

석사학위논문

모바일 환경에서 OTT 시청률
측정을 위한 콘텐츠 식별 기법

2024년

한 성 대 학 교 대 학 원

컴 퓨 터 공 학 과

컴 퓨 터 공 학 전 공

김 성 진

석사학위논문
지도교수 허준영

모바일 환경에서 OTT 시청률 측정을 위한 콘텐츠 식별 기법

Content Identification Techniques for Measuring
OTT Viewership in Mobile Environments

2024년 6월 일

한성대학교 대학원

컴퓨터공학과

컴퓨터공학전공

김성진

석사학위논문
지도교수 허준영

모바일 환경에서 OTT 시청률
측정을 위한 콘텐츠 식별 기법

Content Identification Techniques for Measuring
OTT Viewership in Mobile Environments

위 논문을 공학 석사학위 논문으로 제출함

2024년 6월 일

한성대학교 대학원

컴퓨터공학과

컴퓨터공학전공

김성진

김성진의 공학 석사학위 논문을 인준함

2024년 6월 일

심사위원장 정진만 (인)

심 사 위 원 민홍 (인)

심 사 위 원 허준영 (인)

국 문 초 록

모바일 환경에서 OTT 시청률 측정을 위한 콘텐츠 식별 기법

한 성 대 학 교 대 학 원
컴 퓨 터 공 학 과
컴 퓨 터 공 학 전 공
김 성 진

스마트폰과 태블릿 등의 모바일 기기의 보급과 미디어 콘텐츠의 소비 방식의 변화로 인해 OTT 콘텐츠가 많이 제작 및 유통되고 있다. 이러한 배경 속에서, 콘텐츠의 품질과 인기를 정확하게 측정하기 위한 새로운 시청률 측정 방법이 필요해졌다. 본 논문에서는 제삼자가 OTT 시청률을 측정하기 위해 콘텐츠의 소리를 캡처하여 오디오 지문을 통해 콘텐츠를 식별하는 방법을 제시한다. 이 연구는 다양한 OTT 콘텐츠와 소비 패턴을 정확하게 파악할 수 있는 새로운 도구를 제공함으로써 미디어 산업에 기여할 것으로 기대한다. 콘텐츠 제작자와 배급사가 시청률 데이터를 기반으로 보다 정밀한 시장 분석과 전략 수립을 할 수 있다. 또한, 이 기술로 시청자의 습관을 파악하고 타겟 시청자에게 보다 개인화된 콘텐츠를 제공하는 데 중요한 역할을 할 것으로 기대한다. 기존의 시청률 측정 방식은 주로 전통적인 텔레비전 방송에 한정되어 있어, 다양한 디바이스에서 소비되는 OTT 콘텐츠의 시청 패턴을 반영하는 데 한계가 있었다. 하지만

본 연구에서는 스마트폰, 태블릿 등 다양한 모바일 기기에서의 오디오 데이터를 수집하여, 이를 바탕으로 실시간 시청률을 측정할 수 있다. 이를 통해 다양한 OTT 플랫폼에서 제공되는 콘텐츠를 효과적으로 모니터링하고, 시청률 데이터를 실시간으로 수집할 수 있다. 이러한 기술적 접근은 단순히 시청률 측정에 그치지 않고, 더 나아가 콘텐츠 제작자와 배급사에 중요한 인사이트를 제공할 수 있다. 본 연구는 OTT 서비스의 발전과 함께, 시청률 측정 방법을 제안함으로써 미디어 산업의 경쟁력을 높이는 데 기여할 것이다.

【주요어】 OTT, 안드로이드, 콘텐츠 식별, 오디오 지문, 미디어 산업

목 차

제 1 장 서 론	1
제 2 장 관련 연구	3
제 1 절 오디오 지문 기술	3
제 2 절 OTT 시청률 측정	3
제 3 장 오디오 지문 기술	5
제 1 절 오디오 지문 기술 개요	5
제 2 절 피크 기반 오디오 지문 생성	5
제 3 절 다중 시간 세그먼트 피크 기반 오디오 지문 생성	8
제 4 장 OTT 콘텐츠 시청률 측정	9
제 1 절 콘텐츠 오디오 캡처	9
1) 콘텐츠 오디오 캡처 개요	9
2) OTT 애플리케이션 동작 감지	9
3) 소프트웨어 기반 캡처	10
4) 리버스 엔지니어링을 통한 캡처	11
5) 하드웨어 기반 캡처	12
제 2 절 오디오 지문 생성 및 식별	13
1) 오디오 지문 생성	13
2) 콘텐츠 식별 및 시청률 계산	14
제 5 장 실험결과 및 성능평가	16
제 1 절 전체 실험환경	16
제 2 절 성능평가	16

제 6 장 결 론	18
참 고 문 헌	19
ABSTRACT	22

표 목 차

[표 3-1] 피크 기반 오디오 지문 생성 의사 코드	7
[표 4-1] OTT 프로그램 확인 의사 코드	01
[표 4-2] 오디오 캡처 의사 코드	11
[표 5-1] 지문 검색 정확도	61

그림 목 차

[그림 4-1] 캡처 프로그램 런치	01
[그림 4-2] 넷플릭스 AndroidManifest	21
[그림 4-3] 티빙 AndroidManifest	21
[그림 4-4] SBC를 이용한 캡처 방식	31
[그림 4-5] 원본 PCM 데이터	41
[그림 4-6] Base64 인코딩	41
[그림 4-7] 오디오 지문 검색 순서도	51
[그림 5-1] 샘플링 레이트별 지문 생성 시간	71

제 1 장 서론

최근 몇 년간, 스마트폰과 태블릿과 같은 모바일 기기의 보급률이 급증하면서, 미디어 콘텐츠의 소비 방식에 혁신적인 변화가 일어났다. 전통적인 텔레비전 방송에서 벗어나, OTT(Over-The-Top) 서비스를 통한 콘텐츠 소비가 새로운 표준으로 자리 잡았다. 이러한 변화는 콘텐츠 제작자들에게 새로운 기회를 제공하며, OTT 플랫폼을 통해 다양한 콘텐츠가 제작되고 있다. 이에 따라, 콘텐츠의 품질과 인기를 보다 정확하게 측정하는 새로운 방법의 필요성이 대두되었다.

