실시간 백엔드 딥러닝 영상분석 서비스 프레임워크

이정훈*, 이승수*, 정영빈*, 황광일*,**,1)
*인천대학교 임베디드시스템공학과, **인천대학교 IoT and Big-data 연구센터
e-mail : junghoon450@gmail.com

Real-Time Back-end Deep Learning Video Service Framework

Jeong-Hun Lee*, Seung-Su Lee*, Yeong-Bin Jeong*, Kwang-il Hwang*,**,1)
*Dept. of Embedded Systems Engineering, Incheon National University
**IoT and Big-data Research Center, Incheon National University

요 약

최근 딥러닝의 급격한 발전에 따라, 다양한 영상처리에 기반한 새로운 어플리케이션과 서비스가 개발되어지고 있다. 이러한 다양한 서비스들이 동작되는 플랫폼이 다양하고, 리소스에 제약이 많은 경우가 있어 일반적으로 이러한 딥러닝 기반 영상처리 서비스에서는 클라우드를 이용한 백엔드 서비스를 이용하는 경우가 많다. 그러나, 알고리즘 개발단계에서는 클라우드 백엔드 서비스가 적절하지만, 실제로 제품 및 서비스의 필드 적용에 있어서는 클라우드 사용에 따른 제약이 상당한 걸림돌로 작용한다. 그리하여 본 논문에서는 다양한 디바이스의 영상 데이터를 로컬 또는 기관에 있는 충분한 자원 활용이 가능한 서버로 전송하여 프로세싱 후 실시간 결과를 feedback받을 수 있는 효율적인 back-end 딥러닝 영상분석 서비스 프레임워크를 제안한다.

1. 서론

최근 영상처리 기술의 발전에 힘입어 CAM. CCTV 또는 다양한 영상을 활용한 어플리케이션 또는 서비스들이 폭넓게 개발되고 있다. 특히, 딥러닝의 눈부신 발전은 영상처리 분야에서 그동안 시도해 보지 못했던 다양한 문제에 대한 도전을 가능하게 만들었다. 하지만, 이러한 딥러닝 기술을 보다 적극적으로 활용하기 위해서는 멀티 GPU와 보다 많은 메모리 및 시스템 파워를 가지는 딥러닝 워크스테이션 또는 전용의 서버급의 시스템이 필요하다.

그리하여, 효율적인 어플리케이션 실행과 서비스 구현을 위해서는 실제 어플리케이션이 실행되는 타겟 시스템(클라이언트)에서 데이터만을 전송하고 원격명령에 의해 서버에서 고사양의 프로세싱 능력을 통해 데이터를 분석 후 결과를 다시 타겟시스템으로 피드백해주는 백엔드 딥러닝 서비스 (Back-End Deep Learning Service)가 요구된다.

대체로 이러한 Back-end 서비스는 클라우드 업체에서 제공하는 기술에 의존하여 사용되고 있다. 특히, Amazon Rekognition [1], Azure Stream analytics [2], Google Video Intelligence API [3] 등은 가장 대표적인 백엔드 영상분석 서비스 이다. 클라우드 기반의 백엔드 서비스의 경우 영상처리에 대한 검증된 결과를 사용할 수 있으며, 추가적으로 다른 서비스 또는 Storage 들과 연동하여 확장된 통합 서비스를 구축할 수 있는 장점이 있다.

그럼에도 불구하고, 클라우드 기반의 백엔드 서비스의 실시간 영상처리로의 활용은 다음과 같은 문제점 또는 제약이 있다.

