



#WebRTC 중계 리턴 시스템  
#기술의 특징 #적용 사례

2025 KBS 미디어창의기술전 수상작 소개

# WebRTC를 활용한 중계 리턴 시스템

글. 임준하 KBS 제작기술국 제작기술1

## 시작하며

이번 'KBS 미디어창의기술전'에 'WebRTC를 활용한 중계 리턴 시스템'이라는 주제로 작품을 출품했습니다. WebRTC라는 기술을 활용해 중계 현장에서 RF 카메라 시스템 구성 업무를 수행하며 지속해서 불편하다고 느껴왔던 문제를 해결했다는 점만으로도 개인적으로 의미 있는 작업이었는데 그 결과가 '미디어창의기술전'에서 노력상으로 이어져 더욱 뜻깊은 경험으로 남았습니다.

WebRTC는 웹 브라우저 간에 영상과 음성 데이터를 실시간으로 주고받을 수 있도록 설계된 통신 기술로 별도의 전용 프로그램 설치 없이도 낮은 지연의 영상 전송이 가능하다는 특징을 가집니다. 본 글에서는 이러한 'WebRTC 기술을 중계 리턴 시스템'에 어떻게 적용했는지 그리고 그 과정에서 기존 중계 환경의 어떤 문제를 해결할 수 있었는지를 살펴보고자 합니다.

## 리턴 시스템의 중요성

중계방송 제작에서 카메라 운용자는 리턴 영상을 통해 중계차 PGM을 실시간으로 확인할 수 있습니다. 이 리턴을 통해 카메라 감독은 커팅 시점과 촬영 동선을 판단합니다. 이 과정에서 리턴 시스템이 정상적으로 동작하지 않거나 지연이 발생할 경우 촬영 타이밍을 놓치거나 의도하지 않은 화면이 송출되는 등 방송 사고로 이어질 가능성이 있습니다. 그만큼 리턴은 중계 제작 시스템에서 기본적이면서도 중요한 요소라 할 수 있습니다.



그림 1. 기존 리턴 방식의 스탠드 카메라 & EFP 카메라 뷰파인더

## 중계 환경의 변화

최근 중계 현장의 환경은 과거와 크게 달라졌습니다. 기존의 스탠다드, EFP 카메라 중심의 제작 방식에서 벗어나 RF 카메라, 드론, MNG 카메라, 초고속 카메라 등 다양한 외부 카메라의 활용 빈도가 꾸준히 증가하고 있습니다. 이에 따라 중계 내부 시스템을 직접 거치지 않는 카메라들이 늘어났고 리턴을 요청하는 장비의 수 역시 함께 증가했습니다. 그러나 기존 리턴 시스템은 이러한 외부 카메라 환경을 전제로 설계된 구조가 아니었기 때문에 동일한 방식으로 적용하기에는 여러 가지 한계가 있었습니다.



그림 2. RF 카메라와 초고속 카메라

## 기존 시스템의 한계

기존의 경우 RF 카메라에 리턴 시스템을 적용하기 위해서는 여러 장비를 추가로 구성해야 했습니다. 무선 비디오 송수신 장치[그림 3]를 설치한 뒤 중계차에서 출력된 리턴 영상을 비디오 케이블로 연결하고 이를 확인하기 위한 전용 모니터와 전원을 공급할 배터리[그림 4]를 짐벌에 함께 장착하는 방식이 일반적이었습니다. 이 과정은 단순히 리턴 화면을 확인하기 위한 구성임에도 불구하고 시스템 대비 상당한 장비와 준비 시간을 요구합니다. 결과적으로 리턴 회선을 하나 구성하기 위해 많은 노동량이 필요했고 현장 상황에 따라서는 설치 자체가 부담되는 경우도 발생했습니다.