기존의 시청률 측정 방법은 주로 전통적인 방송에 초점을 맞추었으나, OTT 플랫폼의 다양한 콘텐츠와 그 사용자의 소비 패턴을 정확하게 반영하는 데에는 한계가 있었다. 본 논문에서는 이러한 문제를 해결하기 위해, 제삼자가 OTT 시청률 측정하는 방법을 제안한다. 이 방법은 콘텐츠의 소리를 캡처하여 오디오 지문 변환을 통해 콘텐츠를 식별하는 것으로, 이를 통해 시청자의 시청 습관과 콘텐츠의 인기를 보다 정확하게 측정할 수 있다. 이 기술은 사용자의 시청 습관과 콘텐츠의 인기를 정확히 파악할 수 있는 중요한 도구로 활용될 수 있다.

OTT 서비스의 시청률 측정은 단순히 숫자를 넘어서, 콘텐츠 제작자와 배급사에 중요한 통찰력을 제공한다. 이는 시장 동향을 이해하고, 타겟 오디언스에게 더욱 적합한 콘텐츠를 제공하는 데 필수적이다. 따라서, 본 논문에서 제안하는 측정 기술은 콘텐츠 제작 및 배급의 전략을 수립하는 데 중요한 역할을 할 것이다. 이러한 배경과 필요성을 바탕으로 본 논문은 모바일 환경에서 OTT 콘텐츠의 시청률을 정확하게 측정하기 위한 실용적인 방법을 제시하고자 한다.

OTT 플랫폼은 넷플릭스, 디즈니+, 아마존 프라임 비디오와 같은 글로벌 기업부터, 지역별로 특화된 콘텐츠를 제공하는 로컬 플랫폼까지 다양하게 존재한다. 각 플랫폼은 고유한 사용자 기반을 가지고 있으며, 이들의 시청 패턴은 서로 상이하다. 이러한 복잡성을 고려할 때, 보다 정밀하고 사용자 친화적인 시청률 측정 방법이 요구된다. 특히, 모바일 환경에서의

데이터 수집과 분석은 실시간으로 이루어져야 하므로, 시스템의 효율성과 정확성이 무엇보다 중요하다.

제 2 장 관련 연구

제 1 절 오디오 지문 기술

오디오 지문 기술은 특정 오디오 파일을 고유하게 식별할 수 있는 특징 벡터를 생성하여, 대규모 데이터베이스에서 빠르게 검색하고 비교하는 기술이다. 대표적인 연구로는 Shazam이 있다. Wang은 주파수와 시간의 조합에서 중요한 피크를 추출하여, 이를 고유한 오디오 지문으로 사용한다. 이러한 지문은 데이터베이스에서 해시 테이블을 통해 빠르게 검색할 수 있다.

Phillips et al.은 오디오의 주파수와 도메인에서 특징을 추출하고, 이를 기반으로 해밍 거리 등을 이용해 유사도를 계산하는 방법을 제안했다. 이 방법은 잡음에 강한 오디오 인식 성능을 보여주었다. Ong and Herrmann은 멜 주파수 스펙트럼 계수(MFCC)를 사용하여 오디오 지문을 생성하는 방법을 제안하였다. MFCC는 음향 신호의 주파수 스펙트럼을 압축하여 특징을 추출하는 데 널리 사용된다.

이외에도, 최근 연구에서는 딥러닝 기술을 이용하여 오디오 지문의 정확성을 높이는 시도가 이루어지고 있다. Convolutional Neural Networks(CNN)을 활용한 오디오 인식 모델은 기존의 특징 추출 방법보다 높은 성능을 보이고 있으며, 이를 통해 다양한 응용 분야에서 오디오 지문 기술이 활용되고 있다.

제 2 절 OTT 시청률 측정

OTT 콘텐츠의 시청률 측정에 관한 기존 연구로는 최선역의 연구가 있다. 이 연구에서는 스마트폰을 활용하여 OTT 서비스의 시청 패턴을 추적하는 애플리케이션을 설계하였다. 특히 티빙(tving) 서비스를 중심으로 사용자의 시청 데이터를 수집하고 분석하여 시청 패턴을 파악하는 방법을

제안하였다. 이 연구는 OTT 서비스의 시청률을 보다 정밀하게 측정할 수 있는 가능성을 제시하였으나, 본 논문에서는 이러한 방법을 확장하여 제삼자가 독립적으로 시청률을 측정할 수 있는 방법을 제안한다.

또한, OTT 플랫폼별로 시청률을 측정하는 방식이 다를 수 있다. 예를 들어, 넷플릭스는 자체적으로 수집한 데이터를 바탕으로 콘텐츠의 인기를 평가하며, 이 데이터를 활용하여 사용자에게 맞춤형 콘텐츠를 추천한다. 반면, 제삼자에 의한 독립적인 시청률 측정은 여러 플랫폼에 걸쳐 공통된 기준을 적용할 수 있어, 보다 객관적이고 일관된 데이터를 제공할 수 있다.

