DeepStream 이용한 다중 영상처리 딥러닝 객체검출 기법 성능비교

박진배

한화시스템

smallchange@hanwha.com

Performance comparison of multiple image processing deep learning object detection techniques using DeepStream

Jinbae Park

Hanwhasystems Corp.

요 약

최근, IoT 기술이 발전함에 따라, 엣지 디바이스에서의 딥러닝 기술 적용에 대한 연구가 활발히 진행되고 있다. 기존의 네트워크 엣지에서의 비디오 분석은 엣지 카메라에서 촬영된 영상을 엣지 서버로 전송하여 필요한 작업을 처리하는 방식이었다. 이 방식에서는 다수의 엣지 카메라가 동시에 영상을 전송할 때, 불필요한 통신으로 지연이 발생할 수 있다. 이에 따라 엣지 디바이스에서는 서버 연결없이, 엣지 디바이스에서 처리하는 실시간 성이 요구되고 있다. 본 논문에서는 NVIDIA DeepStream SDK를 이용한 객체 검출 파이프라인을 구성하고, 이에 따른 영상처리 객체 검출에 대한 실시간 성능을 분석한다.

Ⅰ. 서론

기존 IoT에서 카메라 영상 분석은 카메라에서 촬영된 영상을 서버로 전송하여 영상처리를 수행하는 방식이었다. 하지만 이 방식에서는 카메라에서 영상을 서버로 보내는 데 지연이 발생하고, 다수의 영상이 서버로 전송될 경우 병목현상 및 서버에 과부하가 걸릴 수 있다.

최근 이를 해결하기 위해, 엣지 디바이스의 하드웨어 성능이 향상되면서 엣지 디바이스를 이용하여 다양한 어플리케이션에 적용이 가능해졌다. 그 중 NVIDIA에서는 최근 딥러닝에 많이 활용되고 있는 그래픽카드 뿐만 아니라, Tensor Core를 탑재한 GPU가 들어있는 엣지 디바이스인 Jetson 시리즈를 출시하였다.[1] 본 논문에서는, NVIDIA DeepStream SDK를 이용하여 다중 영상처리를 엣지 디바이스에서 수행하고 그 성능을 분석한다.

Ⅱ. 하드웨어 및 파이프라인 구성

본 논문에서는 객체 검출 AI를 적용하기 위한 디바이스로 Jetson Xavier AGX를 이용하였다. Jetson Xavier AGX는 NVIDIA에서 만든 엣지 디바이스로 성능은 다음과 같다.

항목	내용
GPU	512-Core Volta GPU with Tensor Cores
CPU	8-Core ARM v8.2 64-Bit CPU, 8 MB L2 + 4 MB L3
Memory	32 GB 256-Bit LPDDR4x | 137 GB/s
Storage	32 GB eMMC 5.1
DL Accelerator	(2x) NVDLA Engines
Vision Accelerator	7-Way VLIW Vision Processor
Encoder	4Kp60, H265(HEVC)/H.264/VP9 supported
Decoder	8Kp30(H.265 only), 4Kp60, H265(HEVC)/H.264/VP9 supported
Size	105mm x 105mm
Deployment	Module (Jetson Xavier)

표 1 Jetson Xavier AGX 성능

기존 객체검출 방식의 경우, OpenCV 라이브러리를 사용하여 검출을 수행한다. 그러나, Xavier AGX와 같은 Edge 디바이스에서 OpenCV를 사용했을 경우 하드웨어 스펙이 높은 편이 아니기 때문에, 다중 영상처리를 할 경우, 성능 저하가 생긴다. 이를 극복하기 위해 NVIDIA에서는 다중 영상처리를 파이프라인을 구성하여 수행할 수 있도록 Deepstream SDK를 고안하였다. Deepstream SDK는 Open source Multimedia Framework인 Gstreamer를 변형한 형태로[2], TensorRT 엔진을 수행할 수 있는 Gst-nvinfer 플러그인이 탑재되어 있으며, Close Source 형태로 구현되어 있다.

본 논문에서는 다중 영상을 동시에 객체검출을 추론하기 위한 파이프라인을 구성하였다. 객체 검출을 추론하기 위한 파이프라인의 구성은 다음과 같다.

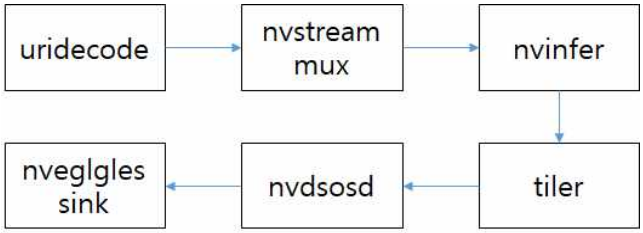

그림 1 파이프라인 구성도

uridecode에서는 영상을 가져와서 디코드하는 역할을 수행한다. 그 다음 디코드 한 영상을 하나의 프레임처럼 만들기 위해 nvstreammux를 사용한다. 그리고 nvinfer를 사용하여 동시에 ai모델을 적용한다. 분석한 프레임을 시각화하여 하나의 영상으로 출력하기 위해, tiler, nvdsosd, nveglglessink를 사용하였다.