특히 짐벌에 모니터와 배터리를 추가로 설치하는 구조는 카메라 운용자에게 물리적인 부담으로 작용했습니다. 이러한 이유로 현장에서는 리턴 모니터 설치를 최소화하거나 아예 리턴을 보지 않고 촬영을 진행하는 경우도 적지 않았습니다. 이 경우 카메라 감독은 인터컴을 통한 PD의 콜에만 의존해 중계 제작을 하게 되는데 이는 PGM 화면을 직접 확인할 수 없는 한계를 동반합니다. 그 결과로 커팅 타이밍을 놓친다거나 다음 동선을 제때 파악하지 못하는 상황이 발생했습니다. 이러한 문제를 해결하기 위한 대안으로 스마트폰 하나만으로도 리턴 영상을 볼 수 있는 'WebRTC를 활용한 중계 리턴 시스템'을 개발하게 되었습니다.

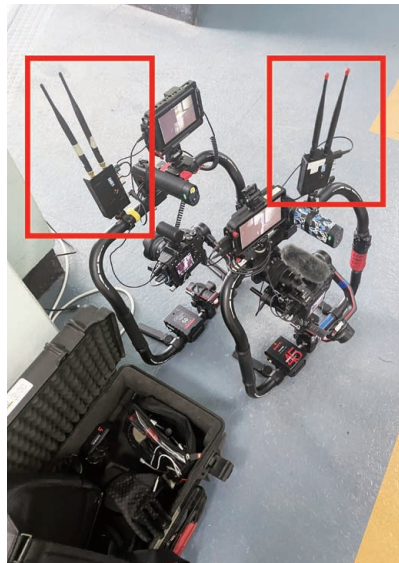


그림 3. 무선 송수신 장비 설치

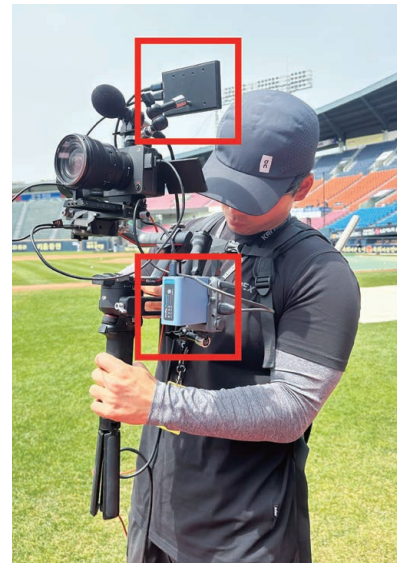


그림 4. 모니터 및 배터리 설치

## WebRTC 기술의 개요와 특징

WebRTC는 Real Time Communication의 약어로 구글이 주도해 개발한 오픈소스로 브라우저끼리 영상·음성·데이터를 직접 주고받아 지연을 최소화하며 별도 프로그램 설치 없이 웹 브라우저만으로 실시간 화상 연결 시스템을 구축할 수 있습니다. 이 기술을 사용해 개발된 대표적인 프로그램은 화상회의 솔루션 프로그램인 ZOOM, 게임 커뮤니티 기반 실시간 음성·영상·채팅 플랫폼은 Discord 등이 있습니다.



그림 5. WebRTC를 활용한 프로그램 ZOOM과 Discord

WebRTC의 대표적인 장점은 서버를 통해 영상을 업로드하고 다시 내려받는 기존 스트리밍 방식과 달리 단말 간 직접 연결을 기본 구조로 한다는 점입니다. 이러한 구조는 영상 전송 과정에서 발생하는 지연을 최소화할 수 있어 실시간성이 중요한 중계방송 현장에서 적합합니다.

