Button을 누르면 Canvas의 영역 안으로 Capture된 영상의 사진이 전송되어 출력된다. 이때, 출력되는 사진의 형식은 JPEG형식으로 지정되어 출력된다. 단, 직접 Image File 형식을 지정하지 않으면, 기본으로 지정되는 PNG 형식의 Image File로 출력이 된다.

#### IV. 결 론

본 논문에서는 사용자가 편집하는 영상이나 화상카메라를 통해 사용자가 녹화나 편집을 실행 한 후 현재 녹화나 편집되고 있는 영상에 대해 멀티쓰레드 기법을 통해 영상의 녹화가 종료가 되지 않더라도 일정시간 단위의 저장되고 있는 영상을 쓰레드 저장기법을 통해 실시간으로 웹을 통한 영상의 스트리밍 서비스가 가능한 시스템에 대해 기술하였다. 이는 현재 영상의 녹화가 EOF가 되기 전까지는 스트리밍을 제공하지 못하는 현재 스트리밍 서비스의 단점을 해결할 수 있을 것이다.

향후 본 논문에서 제시한 시스템을 바탕으로 멀티미디어 콘텐츠 적용 및 분석에 대한 연구할 필요할 것으로 사료된다.

#### 참고문헌

- [1] Seongwoo Kwak, Hong Choi, Jungmin Yang, "A Real-time Video Transferring and Localization System in HSDPA Network", The Korea Institute of Electronic Communication Sciences, Vol.7, No.1, pp.21-26, 2012.2
- [2] Seungju Jang, Wonyeong Heo, Hyunmin Yoo, Changhoon Lee, Woocho Shin, "A Study of Real-Time Video Streaming Data Service on the Linux Server", The Korea Institute of Information and Communication Engineering, Vol.17, No.4, pp.893-901, 2013.4
- [3] Hosun Yoo, Ilseung Kim, Taeho Kim, Jeehyun Jeon, Jechang Jeong, "H.264/AVC Decoder Parallelization Methods for Real-time Full-HD Image Processing", The Korea Society of Broadcast Engineers, Vol.1, pp.452-455, 2012.7
- [4] Seungcheol Choi, Yunhyoung Kim, Mankyu Lee, Seokrim Choi, "Development of Adaptive Streaming Systems for Hybrid TV Service", The Korea Contents Association, Vol.39, No.7, pp.467-476, 2014.7