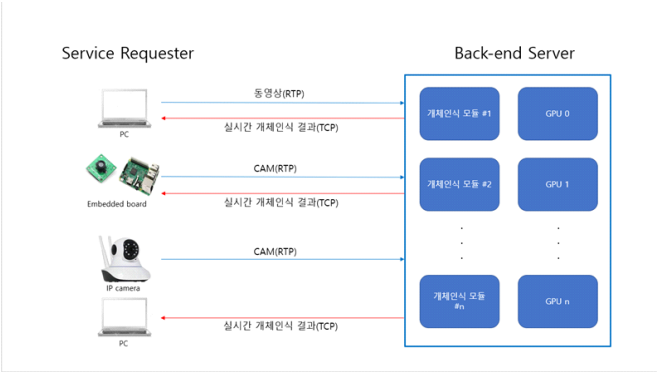
○ 데이터 전송과 저장 처리에 따른 과금으로 인한 비용 발생
○ 외부 인터넷으로 연결이 필요
○ 인터넷 연결 상황에 따른 변동적인 처리속도

특히, 알고리즘 개발단계에서는 클라우드 백엔드 서비스가 적절하지만, 실제로 제품 및 서비스의 필드 적용에서는 위의 제약이 상당한 걸림돌로 작용한다.

그리하여, 실제 영상처리 기반 애플리케이션 또는 서비스를 위해서는 자체적인 서버에 기반한 로컬 또는 기관 백엔드 딥러닝 서비스 기술 개발 필요하다.

그리하여 본 논문에서는 다양한 디바이스 (노트북, 임베디드 보드, IP cam)의 영상 데이터를 로컬 또는 기관에 있는 충분한 자원 활용이 가능한 서버로 전송하여 프로세싱 후 실시간 결과를 feedback 받을 수 있는 back-end 딥러닝 영상분석 서비스 프레임워크를 제안한다.

1) 본 연구는 중소기업벤처부 산학연공동연구개발사업으로 지원되었음 (과제번호: S2654155)



(그림 1) 백엔드 영상처리 서비스 프레임워크

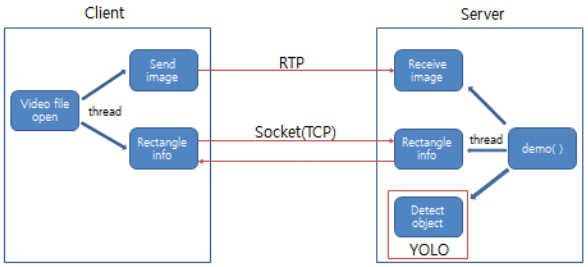
2. 백앤드 딥러닝 서버 기반 영상처리 프레임워크

2.1 백엔드 영상분석 서비스 개요

원격디바이스로부터 서버로 실시간 데이터 전송을 위해서는 잘 알려진 웹을 기반으로 전송하는 HLS [4] 또는 RTP [5]를 이용하여 멀티미디어 데이터를 전송하는 기술이 있다. 이러한 데이터 전송과 더불어 RTCP [6]를 이용하여 데이터와 별도의 채널을 통해 비디오 실행 및 제어에 관한 기능을 수행한다.

본 논문에서도 RTP를 이용하여 비디오 데이터를 실시간 전송한다. 하지만, 기존 RTCP를 사용하는 기존 기술과 달리 새로운 TCP 포트와 전용의 프로토콜을 통해 더욱 빠르고 강력한 실시간 영상처리 서비스를 수행한다.