제 3 장 오디오 지문 기술

제 1 절 오디오 지문 기술 개요

오디오 지문 기술은 오디오 파일의 고유한 특성을 추출하여 일종의 지문을 생성하는 기술이다. 이는 오디오 파일의 식별, 검색, 매칭 등에 사용되며, 음악, 방송, 저작권 보호, 광고 모니터링 등 다양한 응용 분야에서 중요한 역할을 한다. 본 장에서는 오디오 신호로부터 지문을 추출하는 기술 중에서 피크 기반의 지문 추출 방법과 본 연구에서 제안하는 다중 시간 세그먼트 피크 기반 오디오 지문 생성 방법을 소개한다.

오디오 지문 기술은 일반적으로 크게 두 가지 단계로 나뉜다. 첫 번째는 오디오 신호로부터 특징 벡터를 추출하는 단계이고, 두 번째는 추출된 특징 벡터를 데이터베이스에서 검색하고 비교하는 단계이다. 이 과정에서 중요한 것은 잡음이나 왜곡에 강인한 특징을 추출하는 것이다. 이를 위해 다양한 신호 처리 기법과 알고리즘이 사용된다.

제 2 절 피크 기반 오디오 지문 생성

피크 기반 오디오 지문 생성은 주파수 도메인에서 중요한 피크를 탐지하여 고유한 오디오 지문을 생성하는 방법이다. 이 방법은 스펙트럼 피크를 이용하여 오디오의 특징을 추출한다. 피크 기반 오디오 지문 생성 과정은 다음과 같다.

먼저 오디오 신호를 일정 길이의 청크로 분할한다. 여기서 청크 길이는 주어진 세그먼트 길이를 샘플링 레이트로 곱한 값이다. 본 연구에선 샘플링 레이트를 22050Hz의 오디오 데이터를 사용한다. 세그먼트 길이가 5초이고 샘플링 레이트가 22,050Hz일 경우, 청크 길이는 $5\text{초} \times 22,050\text{Hz} = 110,250$ 이 된다. 청크 크기를 키우면 주파수 해상도가 증가하고, 시간 해상도가 감소한다. 반대로 청크 크기를 줄이면 주파수 해상도가 감소하고,

시간 해상도가 증가한다. 주파수 해상도가 증가하면 음악과 같이 주파수 성분이 지속적으로 변하지 않는 신호에서 효과적이다. 시간 해상도가 증가하면 순간적인 변화나 짧은 시간 동안의 변동을 감지하는 데 효과적이다.

각 청크에 대해 FFT(Fast Fourier Transform)를 수행하여 주파수 영역으로 변환한다. FFT는 시간 도메인 신호를 주파수 도메인으로 변환하여, 신호의 주파수 성분을 분석할 수 있게 한다. FFT 변환을 통해 얻어진 스펙트럼의 절댓값을 계산한다.

다음으로, 주파수 범위(예: 40Hz, 80Hz, 120Hz, 180Hz, 300Hz)에 따라 각 주파수 범위 내에서 가장 강한 피크를 탐지한다. 이를 위해 로그 스펙트럼을 사용하여 피크의 강도를 계산하고, 가장 강한 피크 주파수를 선택한다. 주파수 범위는 일반적으로 인간의 청각 특성을 반영하여 선택된다. 인간의 귀는 특정 주파수 대역에서 더 민감하게 반응하므로, 이러한 대역을 중심으로 주파수 범위를 설정한다.

탐지된 피크 주파수를 기반으로 해시를 생성한다. 해시는 각 주파수 피크의 값을 특정 값으로 정규화한 후, 이를 결합하여 고유한 해시 값을 생성한다. 여기서 사용되는 정규화 값은 'FUZ_FACTOR'로 불리며, 이는 주파수 값의 정밀도를 줄여 근사치를 사용함으로써 잡음에 대한 강건성을 높이는 역할을 한다. 주파수 피크 값이 123일 때, 'FUZ_FACTOR'가 2라면, 이 값은 2의 배수인 122로 정규화된다. 이렇게 함으로써 근접한 주파수 값들을 같은 해시 값으로 묶어 잡음과 미세한 변동에 대해 강건한 해시 값을 생성할 수 있다.

피크 기반 오디오 지문 생성의 주요 장점은 계산의 효율성과 잡음 저항성이다. 이는 특히 다양한 환경에서 안정적인 오디오 인식을 가능하게 한다. 피크 기반 지문 생성은 계산 과정이 단순하며, 특정 주파수 피크의 강도를 반영하여 잡음이 많은 환경에서도 오디오 신호를 정확하게 인식할 수 있다.

Definitions:

RANGES = [40, 80, 120, 180, 300]

FUZ_FACTOR = 2

segment_length = 5 // Length of each audio segment in seconds

Function get_index(freq):

for each val in RANGES:

if freq < val:

return index of val

return last index in RANGES

Function hash_points(points):

result = 0

for each point in points reversed:

result = result * 100000 + (point - (point % FUZ_FACTOR))

return result

Function get_fingerprints(audio, sample_rate):

Define chunk_size, hop_size

Initialize peaks list

for each chunk in audio:

Calculate spectrum

Initialize points and highscores for frequency ranges

Identify peak frequencies and update points list

Calculate hash value from points and append to peaks

return peaks

Main Matching Logic:

queryBlock = get_fingerprints(query_audio, sample_rate)

bestMusic = none

minimumBER = 1.0

for each fingerprint in queryBlock:

indexNode = Lookup in database using fingerprint hash

while indexNode is not null:

Calculate BER with indexNode

Update bestMusic if BER is new minimum and within threshold

Move to next node in database chain

return bestMusic

[표 3-1] 피크 기반 오디오 지문 생성 의사 코드

제 3 절 다중 시간 세그먼트 피크 기반 오디오 지문 생성

다중 시간 세그먼트를 이용한 피크 기반 오디오 지문 생성 방법은 다양한 시간 구간에서 피크를 추출하여 오디오 신호의 다양한 시간 구간의 특성을 모두 반영하는 고유한 지문을 생성하는 것이다. 본 논문에서는 오디오 지문의 검색 정확도를 높이기 위해 여러 시간 세그먼트를 사용하는 방법을 제안한다. 이는 다양한 길이의 세그먼트를 통해 오디오의 시간적 특성을 더욱 상세하게 반영한다.