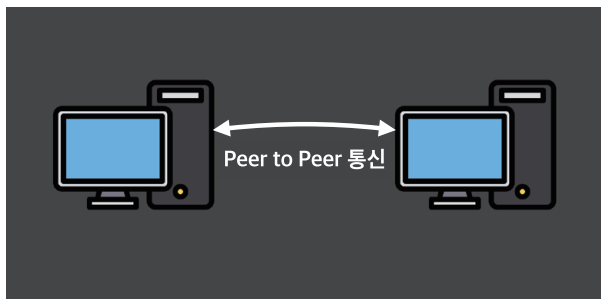


그림 6. P2P 통신 개요도

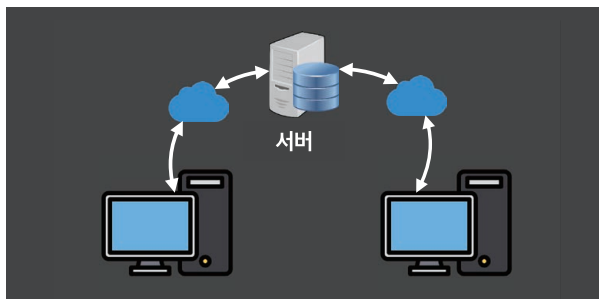


그림 7. 스트리밍 방식 개요도

중계 현장에서는 실시간이 중요한 환경이기 때문에 아무리 안정적인 영상이라 하더라도 지연이 크게 발생하면 리턴으로서의 의미는 떨어집니다. WebRTC는 이러한 요구에 만족하는 기술로 네트워크 환경이 허용하는 범위 내에서 매우 낮은 지연으로 영상을 전달할 수 있습니다. 또한 웹 브라우저 기반으로 동작하기 때문에 특정 운영체제나 장비에 종속되지 않고 스마트폰, 태블릿, 노트북 등 다양한 단말에서 동일한 방식으로 접근할 수 있다는 점도 중계 현장에 적용하기에 적합한 조건으로 작용했습니다.

## 프로그래밍 개발

본 시스템은 Node.js와 HTML, CSS, JavaScript를 활용하여 구현하였습니다. 전체 구조는 웹 기반으로 설계하여 별도의 전용 프로그램 설치 없이 브라우저만 실행하면 송신과 수신이 모두 가능하도록 중점을 두었습니다. 이를 통해 운영체제나 단말 종류와 관계없이 동일한 환경에서 리턴 시스템을 사용할 수 있도록 구성하였습니다.



그림 8. Node.js & HTML, JavaScript, CSS

## 서버 구성

서버 측은 Node.js 환경에서 Express 프레임워크를 사용해 웹 서버를 구축하였으며 그 위에 Socket.io를 적용해 WebRTC 연결에 필수적인 시그널링 서버 기능을 구현하였습니다. WebRTC는 단말 간 직접 연결을 기본으로 하지만 연결을 시작하기 위해서는 서로의 접속 정보와 미디어 정보를 교환하는 과정이 필요합니다. 이 역할을 시그널링 서버가 담당하며 본 시스템에서는 Socket.io를 통해 Offer와 Answer SDP를 교환하고 ICE 후보 정보를 전달하는 구조로 구성하였습니다.