2.2 서비스 구조



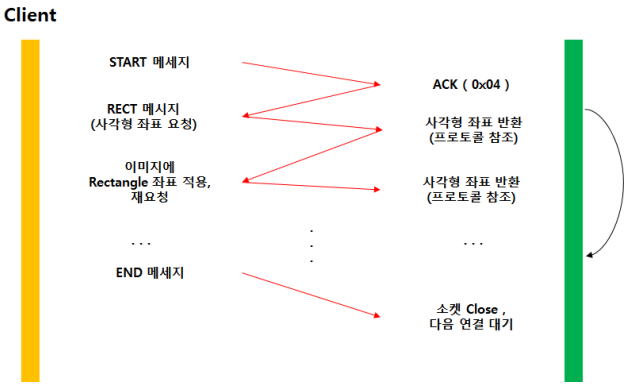
(그림 2) 백엔드 서비스 구조

기본적으로 제안 서비스는 디바이스에서 수행되는 클라이언트 파트와 백엔드 서버파트로 구성되어 있다. 그림 2에서 보는 바와 같이 클라이언트에서는 비디오 파일 OPEN 또는 CAM 영상 실행 모듈과 RTP 서버 기능을 이용한 영상전송 모듈이 있다. 백엔드 서버에는 클라이언트의 영상을 RTP로 받기 위한 영상 수신부가 있으며 이를 기반으로 딥러닝 기반 실시간 개체인식 모듈을 실행한다. 그 결과 정보를 바탕으로 TCP 소켓을 이용하여 클라이언트 디바이스로 처리 결과를 실시간 전송한다. 본 실험에서 우리는 개체인식을 위해 YOLOv3 [7] 를 사용했지만, 해당 개체인식 모듈은 다른 개체인식 모듈 SSD[8]나 사용자 개발 모듈로 다른 부분의 변경 없이 대체될 수 있다.

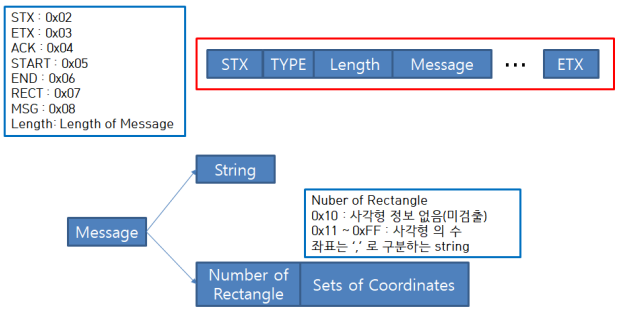
2.3 서비스 프로토콜

그림 3에서 보는 바와 같이 기본적으로 백엔드 서버에서는 클라이언트 (디바이스들)의 연결을 대기하고 있다. 클라이언트는 먼저 백엔드 서버로 TCP 접속을 시도하고, 연결이 수립되면 해당 영상에 대한 정보 (비디오 파일, CAM 데이터 등)를 전송하고, 바로 RTP를 이용하여 백엔드 서버로 영상 전송을 시작한다.

이와 동시에 클라이언트에서는 처리요청 메시지를 실시간 전송하고 (본 예제에서는 개체인식에 대한 BOUNDING BOX 정보를 요청), 이에 대한 영상처리 결과가 실시간 전송된다.



(그림 3) 서비스 메시지 Flow



(그림 4) 메시지 Format

그림 4는 이러한 백엔드 서비스를 위한 기본 메시지 포맷을 정의한다. 해당 포맷은 메시지의 타입을 추가로 정의하고 그에 대한 포맷을 추가로 정의하여 확장할 수 있다.

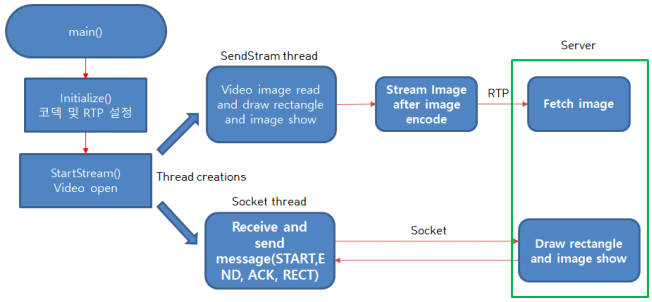
3. 실험 결과

3.1 실험 환경

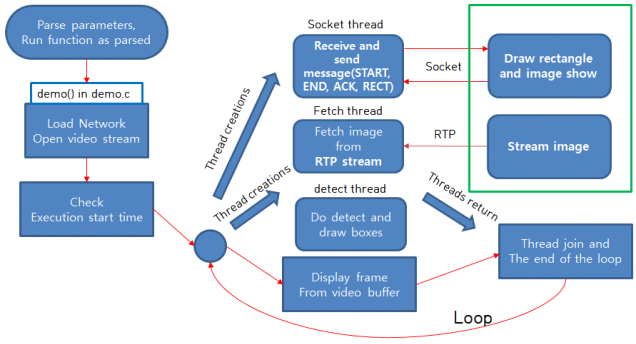
본 논문에서 제안하는 서비스 프레임워크는 서버와 클라이언트 파트로 구분되어 소프트웨어로 구현하였다. 서버는 일반적으로 사용할 수 있는 가장 최소사양의 GPU(GEFORCE 1060)를 탑재한 PC를 사용하였으며, 클라이언트로는 파이 카메라가 연결된 라즈베리 파이, 일반 노트북과 동영상 파일, 그리고 IP CAM과 PC를 사용하였다.

