

## 동영상 재생 안드로이드 프로그램 구현

임재걸<sup>○</sup>, 이경훈<sup>\*</sup>

<sup>○\*</sup> 동국대학교 컴퓨터공학과

e-mail: yim@dongguk.ac.kr, darkleekh@dongguk.ac.kr

## Implementation of an Android Video Player

Jaeeol Yim<sup>○</sup>, Gyeonghoon Lee<sup>\*</sup>

<sup>○\*</sup> Dept. of Computer Engineering, Dongguk University at Gyeongjoo

### ● 요약 ●

본 논문은 동영상 재생 안드로이드 프로그램 구현 사례를 소개한다. 이를 위하여 동영상 스트리밍 서버도 구축하였고, 코덱도 사용하였다. 근래에 인터랙티브 동영상에 대한 관심이 큰데, 본 연구는 이러한 콘텐츠 개발의 첫 걸음이다.

키워드: 안드로이드(android), 동영상(video), 재생기(player)

### I. 서론

인터랙티브 동영상은 재생 중에 사용자가 화면상의 관심 있는 개체를 지시하면 현재 콘텐츠를 중단하고 관련 콘텐츠를 재생하거나 사용자가 원하는 바를 참고하여 스토리를 바꾸어 줌으로써 사용자의 흥미를 더욱 충족시킨다. 인터랙티브 콘텐츠 제작 기술 개발의 첫걸음으로 콘텐츠 재생 안드로이드 프로그램을 개발한 사례를 소개한다.

### II. 관련 연구

동영상 재생 메소드를 제공하는 클래스에는 VideoView 클래스와 녹화도 가능한 MediaPlayer 클래스가 있다. 비디오는 안드로이드 프로그램의 리소스로 내장하기, SD 카드에 저장하기, 서버에서 다운 받거나 실시간으로 받아서 재생하기가 있다. 서버에서 받아오려면 스트리밍 서버를 설치해야 하는데 다운스트리밍 서버를 비롯한 많은 서버가 사용되고 있다.

### III. 프로그램 구현 및 실험

안드로이드 프로그램의 res/raw에 비디오 파일을 저장(raw파일 밑에 kum.mp4 를 kum으로 저장)하고 videoView를 이용하여 재생하는 루틴은 다음과 같다.

```
raw2file(this, R.raw.kum, "kum.mp4");
String path=getFilesDir().getAbsolutePath()+"/kum.mp4"
videoView.setVideoPath(path);
videoView.start();
```

SD 카드에 비디오 파일을 저장하려면 DDMS에서 SD 카드의 권한을 그림 1처럼 쓰기 가능으로 변환해야 한다. 그림 2는 SD 카드에 있는 동영상 파일을 접근하여 재생하는 실행 화면이다.



그림 1 SD 카드의 Permissions 변경 화면  
Fig. 1 The permission of SD card should be writable



그림 2. SD 카드 내의 동영상 재생화면  
Fig. 2. A Screenshot of Playing Video from SD card

동영상 파일 형식은 여러 가지가 있는데 본 실험에서는 mp4 형식만 재생이 가능하여 다른 형식의 파일은 인코더를 이용하여 형식을 변환한 후 사용하였다. 파일 형식을 변환하는 프로그램은 여러 가지가 있는데 본 실험에서는 Umile encoder를[1] 사용하였다.

서버에 있는 동영상 파일을 실시간으로 다운 받아 재생하는 것

이 가능하도록 하려면 다윈 스트리밍 서버를[2] 설치해야 한다. 설치 방법은 다음과 같다.

- 1) windows 서버 설치
- 2) iis (인터넷 정보 서비스) 설정
- 3) 액티브Perl [3] 프로그램 설치



그림 3 액티브 펄 설치 화면

Fig. 2 A Screenshot of Installing ActivePerl

- 4) 다윈 윈도우 서버용 설치

그림 3은 다윈 서버 실행 화면이다. 메뉴에서 Connected Users 를 선택하면 접속 중인 클라이언트들의 리스트가 출력된다.