먼저, 오디오 신호를 t 초 길이의 세그먼트로 분할한다. 여기서 t 는 1초, 3초, 5초와 같이 여러 시간 길이를 포함할 수 있다. 이를 수식으로 표현하면 (a)와 같다.

$$S(t) = s(t) | t \in \{1, 3, 5\} \quad (a)$$

여기서 $S(t)$ 는 시간 t 초에 대한 오디오 세그먼트 집합이다. 각 세그먼트 $S(t)$ 에 대해 앞서 설명한 피크 기반 지문 생성 과정을 적용한다. 이러한 방식으로 생성된 지문은 데이터베이스에 저장되어, 검색 시 가장 많이 검색된 콘텐츠가 선택되는 방식이다. 이는 다양한 길이에서 오디오 신호의 특징을 추출하여, 잡음에 대한 강건성을 높이고 검색 정확도를 향상시키는데 유리하다.

다중 시간 세그먼트를 사용할 경우, 생성 시간과 데이터 크기를 고려하여 샘플링 레이트를 낮추는 것이 중요하다. 본 연구에서는 다중 시간 세그먼트 시 샘플링 레이트를 5,050Hz로 설정하여 데이터 크기를 줄이고 지문 생성 시간을 단축시켰다.

제 4 장 OTT 콘텐츠 시청률 측정

제 1 절 콘텐츠 오디오 캡처

1) 콘텐츠 오디오 캡처 개요

OTT 콘텐츠의 시청률을 측정하기 위해선 우선 현재 재생중인 콘텐츠가 무엇인지 식별하는 것이 선행되어야 한다. 본 논문에서는 대표적인 모바일 운영체제인 안드로이드에서 소프트웨어를 이용한 방법과 추가 하드웨어 장치를 이용한 방법 두가지 방법을 통하여 오디오 지문을 생성하여 식별하는 실험을 진행한다.

2) OTT 애플리케이션 동작 감지

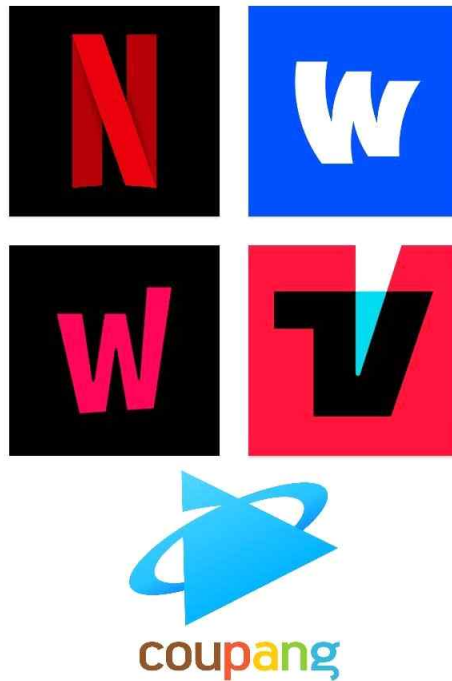
소프트웨어 기반 캡처를 진행하는 경우, 우선 OTT 서비스가 실행 중인지 확인이 필요하다. UsageStatsManager는 안드로이드 프레임워크에서 애플리케이션의 사용 통계를 수집하고 제공하는 서비스이다. 이를 통해 특정 애플리케이션이 실행 중인지, 특정 기간 동안 얼마나 자주 사용되었는지 등의 정보를 얻을 수 있다.

본 연구에서는 UsageStatsManager의 queryUsageStats를 응용하여 OTT 서비스가 실행 중인지 검사한다. queryUsageStats는 장치 사용 기록 및 통계를 가져온다. 이를 마지막 사용 시간을 기준으로 정렬하여 제일 최근 기록의 package name이 검출하려는 OTT 서비스의 패키지명과 동일한지 확인한다. UsageStatsManager를 사용하기 위해선 PACKAGE_USAGE_STATS 권한이 부여되어야 하며, 장치가 잠금 해제 상태여야 한다.

```
WHILE  
  READ UsageStats  
  
  IF UsageStats.lastTimeUsed > 0 THEN  
    IF len UsageStats > 0 THEN  
      IF package name in first UsageStats is OTT THEN  
        start capture
```

[표 4-1] OTT 프로그램 확인 의사 코드

3) 소프트웨어 기반 캡처



[그림 4-1] 캡처 프로그램 런처

안드로이드에서 오디오를 캡처하는 방법은 MIC를 이용하는 방법과 AudioPlaybackCapture를 이용한 콘텐츠의 오디오를 직접 캡처하는 방법 두 가지가 있다. AudioPlaybackCapture 기능은 Android 10부터 사용 가능하며, 다른 앱에서 재생되는 오디오를 가져오기 때문에 주변 환경의 영향이 없다는 장점이 있다. 하지만 다른 앱에서 AudioPlaybackCapture를 통해 오디오를 가져가지 못하게 allowAudioPlaybackCapture를 통해 설정할 수 있다. 따라서 우리가 캡처하려는 OTT 서비스의 설정 상태에 따라서 캡처 성공 여부가 달라진다. MIC를 통한 캡처 방법은 이러한 문제에서 좀 더 자유롭다는 장점이 있다. 하지만 MIC를 통한 캡처 방법은 주변 환경에 따라 인식률에 차이를 보이는 문제가 발생한다.

```

INIT AudioRecord
  SET AudioFormat
  SET AudioPlaybackCaptureConfig
  CALL startRecording()

SAVE Audio

CALL stop() in AudioRecord
CALL release() in AudioRecord

```

[표 4-2] 오디오 캡처 의사 코드

4) 리버스 엔지니어링을 통한 오디오 캡처

관련 연구에서 리버스 엔지니어링을 통해 시청 데이터를 수집하고 분석하여 시청 패턴을 파악하는 방법을 제안하였다. 그러나 모든 OTT 서비스에 적용하기 위해서는 각 OTT 서비스를 모두 분석하고 수정해야 하는 문제가 발생한다. 또한, 애플리케이션이 업데이트될 경우 다시 분석과 수정이 필요하다.