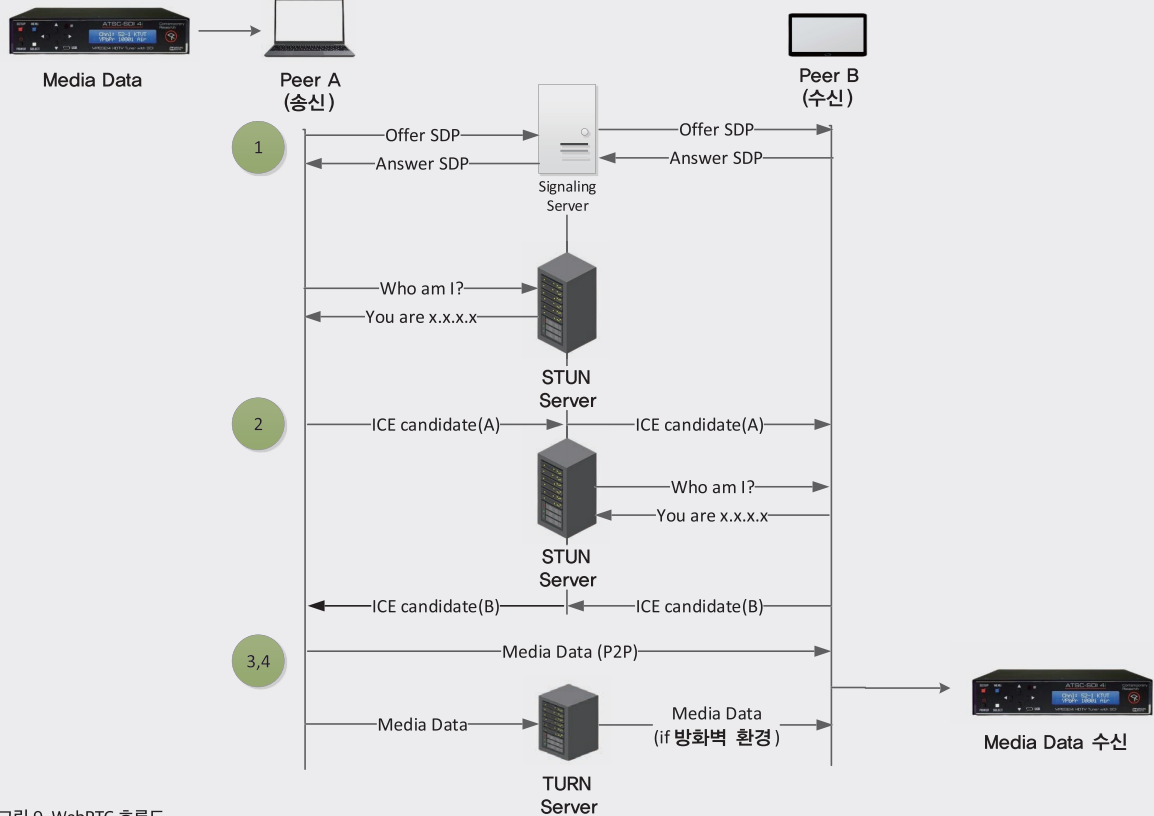


그림 9. WebRTC 흐름도

[그림 9]의 구성도에서와 같이,

- 1 송신 단말과 수신 단말은 먼저 시그널링 서버를 통해 SDP 정보를 교환하며 연결을 준비합니다.
- 2 이후 각 단말은 STUN 서버를 통해 자신의 공인 IP와 네트워크 정보를 확인하고 이를 ICE Candidate 형태로 서로에게 전달합니다.
- 3 양 단말 간 직접 통신이 가능한 경로를 탐색하게 되며 연결이 성립되면 미디어 데이터는 서버를 거치지 않고 P2P 방식으로 직접 전송됩니다. 그렇지 않고 방화벽이나 네트워크 환경으로 인해 직접 연결이 어려운 경우에는 TURN 서버를 통해 미디어 데이터가 중계되도록 구성하여 다양한 네트워크 환경에서도 안정적인 리턴 영상 수신이 가능하게 합니다.