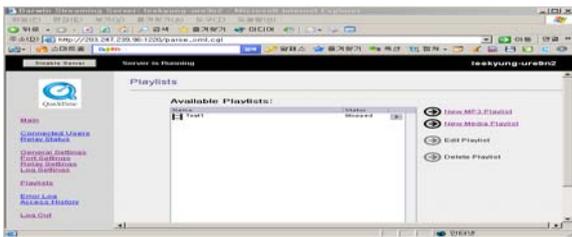


그림 4 다윈 서버 실행 화면

Fig. 2 A screenshot of Darwin server

재생될 동영상 파일을 서버에 올려 놓을 때, 다음과 같이 MP4Box를 실행해 준다.

MP4Box -hint 파일명.MP4

서버에 있는 동영상 파일을 실시간으로 다운 받아 재생하려면 재생 프로그램의 AndroidManifest.xml에 INTERNET 허용을 명시해야 한다.

video.setVideoURI(Uri.parse("rtsp://203.247.239.96/kum.mp4"));

인터랙티브 콘텐츠를 만들려면 정지, 실행 등의 버튼도 있어야 하고, 정지 당시의 파일 위치를 기억하였다가 재생시 그 곳으로부터 시작해야 한다. 이를 위하여 MediaPlayer 클래스를 이용하여 동영상을 재생한다. 우선 재생할 파일을 사용자가 선택할 수 있도록 리스트 뷰에 동영상을 재생하는 클래스를 바인딩한다. 리스트 뷰에 있는 목록들을 selectonItemClickListener을 이용하여 선택 시 해당 이벤트 처리를 한다. 이벤트 내용은 인텐트를 사용하여 하위 액티비티를 호출하고 하위 액티비티에서 동영상을 재생시킨다. 이때, 하위 액티비티를 android.manifest에 등록시켜야 한다.

surface view를 이용하여 동영상을 재생할 화면을 지정하고, 버튼 클릭 이벤트를 통하여 재생과 정지버튼에 대한 이벤트 처리를 한다. Surface holder로 화면에 대한 크기를 고정시키고

setDatasource에 Uri 주소를 아래와 같은 형식으로 넣어준다.

rtsp:"xxx.xxx.xxx.xx/파일명"

멈춤은 MediaPlayer.stop(), 재생은 start() 메서드로 구현한다.

HTTP를 연결할때 현재의 네트워크 상황을 체크하여 wifi가 아닌 3G일 때 그 네트워크를 종료하고 wifi 네트워크로 접속을 해야 한다. Connectivity Manager의 Network Info메서드를 사용하여 현재 네트워크 상태를 체크하고 네트워크설정이 Wifi가 아니면 getNetwork Info메서를 사용하여 Wifi가 사용 가능한지 체크하고 쓸 수 있으면 Wifi에 접속한다.



그림 5 MediaPlayer 실행 화면

Fig. 5. A Photo of a Smart Phone running the player

이어보기를 위하여 getCurrentPosition(); 으로 현재 상영중인 위치를 파악하여 Lengthdata라는 변수에 저장하였다가 SeekTo() 메소드에 넣어서 지금까지 본 시간 만큼을 건너뛰고 재생을 시작한다.

재생화면을 클릭할 때, 클릭 지점의 좌표를 가져오기 위하여 Event.getPointerCount를 활용하여 멀티터치이벤트인지 싱글 터치 이벤트인지를 파악하고, Event.getX()메소드, Event.gety()메소드로 터치이벤트가 발생한 좌표 값을 가져온다.

#### IV. 결론

본 논문은 인터랙티브 콘텐츠 개발을 위하여 필요한 스트리밍 서버 구축, 파일 형식 변환, 힌트 처리, 그리고 클라이언트 쪽의 mediaPlayer, 출력화면 지정, 비디오 선택 방법, 이어보기 구현 방법, 버튼 클릭 위치 찾기 방법 등 다양한 방법 구현 결과를 소개하였다.

#### 참고문헌

- [1] <http://www.umile.kr/encoder/>
- [2] <http://www.jquery.kr/>
- [3] <http://www.activestate.com/activeperl>