본 논문에서는 안드로이드 매니페스트 파일의 ‘allowAudioPlaybackCapture’ 설정에 초점을 맞춘다. 기본적으로 이 설정은 true로 되어있다. 그림 4-2와 그림 4-3은 각각 넷플릭스와 티빙의 안드로이드 매니페스트

파일을 분석한 결과이다. 분석 결과 티빙의 ‘allowAudioPlaybackCapture’의 경우 true로 되어있지만 넷플릭스의 경우 false로 되어있는 것을 확인할 수 있다.

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    android:allowAudioPlaybackCapture="false"
    android:compileSdkVersion="34"
    android:compileSdkVersionCodename="14"
    android:installLocation="auto" package="com.netflix.mediaclient"
    platformBuildVersionCode="34"
    platformBuildVersionName="14">
```

[그림 4-2] 넷플릭스 AndroidManifest

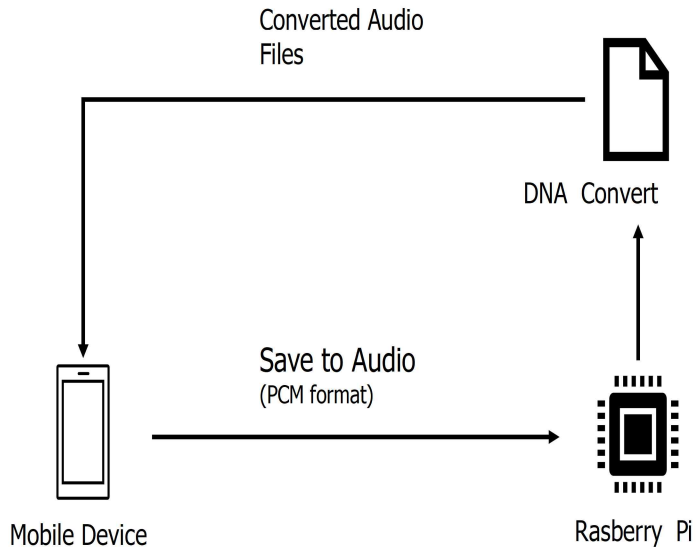
```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    android:compileSdkVersion="33"
    android:compileSdkVersionCodename="13"
    android:installLocation="auto"
    package="net.cj.cjtv.gs.tving"
    platformBuildVersionCode="33"
    platformBuildVersionName="13">
```

[그림 4-3] 티빙 AndroidManifest

본 논문에서는 실험적인 접근 방식으로 ‘allowAudioPlaybackCapture’를 false로 설정한 테스트 애플리케이션을 개발하여 실험을 진행하였다. 그런 다음, 애플리케이션을 디컴파일하여 해당 설정을 true로 변경하고 애플리케이션을 리패키징한 후 실험을 수행하였다. 이 과정을 통해 오디오 캡처 관련 설정을 조정할 수 있는 유연성과 가능성을 확인할 수 있었다.

5) 하드웨어 기반 캡처

하드웨어 기반 캡처는 단일 보드 컴퓨터(SBC)를 기반으로 하여 다양한 환경에서 실행되는 OTT 플랫폼과의 호환성을 보장한다. 안드로이드 단일 사용으로는 콘텐츠 오디오를 가져오기 위해선 AudioPlaybackCapture와 같은 방법을 사용해야 하지만, SBC는 블루투스를 통해 원본 오디오를 가져오며, 해당 오디오를 저장 및 가공할 수 있다. 이 방법은 추가 하드웨어가 필요하지만, 기존 시청률 조사 방법에서도 하드웨어가 사용되었던 것을 고려하면 크게 문제가 되진 않을 것이다.



[그림 4-4] SBC를 이용한 캡처 방식

제 2 절 오디오 지문 생성 및 식별

1) 오디오 지문 생성

3장에서 소개한 오디오 지문 생성 알고리즘은 모바일 환경에서 실시간으로 처리되기 때문에, 속도 측면에서 C/C++로 구현되었다. 이를 JNI를 통해 안드로이드에서 구동한다.

사용자가 OTT 서비스를 이용하면 오디오 데이터를 주기적으로 캡처한다. 이때, 하드웨어의 발전으로 마이크가 높은 품질의 오디오 데이터를 포착할 수 있게 되었음에도 불구하고, 데이터 처리 시간과 효율성을 고려하여 샘플 속도와 채널 수를 낮게 설정한다. 이는 데이터의 크기를 줄이고, 실시간 처리의 부담을 경감시키는 동시에 필요한 정보를 유지하기 위한 전략이다. 다운샘플링된 오디오 데이터는 이후 오디오 지문 기술로 처리된다.