3.2 클라이언트/서버 백엔드 서비스 구현

RTP의 서버와 클라이언트 구현은 FFMPEG 라이브러리 [8]를 이용하여 구현하였으며, 서버는 Linux 기반의 소프트웨어를 개발하였으나, 클라이언트는 linux와 windows를 모두 지원할 수 있도록 개발하였다.



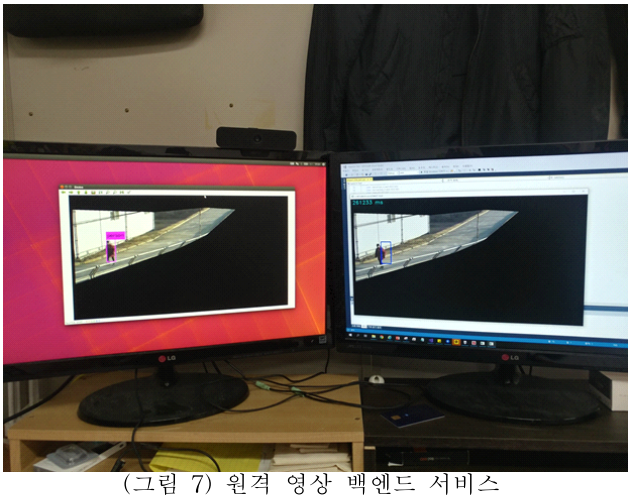
(그림 5) 클라이언트 소프트웨어 구성



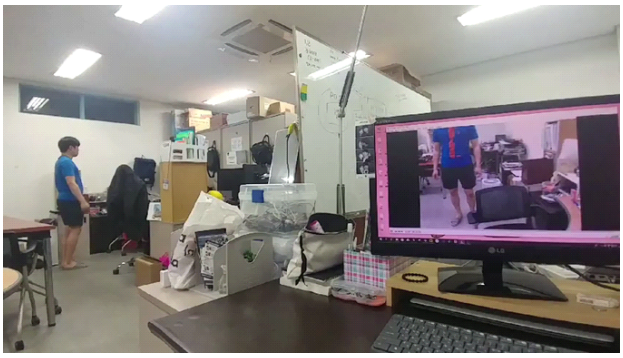
(그림 6) 서버 소프트웨어 구성

3.3 서비스 구현 결과

그림 7은 PC 클라이언트에서 실행시킨 동영상 파일을 백엔드 서비스를 통해 실행 중에 개체의 움직임 및 인식 결과를 실시간으로 서비스 받는 결과를 보인다. 그림에서 왼쪽은 서버의 실행결과이며, 동시에 오른쪽은 클라이언트에서 실시간 서비스를 받는 결과를 보인다.



(그림 7) 원격 영상 백엔드 서비스



(그림 8) IP CAM을 이용한 원격 영상 백엔드 서비스

그림 8에서는 IP CAM의 카메라 영상을 백엔드 서버로 전송하고, 이를 다른 PC에서 영상분석 결과를 지원받는 서비스의 구현을 보인다.

3.4 성능평가

3.4.1 Video Throughput

영상전송 성능을 보기 위해 우리는 video 데이터의 throughput을 측정했다. 표에서 보는 바와 같이 서로 다른 동영상 5개에서 모두 300Mbps 이상의 throughput을 보였으며, 이를 통해 평균적으로 15 frame/sec을 초과하는 성능으로 성공적인 동영상 실시간 전송을 가능케 했다.