## 송수신 페이지 구현

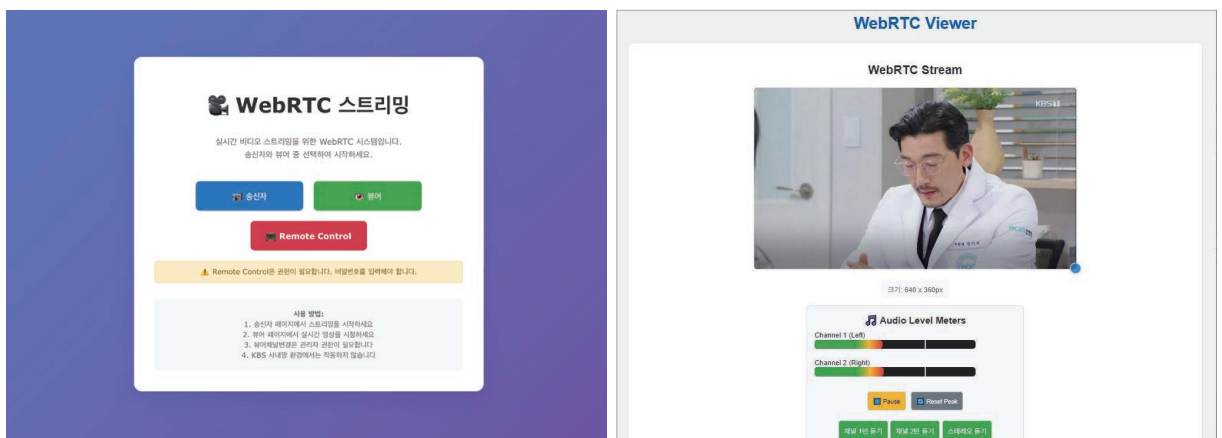


그림 10. 웹브라우저 화면(메인 페이지, 수신 페이지)

클라이언트 측은 HTML과 CSS, JavaScript를 사용해 송신 페이지와 수신 페이지를 각각 구현하였습니다. 송신 페이지에서는 중계차에서 입력된 PGM 영상을 WebRTC 스트림으로 송출하고 수신 페이지에서는 브라우저를 통해 해당 영상을 실시간으로 수신하는 방식입니다. 사용자는 단순히 웹 페이지에 접속하는 것만으로 리턴 화면을 확인할 수 있으며 이로 인해 현장에서의 접근성과 운용 편의성이 크게 향상되었습니다. 이러한 과정을 통해 [그림 11]과 같이 사진과 같이 스마트폰을 통해 간단하게 중계 리턴 영상을 실시간으로 확인할 수 있는 시스템을 구현할 수 있었습니다.



그림 11. 스마트폰으로 중계차 영상 시청

## 적용 사례

### 프로야구

프로야구 중계 현장에서 RF 카메라 감독들은 이닝 종료, 홈런 상황에서 그라운드 위를 끊임없이 이동하며 상황에 따라 빠르게 대응해야 합니다. 이러한 환경에서는 짐벌의 무게가 곧 기동성과 직결되며, 장비가 무거워질수록 움직임에 제약이 발생하게 됩니다. 특히 장시간 촬영이 이어질 경우 무게 부담은 촬영 정확도와 집중도에도 영향을 미칠 수밖에 없습니다. ‘WebRTC를 활용한 중계 리턴 시스템’을 적용함으로써 스마트폰 하나로 중계차 PGM 신호를 확인할 수 있는 환경을 구축하였고, 이에 따라 기존에 필요했던 리턴 모니터와 배터리를 제거할 수 있었습니다. 그 결과 짐벌 구성은 크게 경량화되었으며, 카메라 감독들은 자유로운 움직임 속에서 방송 화면을 직접 확인하며 중계 제작에 집중할 수 있게 되었습니다. 이는 현장 운용의 부담을 줄이는 동시에 중계 화면의 완성도를 유지하는 데에도 긍정적인 효과로 이어졌습니다.



그림 12. 프로야구 중계방송 제작

### 광복 80주년 경축 행사

광화문에서 열린 광복 80주년 경축 행사에서도 ‘WebRTC를 활용한 중계 리턴 시스템’을 적용하였습니다. 해당 행사는 무대 위와 무대 밖의 동선이 복잡하게 구성되어 있었으며 대통령의 이동 동선까지 고려해야 하는 특성상 RF 카메라는 여러 지점을 오가며 중계방송을 제작해야 했습니다. 이로 인해 촬영 위치가 수시로 변경되었고 리턴 화면을 안정적으로 확인할 수 있는 환경을 확보하는 것이 중요한 과제로 작용했습니다.