Ⅲ. AI 모델 경량화

엣지 디바이스에서는 하드웨어 성능이 PC보다 대체적으로 낮기 때문에, PC에서 적용하는 AI 모델을 그대로 가져와서 사용하기 어렵다. 또한 자율주행과 같이, 실시간 성이 중요시 되는 분야에서는 모델의 정확도도 중요하지만, 모델이 동작하는 속도가 관건이기 때문에 모델 경량화가 중요한 팩터이다.

본 논문에서는 Yolov3 모델을 엣지 디바이스에 적용하기 위해, 기존 Float32 모델을 경량화하고자 TensorRT 기법을 적용한다[3],[4],[5]. TensorRT는 모델 추론 최적화/가속화를 위한 AI 모델 경량화 툴이다. 다음은 TensorRT가 동작하는 원리를 대략적으로 나타낸 것이다.

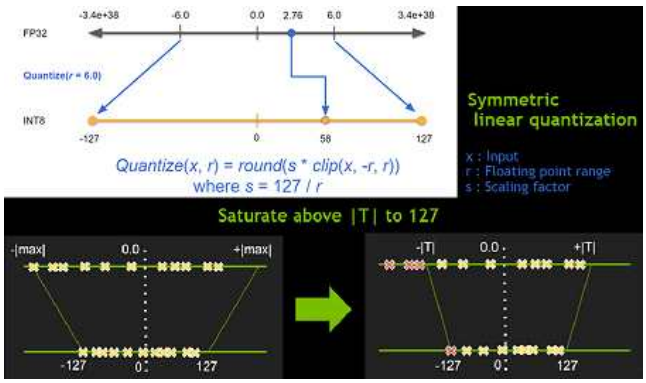

그림 2 TensorRT Quantization & Precision Calibration

Ⅳ. 실험결과

본 논문에서는 다중 영상처리 객체검출 기법에 대한 성능을 분석했다. 객체 검출용 영상은 딥스트림에서 제공하는 기본 샘플 비디오를 사용하였다. 영상 속성은 해상도 1280x720, 프레임 30fps, h264 코덱이다. 객체검출용 모델은 YoloV3(Input 608)을 적용하였다. 엣지 디바이스에서의 성능을 조금 더 세분화하여 측정하고자, Tensor RT로 모델을 경량화하여 Float32, Float16, Int8 별로 측정하였다. 영상은 최대 4개까지 동시에 영상처리를 수행하여 성능을 비교하였다. 영상의 최대 개수는 4개이다.

그림 3 객체검출 추론을 수행한 영상

연산량 별 다중 영상처리 객체검출 성능(FPS)의 결과는 다음과 같다.

	1개	2개	3개	4개
FP32	9.80	4.87	3.0	2.0
FP16	33.61	17.81	12.13	8.94
Int8	59.61	34.28	20.94	17.97

표 2 Yolov3 연산량 별 FPS 결과

연산량이 낮아질수록 객체검출 추론속도가 확연히 빨라짐을 볼 수 있다. 또한 기존 FP32 모델을 사용했을 경우에는, 영상 4개를 수행했을 때 엣지 디바이스 성능이 저하됨을 볼 수 있다. 실제로 온도 측정을 해보니, GPU에서 성능처리 INT8로 경량화 했을 경우 어느정도 실시간 성을 보장할 수 있었음을 알 수 있다.

Ⅴ. 결론

본 논문에서는 다중 영상처리 객체검출 기법에 대한 성능을 분석했다. 엣지 디바이스에서 영상의 개수에 따른 성능 분석도 있었지만, 사실 영상을 많이 띄울수록 성능이 낮아지는 것은 당연한 결과이다. 그럼에도 불구하고, 객체검출 경량화에 따른 성능차이가 생각보다 많았음을 알 수 있었다. 실시간성을 보장하기 위해서는 딥러닝 객체검출 모델에 대한 경량화는 필수적으로 적용되어야 할 것이다.

참 고 문 헌

[1] NVIDIA, "NVIDIA Jetson의 임베디드 시스템", (https://www.nvidia.com/ko-kr/autonomous-machines/embedded-systems/)

[2] NVIDIA, "NVIDIA DeepStream SDK | NVIDIA Developer", (https://developer.nvidia.com/deepstream-sdk).

[3] Gstreamer."Open Source Multimedia Framework", (https://gstreamer.freedesktop.org/)

[4] NVIDIA, "NVIDIA TensorRT", (https://developer.nvidia.com/tensorrt).

[5] YOLO, "YOLO: Real-Time Object Detection", (https://pjreddie.com/darknet/yolo/).