(표. 1) Throughput 실험 결과

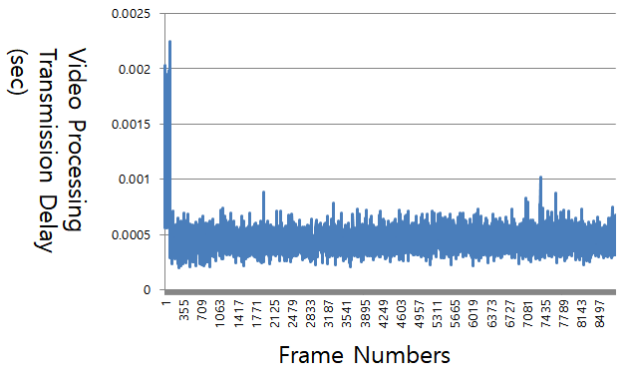
동영상	A	B	C	D	E	평균
Throughput(bps)	352922	437743	436280	313987	325274	373241

3.4.2 프로세싱 처리 결과 전송 지연

Video throughput이 RTP에 의한 영상 데이터의 전송속도 및 성능을 측정한 반면, 전송된 영상에 대한 분석 결과 피드백을 받는 데 있어서의 전송 지연 또한 백엔드 서비스에서의 중요한 성능지표 중 하나이다. 우리는 서비스 요

청마다 이에 대한 피드백 결과를 받는 시간 (즉, Round trip 시간)을 측정했다. 그림 9에서 보는 바와 같이 실험 결과는 전체적으로 평균 50usec로 영상 데이터를 분석한 결과가 지속해서 전송되고 있는 것을 확인할 수 있으며, 이것은 매우 빠른 속도로 처리 결과가 피드백되고 있음을 보여준다.

초반 약 2msec의 지연이 발생하는 이유는 초기 영상 정보에 대한 파일 전송 및 백엔드 서버에서 개체인식 모듈 로딩을 위한 시간이 추가되는 부분이다.



(그림 9) 프로세싱 처리 결과 시간 측정

4. 결론

본 논문에서는 다양한 디바이스의 영상 데이터를 로컬 또는 기관에 있는 충분한 자원 활용이 가능한 서버로 전송하여 프로세싱 후 실시간 결과를 feedback 받을 수 있는 효율적인 back-end 딥러닝 영상분석 서비스 프레임워크를 제안했다. 실험을 통해 제안된 서비스가 다양한 환경에서 잘 동작하는 것을 확인하였으며, 성능적인 측면에서 우수한 비디오 전송속도와 빠른 처리 결과 전송시간을 보이는 것을 확인하였다. 본 논문에서 제안된 로컬 백엔드 서비스 프레임워크를 통해 보다 다양한 영상 소스로부터 보다 공격적인 백엔드 영상분석을 수행할 수 있는 개발 및 필드 환경이 가능해질 것으로 사료된다.

참고문헌

[1] Amazon Rekognition, https://aws.amazon.com/ko/rekognition/

[2] Azure Stream analytics, https://azure.microsoft.com/ko-kr/services/stream-analytics/

[3] Google Video Intelligence API, https://cloud.google.com/video-intelligence/

[4] HLS, Pantos, R.P. (April 2016). "HTTP Live Streaming draft-pantos-http-live-streaming-19". Network Working Group: 1.

[5] RFP 3550, RTP Full Standard, http://www.cs.columbia.edu/~hgs/rtp/rfc3550_full.txt

[6] OTT, Joerg; PERKINS, Colin. Guidelines for extending the RTP control protocol (RTCP). 2010.

[7] Shafiee, Mohammad Javad, et al. "Fast YOLO: a fast you only look once system for real-time embedded object detection in video." arXiv preprint arXiv:1709.05943 (2017).

[8] Liu, Wei, et al. "Ssd: Single shot multibox detector." European conference on computer vision. Springer, Cham, 2016.

[9] FFMPEG, https://ffmpeg.org/download.html