```
[3, 0, 8, 0, 8, 0, 10, 0, 12, 0, 8, 0, 7, 0, 3, 0, -1, -1, 6, 0, 4, 0, 10, -1, -16, -1, -16, -1, -17, -1, -19, -1, -25, -1, -25, -1, -26, -1, -24, -26, -1, -26, -1, -27, -1, -27, -1, -28, -1, -22, -1, -26, -1, -26, -1, -5, 0, 10, 0, 11, 0, 10, 0, 6, 0, 3, 0, 2, 0, 2, 0, 0, 0, -3, -1, 5, 0, 6, 43, 0, 46, 0, 42, 0, 46, 0, 47, 0, 46, 0, 47, 0, 49, 0, 50, 0, 44, 0, 41, -1, -7, -1, -5, -1, -9, -1, 3, 0, 8, 0, 5, 0, 1, 0, 2, 0, 2, 0, -2, -1, -2, -10, -1, -2, -1, -2, -1, -8, -1, -12, -1, -9, -1, -12, -1, -10, -1, -1, -14, -1, -17, -1, -10, -1, -13, -1, -14, -1, -12, -1, -19, -1, -14, -14, -1, -10, -1, -3, -1, -4, -1, 3, 0, 3, 0, 5, 0, 1, 0, -2, -1, 2, 0, 1, 0, 35, 0, 37, 0, 41, 0, 46, 0, 50, 0, 47, 0, 45, 0, 37, 0, 35, 0, 38, 0, 0, 8, 0, 1, 0, 5, 0, 2, 0, 4, -1, -15, -1, -22, -1, -15, -1, -22, -1, -2, -1, -18, -1, -17, -1, -10, -1, -12, -1, -9, -1, -6, -1, -4, -1, -10, -1, -34, -1, -38, -1, -46, -1, -51, -1, -47, -1, -47, -1, -49, -1, -43, -1, -1, -11, -1, -10, -1, -12, -1, -13, -1, -9, -1, -4, -1, 1, 0, 0, 0, 7, 0, -22, -1, -21, -1, -25, -1, -18, -1, -11, -1, -12, -1, -4, -1, 3, 0, 1, 0, 0, 34, 0, 35, 0, 36, 0, 36, 0, 26, 0, 18, 0, 14, 0, 11, 0, 8, 0, 10, 0, 8, 0, 20, 0, 17, 0, 22, 0, 23, 0, 19, 0, 20, 0, 22, 0, 18, 0, 18, 0, 23, 0, 0, 11, 0, 16, 0, 3, 0, -4, -1, 0, 0, 1, 0, 5, 0, -8, -1, -6, -1, -6, -1, -1, -12, -1, -13, -1, -14, -1, -18, -1, -11, -1, -15, -1, -19, -1, -11, -1, -12, -1, -19, -1, -20, -1, -12, -1, -9, -1, -6, -1, -12, -1, -3, -1, 18, 0, 20, 0, 19, 0, 14, 0, 13, 0, 14, 0, 7, 0, 0, 0, -4, -1, -4, -1, -5, 8, 0, 5, 0, 6, 0, 3, 0, 4, 0, 4, 0, 6, 0, 7, 0, 6, 0, 6, 0, 14, 0, 10, 0, 12, 0, 12, 0, 8, 0, -2, -1, 0, 0, -2, -1, -7, -1, -11, -1, -17, -1, -18, -12, -1, -6, -1, -6, -1, -6, -1, 5, 0, 5, 0, 7, 0, 3, 0, -3, -1, 5, 0, 4, -4, -1, -2, -1, 2, 0, 7, 0, 1, 0, -2, -1, -5, -1, -1, -1, -4, -1, -8, -1, -6, -1, -7, -1, -5, -1, 2, 0, 9, 0, 9, 0, 13, 0, 15, 0, 23, 0, 23, 0, 18, 7, 0, 9, 0, 3, 0, -1, -1, 2, 0, 1, 0, -4, -1, 6, 0, 9, 0, 10, 0, 8, 0, 4, -1, -15, -1, -18, -1, -18, -1, -18, -1, -23, -1, -28, -1, -32, -1, -33, -25, -1, -14, -1, -11, -1, -6, -1, -3, -1, -4, -1, -7, -1, 1, 0, -2, -1, -20, -1, -15, -1, -14, -1, -16, -1, -16, -1, -13, -1, -19, -1, -15, -1, -14, 0, 17, 0, 19, 0, 24, 0, 24, 0, 25, 0, 29, 0, 28, 0, 32, 0, 37, 0, 36, 8, 0, 11, 0, 12, 0, 17, 0, 18, 0, 22, 0, 30, 0, 25, 0, 19, 0, 27, 0, 21,
```

[그림 4-5] 원본 PCM 데이터

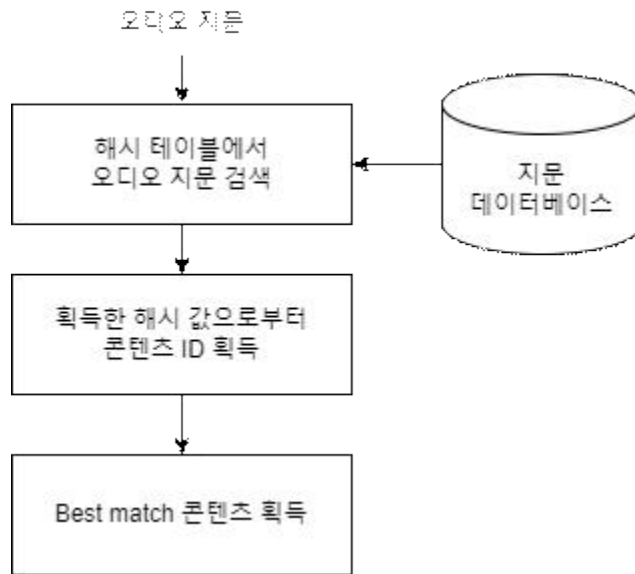
오디오 데이터는 PCM 형태로, 22KHz, MONO, 16bit의 특성을 가진 약 5초 분량의 데이터를 사용한다. 이 PCM 데이터는 약 110,000의 크기를 가지며, 지문 추출의 과정에서 1,285의 데이터로 변환된다. 추출된 지문 데이터는 Base64 인코딩을 거쳐 서버로 전송된다.

