

Deepstream을 활용한 고속도로 다중 CCTV 영상 기반 실시간 교통 흐름 분석 기술 연구

백장현, 하상정, 한서우, 장수현*

한국전자기술연구원

baek0307@keti.re.kr, sangjeong@keti.re.kr, khan5555@keti.re.kr, *shjang@keti.re.kr

A Study on Real-Time Traffic Flow Analysis systems based on Highway CCTV Multi-Video Stream Using Deepstream

Baek Jang Hyun, Ha Sang Jeong, Han Seo Woo, Jang Soo Hyun*

Korea Electronics Technology Institute (KETI)

요약

본 논문은 고속도로 정체 구간을 줄이기 위해 고속도로에 설치된 여러대의 CCTV 영상을 하나의 모델에서 실시간으로 분석하고 그 결과 데이터를 서버에 전송하는 연구를 진행하였다. 딥러닝을 활용하여 차량의 속도를 추정하고 차로별 교통량을 분석하여 운전자가 능동적으로 경로를 선택하게 함으로써 고속도로 내 정체 구간의 길이가 줄어들도록 유도한다. Deepstream을 활용하여 기존의 다채널 비디오 분석 구조가 가지는 하드웨어 자원 활용의 비효율성이라는 문제점을 개선하고 적은 하드웨어 자원으로도 동일한 성능을 유지하며 실시간성을 확보하는 방법을 제시하였다.

I. 서론

고속도로 운행량이 늘어나면서 교통 혼잡 및 도로 내 정체 상황 발생이 증가하고 있다. 고속도로의 경우 많은 분기 및 합류 도로가 존재하여 특정 구간에서 빈번한 정체 상황이 발생하고, 이러한 구간의 정체가 지속 되면 전반적인 정체 구간의 길이가 늘어나게 된다. 개별 차량의 주행 속도 및 차로별 교통량을 실시간으로 분석하고 그 정보를 운전자에게 전달할 수 있다면 어떠한 차선으로 주행하는 것이 가장 효과적일지에 대해 운전자 스스로 예측하고, 정체 구간에 진입하기 전 능동적으로 차선을 변경할 수 있을 것이며 결과적으로 정체 구간의 길이를 줄일 수 있다.

본 논문에서는 고속도로 다채널 CCTV 영상을 하나의 모델 내에서 실시간으로 분석하여 교통 흐름에 대한 정보를 서버에 전송하는 시스템을 제안 한다.

II. 본론

2.1 Deepstream을 활용한 실시간 교통 흐름 분석 모델

본 논문은 영동 고속도로 내 특정 구간(마성IC ~ 신갈 JC)에 설치된 8개의 CCTV 영상을 RTSP로 실시간 스트리밍으로 받아오며 그림1과 같은 방식으로 CCTV 영상에서 개별 차량의 속도 및 차로별 교통량을 분석하고 결과 데이터를 서버에 전송한다.

먼저 입력으로 들어오는 모든 영상에 대해 몇 번째 CCTV의 영상인지 확인 후 불필요한 부분은 마스킹 처리한다. 이후 프레임부터는 YOLO V3 네트워크를 이용하여 영상에 존재하는 모든 차량을 검출하고 NVIDIA에서 제공하는 NvDCF 트래커를 활용하여 차량별 ID를 받는다. 검출된 차량의 Bounding Box 하단 중점 좌표의 움직임을 기반으로 차량별 속도를 추정하고 해당 차량이 몇 번 차로에 존재하는지 파악 한다. 현재 각 차로별 존재하는 차량의 평균 속도를 계산하고 이렇게 분석된 데이터를 Apache Kafka를 통해 서버에 JSON 포맷으로 전송한다.

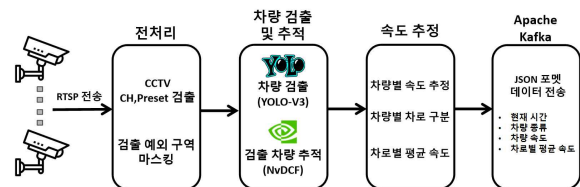


그림 1. 다채널 CCTV 영상 기반 실시간 교통 흐름 분석 시스템 흐름도



그림 2. 다채널 CCTV 영상 기반 실시간 교통 흐름 분석 이미지

2.2 Deepstream을 이용한 다중 CCTV 영상 분석 및 기존 방법 비교

기존의 딥러닝 프레임워크의 경우 하나의 모델이 단일 영상에 대한 처리만을 지원한다. CCTV 8대 영상에 대해 기존의 다채널 비디오 분석 구조로 작업을 진행할 경우 각 CCTV 영상에 대해 하나의 프로세스로 동작하며 독립된 메모리 영역에서 작업을 진행하기 때문에 검출되는 차량의 대수가 늘어날수록 트래킹 및 속도 추정 단계에서 Context Switching이 자주 일어나게 되고 오버헤드 발생이 증가하여 성능이 저하되는 단점이 존재한다.[1] 이러한 문제를 해결하기 위해 본 논문에서는 그림3과 같이 NVIDIA에서 제공하는 Deepstream5.1 프레임워크를 활용하여 하나의 모델에서 8개의 CCTV 영상에 대한 실시간 처리를 진행하고 처리된 교통 데이터를 서버에 전송하는 연구를 진행하였다.