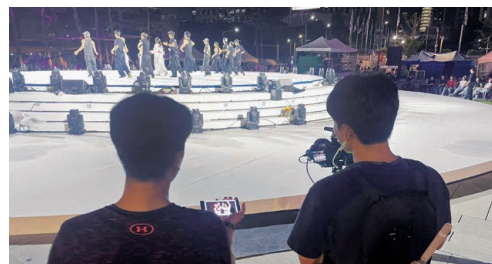


그림 13. 광복 80주년 경축 행사 중계방송 제작

기존 방식대로라면 리턴 모니터를 설치하기 어렵고 설치하더라도 긴 전파 거리와 다수의 전파 간섭으로 인해 리턴 신호를 안정적으로 수신할 수 있을지 확인할 수 없었습니다. 그러나 LTE 통신을 활용한 'WebRTC 기반 리턴 시스템'은 무대 안팎을 포함한 행사장 전반에서 안정적으로 동작하였고 RF 카메라 감독은 이동 중에도 리턴 영상을 지속해서 확인할 수 있었습니다. 그 결과 동선이 복잡하고 작업 난도가 높은 행사였음에도 불구하고, 안정적인 중계방송 제작 환경을 유지하며 행사를 성공적으로 마무리할 수 있었습니다.

### 🔍 슈퍼레이스

대한민국을 대표하는 모터스포츠 대회인 슈퍼레이스 중계방송 제작 현장에서는 무엇보다 촬영 인력의 안전 확보가 중요한 과제로 작용합니다. 특히 RF 카메라는 피트 스탑 구간에서 차량과 스태프가 밀집된 상황을 근접 촬영해야 하므로 중계 흐름을 정확히 인지한 상태에서의 신속한 판단이 필수적입니다.

그러나 기존 중계 환경에서는 RF 카메라 감독이 리턴 영상으로 유튜브 송출 화면에 의존하는 경우가 많았고, 이로 인해 약 10초 이상의 지연이 발생하여 실제 방송 진행 상황을 즉각적으로 파악하는 데 한계가 있었습니다. 이러한 지연은 촬영 타이밍 판단과 이동 동선 결정에 영향을 주어 중계 방송 제작 중 안전 측면에서 취약하였습니다.

하지만 'WebRTC 기반 중계 리턴 시스템'을 도입한 이후에는 RF 카메라 감독이 실시간 PGM 화면을 직접 확인하며 촬영을 수행할 수 있게 되었습니다. 이를 통해 차량 진입 시점과 방송 흐름을 사전에 인지할 수 있었고 불필요한 이동을 최소화한 상태에서 안전한 위치를 확보하며 촬영을 진행할 수 있었습니다. 그 결과 중계방송 제작 현장의 안전성을 크게 높이는 동시에 현장감 있는 화면을 안정적으로 제공할 수 있는 제작 환경을 구축할 수 있었습니다.



그림 14. 슈퍼레이스 중계방송 제작

### 마치며

이번 'WebRTC를 활용한 중계 리턴 시스템'은 거창한 목표보다는 중계 현장에서 반복적으로 느껴왔던 불편함을 조금이라도 줄여보자는 생각에서 출발한 작업이었습니다. 시스템을 구성하는 것과 별개로 '미디어창의기술전'에 출품하는 과정이 쉽지만은 않았고 바쁜 현업 일정 속에서 번거롭게 느껴졌던 순간도 있었습니다. 그럼에도 불구하고 출품을 통해 선·후배들과 기술적인 의견을 나누고 다양한 피드백을 받을 수 있었던 경험은 예상보다 큰 의미로 다가왔습니다.

현업에서 느낀 불편함을 WebRTC라는 기술을 통해 개선하고 그 시도가 노력상 수상으로까지 이어졌다는 점에서 이번 '미디어창의기술전'은 저에게 뜻깊었고 현장에서 마주하는 불편함을 당연하게 받아들이지 않고 기술적 관점에서 해결 가능성을 고민하며 한 단계씩 발전해 나가는 자세야말로 엔지니어에게 필요한 덕목이라고 다시 한번 느꼈습니다. 📌