```
P+q5nue3tCJJ7KSn7YaEOu2FqT3sq7057JK1ceyktjIuIblu6Pro48/W+yTqys/P1LroJXtqKTSUkHHe  
yUmStjYT90auOBkuuQij8d7Jym7LG0YT8/6aetPz8dHmY/aT8/P+upqDQI65yY7LqzPz/InIjInjsqgY  
2I3HsnL7iYox7KuCP+2cmOyJtz8p7I6I76SxCOyztT9o7YKA64Krl1vsjJfsvLRl7Ii+R+eStWTrhYE0J  
1nsjYzrrYRlPynjip4IKuuWnCcJp3IyJT8tPz8vNjPspqIS7YOWVVOuHIA==
```

[그림 4-6] Base64 인코딩

2) 콘텐츠 식별

서버로 전송된 데이터는 Base64 디코딩을 통해 지문으로 변환 후 이를 데이터베이스를 통해 현재 콘텐츠를 식별한다. 이를 위해 데이터베이스에는 미리 정의된 다양한 콘텐츠의 오디오 지문이 저장되어 있어야 한다. 그림 4-7은 지문을 데이터베이스와 비교하여 콘텐츠를 식별하는 과정을 나타낸다.



[그림 4-7] 오디오 지문 검색 순서도

제 5 장 실험결과 및 성능평가

제 1 절 전체 실험환경

본 논문에서 설명한 피크 기반 오디오 지문 시스템과 다중 시간 세그먼트 피크 기반 오디오 지문 시스템의 검색결과 측정을 위해 퍼블릭 도메인 영화 파일을 획득하여 해시 테이블 데이터 베이스로 이용하였고, 해당 영화 파일의 무작위 위치를 캡처한 오디오 데이터를 피크 기반 오디오 지문과 다중 시간 세그먼트 피크 기반 오디오 지문 검색에 사용하였다.

쿼리는 5초의 길이를 사용하였으며, 다중 시간 세그먼트 피크의 경우 (1, 3, 5)초를 사용하였다. MIC의 노이즈를 가정하여 클린 데이터에 5dB 과 10dB의 노이즈를 강제로 추가한 데이터와 실제 안드로이드에서 캡처된 콘텐츠 오디오 데이터를 사용하여 실제 환경에서 발생할 수 있는 다양한 수준의 잡음에 대해 얼마나 견고한지를 테스트하였다.

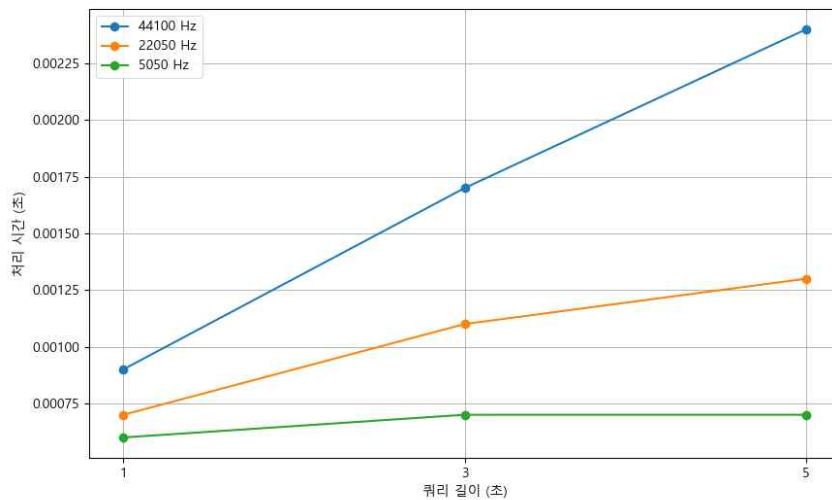
제 2 절 성능평가

본 논문에서 설명한 피크 기반 오디오 지문 시스템과 다중 시간 세그먼트 피크 기반 오디오 지문 시스템의 결과를 측정하였다. 표 5-1은 검색 정확도를 기록한 표이다.

	22Mhz 5초	5Mhz (1, 3, 5)초	22Mhz (1, 3, 5)초
clean	70%	70%	50%
noise 5dB	30%	60%	20%
noise 10dB	10%	60%	40%
capture	60%	60%	40%

[표 5-1] 지문 검색 정확도

실험결과 clean 및 안드로이드에서 캡처된 오디오의 경우 22Mh 샘플 레이팅의 5초 단일 쿼리와 5Mhz 샘플레이팅 다중 세그먼트의 결과가 같았다. 그러나 강제로 노이즈를 5dB, 10dB를 주었을 경우, 다중 세그먼트의 성능이 더 뛰어난 것이 확인 할 수 있다. 추가로 다중 세그먼트의 샘플링 레이트를 22Mhz로 테스트 했을 경우 오히려 정확도가 떨어지는 결과를 얻었다.



[그림 5-1] 샘플링 레이트별 지문 생성 시간

실험결과 샘플링 레이트를 줄이면서 다중 시간 세그먼트 기법을 사용하면, 단일 5초 쿼리를 사용한 것과 지문 생성 시간이 비슷하며, 실제 환경에서 발생하는 노이즈와 같은 문제에서 성능이 더 좋은 것이 확인되었다.

제 6 장 결론

본 논문에서는 모바일 환경에서 OTT 콘텐츠의 시청률 측정을 위한 방법을 제시하였다. 특히, 다중 시간 세그먼트 피크 오디오 지문 시스템을 이용하여 콘텐츠를 식별하고 시청률을 측정하는 새로운 접근 방식을 개발하고, 이를 실제 환경에서 실험함으로써 그 가능성을 입증하였다. 본 연구는 OTT 서비스의 다양한 콘텐츠와 사용자의 소비 패턴을 정확하게 파악할 수 있는 기술을 개발함으로써, 미디어 산업에 이바지한다.

본 연구의 핵심은 오디오 데이터의 캡처 및 오디오 지문 기술을 활용하여, 다양한 콘텐츠를 식별하는 것이다. 이를 통해 콘텐츠 제작자와 배급자는 시청률 데이터를 기반으로 보다 정확한 시장 분석과 전략 수립이 가능하게 되었다. 또한, 이 기술은 사용자의 시청 습관을 파악하고, 타깃 오디언스에게 보다 맞춤형 콘텐츠를 제공하는데도 중요한 역할을 할 것으로 기대된다.

추가로, 본 연구는 향후 연구 방향에 대해 몇 가지 제안을 한다. 첫째, 다양한 환경에서의 오디오 캡처 성능을 더욱 향상시키기 위한 알고리즘의 개선이 필요하다. 둘째, 딥러닝 기반의 오디오 지문 인식 기술을 도입하여, 잡음에 더욱 강인한 시스템을 개발할 필요가 있다. 셋째, 실시간으로 대규모 데이터를 처리할 수 있는 분산 처리 시스템의 도입을 고려해야 한다. 넷째, 영상 데이터와 함께 처리하여 식별률을 높일 수 있도록 개발할 필요가 있다. 이러한 연구 방향은 본 논문의 연구를 더욱 발전시키고, 실용적인 응용 가능성을 높일 것이다.

참 고 문 헌

1. 국내문헌

최선영, 김민수, 김명준. (2014). 스마트폰에서의 OTT(Over The Top) 서비스 시청패턴 추적 어플리케이션 설계: 티빙(tving)을 중심으로. 『한국정보통신학회논문지』, 18(4), 1,000-1,006.

박지현, 윤영석, 김정현, 유원영. (2014). 『모바일 기기에서의 방송프로그램 실시간 식별을 위한 오디오 핑거프린팅』 한국통신학회: 학술대회논문지

한경복, 조혜승, 김형국. (2014). 『오디오 핑거프린트 검색성능 향상을 위한 피크쌍 타겟존 설정방식』 한국통신학회: 학술대회논문지

서진수, 김정현, 김혜미. (2019). 파워 가중치를 이용한 오디오 핑거프린트 정합 『한국음향학회지』, 38(6), 716-723.

류상현, 김형국. (2014). TV 광고 식별을 위한 Constant-Q 변환 기반의 오디오 핑거프린팅 방식. 『한국음향학회지』, 33(3), 210-215.

박만수, 김희린. (2006). 실제 잡음 환경에 강인한 오디오 핑거프린팅 기법. 『Telecommunications Review』, 16(3), 435-446.

한경복, 김형국. (2014). 『강인한 오디오 핑거프린트 피크추출을 위한 역방향 문턱값 적용방식』 대한전자공학회: 학술대회

서진수. (2013). 경첩 손실 함수 최소화를 통한 오디오 핑거프린트 이진화. 『한국음향학회지』, 32(5), 415–422.

2. 국외문헌

Anguera, X., Garzon, A., Adamek, T. (2012). MASK: Robust local features for audio fingerprinting. In Proc. of IEEE International Multimedia and Expo, 455–460.