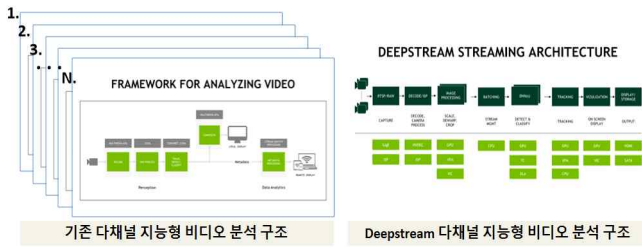


그림 3 . 기존 다채널 영상 분석 구조와 Deepstream 다채널 영상 분석 구조 비교

Deepstream을 이용한 다중 CCTV영상 분석 및 기존 방법 성능 비교를 위해 FHD 화질의 영동선 실증구간 고속도로 CCTV 8대의 녹화 영상으로 테스트를 진행하였다. 차량의 검출을 위해서 직접 가공한 15000장의 데이터로 학습시킨 YOLO v3 네트워크를 사용하였으며 실시간성을 위해 두가지 방식 모두 INT8 정밀도의 TensorRT 엔진으로 모델을 변환하였다. 차량의 추적을 위해서 기존 다채널 지능형 비디오 분석 구조에서는 DeepSORT 트래커를 사용하였고 Deepstream을 활용한 다채널 지능형 비디오 분석 구조에서는 NVIDIA에서 제공하는 NvDCF 트래커를 사용하였다.[2]

성능 평가 결과 같은 검출 모델을 사용하여 mAP 95.6%의 검출 성능을 보여주었으나 테스트를 진행하는 동안 하드웨어 자원 사용량에서 두 방식은 표1, 그림4와 같이 큰 차이를 보여주었다. 기존의 멀티 프로세스 개념의 다채널 지능형 비디오 분석 구조의 경우 CPU 사용량은 87.2% , GPU 메모리 사용량은 91.3%에 달하였다. 반면 Deepstream 기반 다채널 지능형 비디오 분석 구조의 경우 CPU 사용량은 14.8%, GPU 메모리 사용량은 10.9%로 기존 방식의 1/5 이하의 하드웨어 자원을 소모하면서도 30fps 영상에 대한 실시간 인퍼런스가 가능하였다.

따라서, 다채널 지능형 비디오 분석 구조에서 Deepstream을 활용 하면 더 적은 하드웨어 자원으로도 동일한 검출 및 추적 성능을 보이면서 동시에 실시간 분석을 위한 대량의 연산 문제도 해결할 수 있다.

표 1. 하드웨어 자원 사용량

	기본 분석 구조	Deepstream 분석 구조
Inference Resolution	608 * 608	608 * 608
Precision	INT 8	INT 8
Model Accuracy	95.6%	95.6%
CPU 사용량	87.2 %	14.8%
GPU 메모리 사용량	91.3%	10.9%
인퍼런스 처리 속도 (FPS)	16	30

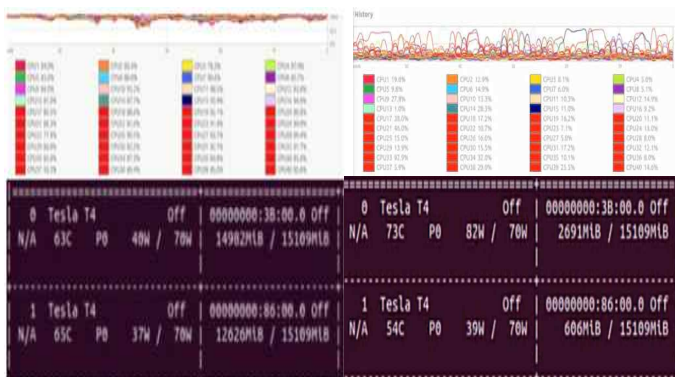


그림 4 . 기존 다채널 영상 분석 구조(좌)와 Deepstream 다채널 영상 분석 구조 하드웨어 사용량 비교

III. 결론

본 논문에서는 고속도로 정체 구간을 줄이기 위해 고속도로 내 여러대의 CCTV 영상을 하나의 모델에서 실시간으로 분석하여 결과 데이터를 서버에 전송하는 방법을 제안하였다. 기존의 멀티 프로세스를 통한 작업을 진행하던 다채널 영상 분석 구조에서 하나의 프로세스 내에서 다중 영상처리를 가능하도록 NVIDIA Deepstream5.1을 이용하였다. 결과적으로 GPU와 CPU 등 하드웨어 자원을 최대한 효율적으로 사용하였고 이를 통해 기존의 다채널 영상 분석 구조에 비해 적은 하드웨어 자원을 사용하면서 기존 방식과 동일한 검출 성능과 교통 흐름 분석의 실시간성을 확보하였다.

ACKNOWLEDGMENT

본 연구는 한국지능정보사회진흥원 국가인프라 지능정보화 사업인 “5G와 지능형 인프라를 통한 고속도로 운영관리 시스템 실증” 과제의 지원에 의해 수행되었음

참 고 문 헌

- [1] Seo Dae Wha. "Parallel Processing : Distributed Shared Memory Scheme for Multi-thread programming". The KIPS Transactionsty, vol. 3, pp.791-802, 1996.
- [2] Abdulghafoor, Nuha H.; Abdullah, Hadeel N. "Real-Time Moving Objects Detection And Tracking Using Deepstream Technology", Journal of Engineering Science and Technology, 16.1: 194-208, 2021.
- [3] www.nvidia.com