Wang, A. (2003). An Industrial-Strength Audio Search Algorithm. In Proceedings of ISMIR, 7–13.

Baluja, S., Covell, M. (2007). Audio Fingerprinting: Combining Computer Vision & Data Stream Processing. In Proceedings of IEEE ICASSP, 2, 213–216.

Mihcak, M. K., Venkatesan, R. (2001). A perceptual audio hashing algorithm: A tool for robust audio identification and information hiding. LNCS, 2137, 51–65.

Shen, L. (2009). Fast audio fingerprint search strategy for song identification. In Proc. of International Networking and Digital Society, 2, 259–262. Guiyang, China.

Wang, A. (2006). The Shazam Music Recognition Service. Com. ACM, 49(8), 44–48.

- Hartung, F., Kutter, M. (1999). Multimedia watermarking techniques. Proceedings of the IEEE, Special Issue on Identification and Protection of Multimedia Information, 87(7), 1079–1107.
- Haitsma, J., Kalker, A. (2002). A highly robust audio fingerprinting system. In Proceedings of the International Symposium on Music Information Retrieval (ISMIR).
- Cano, P., Batlle, E., Kalker, T., Haitsma, J. (2002). A review of algorithms for audio fingerprinting. In Proceedings of the International Workshop on Multimedia Signal Processing.

ABSTRACT

Content Identification Techniques for Measuring OTT Viewership in Mobile Environments

Kim, Seong-Jin

Major in Computer Engineering

Dept. of Computer Engineering

The Graduate School

Hansung University

The proliferation of mobile devices such as smartphones and tablets, along with changes in media content consumption patterns, has led to the extensive production and distribution of OTT content. In this context, there is a growing need for a new method to accurately measure the quality and popularity of content. This paper proposes a method for measuring OTT viewership by capturing the audio of the content and identifying it through audio fingerprinting. This study aims to contribute to the media industry by providing a new tool that accurately identifies various OTT content and consumption patterns. Content creators and distributors can utilize viewership data to perform more precise market analysis and strategic planning. Additionally, this technology can play a significant role in understanding viewer habits and offering more

personalized content to targeted audiences. Traditional viewership measurement methods have been mainly limited to conventional television broadcasts and have not adequately reflected the viewing patterns of OTT content consumed on various devices. However, this study collects audio data from various mobile devices such as smartphones and tablets and uses it to measure real-time viewership. This allows for effective monitoring of content provided on various OTT platforms and real-time collection of viewership data. This technical approach not only measures viewership but also provides valuable insights to content creators and distributors. With the advancement of OTT services, the proposed viewership measurement method will contribute to enhancing the competitiveness of the media industry.

【KEYWORD】 OTT, Android, Content Identification, Audio Fingerprinting, Media Industry