드론 기반 실시간 객체 식별을 위한 추론 가속화 평가

권승상 [1*], 문용혁 [2,3**]
[1] 충북대학교 정보통신공학부
[2] 한국전자통신연구원 인공지능연구소
[3] 과학기술연합대학원대학교 한국전자통신연구원스쿨
kss9813@gmail.com, yhmoon@etri.re.kr

An Evaluation of Inference Acceleration for Drone-based Real-time Object Detection

Seung-Sang Kwon[1], Yong-Hyuk Moon[2,3*]
[1] School of Information and Communication Engineering, Chungbuk National University
[2] Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea
[3] University of Science and Technology (UST), Daejeon, Korea

요 약

최근 데이터 획득 위치에 가장 근접하고, 저 수준의 계산력을 제공하는 엣지 기기를 중심으로 직접 딥러닝 추론을 수행하고자 하는 요구가 증가하고 있다. 본 논문에서는 드론에서 촬영한 교통 영상 데이터를 기반으로, 다수의 차량 종류 및 보행자를 식별하는 모델을 Jetson Nano 에 탑재하여 기본 성능을 측정한다. 더불어, 자원제약형 기기 환경에서 TensorRT 와 Deepstream 을 활용하여 객체 식별 모델의 연산 경량화 및 추론 가속화 성능을 극대화하기 위한 구현 및 실험을 수행하여 Anchor-based 및 Anchor-free 객체 식별 모델의 정확도와 실시간 대응력을 평가하고 논의한다.

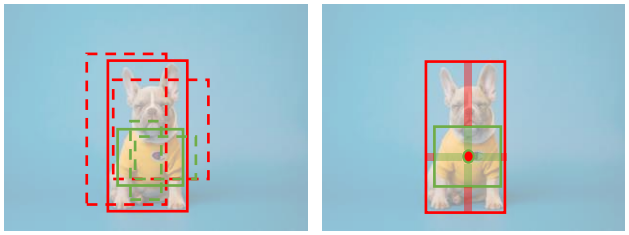
1. 서론

오늘날 인공지능의 정확도는 모델의 크기에 비례하여 향상되어 왔다. 그러나, 최근에는 어느 정도의 정확도 손실을 감내하되 추론 속도를 높이기 위한 딥러닝 기술에 대한 연구개발이 활발하게 전개되고 있다. 본 논문에서는 자원 제약적인 NVIDIA Jetson Nano [1] 기기가 탑재된 드론 기기에서 실시간으로 교통 수단 및 보행자를 식별할 수 있는 Object Detection 모델을 활용하여, 추론 정확도와 속도를 측정 및 평가하고자 한다. 실험 대상 모델은 Anchor-based 기반의 대표 모델인 YOLOv5 [2]와 Anchor-free 기법으로 최근 제안된 YOLOX [3]을 대상으로 한다. 특히, TensorRT [4]을 활용하여 모델 최적화 및 경량화를 수행하고, Deepstream [5] 기반의 스트리밍 연산 속도 개선 비교를 통해 주요 성능 시사점에 대해 논의한다.

2. 객체 식별 모델

본 논문에서는 비교적 높은 정확도와 준수한 연산 속도를 제공하는 사용하는 YOLO 계열의 모델을 사용한다. 또한 Anchor-based/-free 두 기법을 대표하는 모델을 YOLO 계열로 한정함으로써 보다 객관적인 실험 비교가 가능하도록 설정한다. 두 기법의 주요한 학습 방식 차이는 아래와 같이 요약될 수 있다.

<그림 1> Anchor-based model Vs. Anchor-free model



먼저 Anchor-based Detector 는 좌측 그림과 같이 실선으로 표현된 여러 Anchor Boxes (객체 형상에 대한 가정) 중에서 각 객체(개와 옷)와의 IOU(Intersection Over Union) 수치가 특정 값 이상인 Bounding Boxes (점선 박스)을 학습한다. 본 기법은 Anchor Box 크기, 종횡비, 개수에 따라 성능에 민감하다는 단점이 있다. 이를 극복하기 위해 우측 그림과 같이 Anchor Box 없이 (Anchor-free) 객체의 Center Point [6]를 기준으로 Positive Samples 을 식별하고 이를 기준으로 가로, 세로 물체 크기를 예측하는 대안으로 최근 제안되었다.

*교신 저자

<표 1> 이미지 데이터를 활용한 추론 엔진 별 모델 성능

Models	Parameters	PyTorch				TensorRT				TensorRT with Quantization FP16			
		mAP@0 .5	mAP@0.5:0.95	mAR	Latency(ms)	mAP@0 .5	mAP@0.5:0.95	mAR	Latency(ms)	mAP@0 .5	mAP@0.5:0.95	mAR	Latency(ms)
YOLOv5 Small	7.04M	27.0	14.2	30.7	131.5	27.1	14.3	31.0	110.6	27.1	14.3	30.9	75.5
YOLOv5 Nano	1.77M	21.6	10.6	26.3	64.1	21.7	10.7	26.5	48.1	21.7	10.7	26.4	33.7
YOLOX Small	8.94M	24.2	13.2	22.8	235.2	24.3	13.3	22.9	164.4	24.3	13.2	22.9	113.4
YOLOX Tiny	5.04M	15.5	7.9	14.2	84.8	15.5	7.9	14.2	68.0	15.5	7.9	14.2	46.1
YOLOX Nano	0.90M	11.8	5.9	11.1	55.4	11.8	5.9	11.1	39.8	11.8	5.9	11.1	37.5

3. 드론 기반 객체 식별을 위한 시스템 구성

본 장에서는 VisDrone [7] 데이터 셋을 활용하여 드론 기반 객체 식별 모델을 학습하기 위한 기본 구성과 엣지 추론 기기에 대해 논의한다.

3.1. 데이터 구성

드론에서 촬영한 오픈 데이터 셋인 VisDrone2019-DET Dataset 은 총 8,629 장의 Images 와 관련 Annotation 으로 제공되며, “Pedestrian, People, Bicycle, Car, Van, Truck, Tricycle, Awning-tricycle, Bus, Motor”와 같은 10 가지 Object Category 로 구성된다.

<표 2> VisDrone2019-DET Dataset 구성

	Train	Validation	Test
Images	6,471	548	1,610
	74.99 %	6.35%	18.66%

3.2. 베이스라인 모델 학습

모델 훈련을 위해 Ubuntu 18.04.6 LTS 운영체제와 GPU(NVIDIA TITAN RTX 24GB)를 사용하였으며, 이미지 해상도는 YOLOv5-Small/Nano 640, YOLOX-Small 640, YOLOX-Tiny/Nano 416 으로 각각 설정하였다. 공통적으로 Batch Size 32, Epoch 300 으로 설정하여 PyTorch 프레임워크 기반으로 학습하였다.

3.3. 객체 식별 추론 기기

상기 두 가지 객체 식별 모델을 추론 평가하기 위해 이에 적합한 자원 제약형 하드웨어 규격을 제공하는 NVIDIA Jetson Nano 을 활용한다. 본 엣지 기기는 GPU 연산을 지원하며, 5~10W/hrs 전력을 소비하는 특성이 있어 실제 드론에 탑재가 용이하다. 단순히 저장된 이미지 추론이 아니라, 카메라를 통한 실시간 스트리밍 프레임을 처리하기 위해 1080p/30fps 성능을 지원하는 카메라 장비를 탑재하였다.

<표 3> NVIDIA Jetson Nano 엣지 규격

CPU	Quad-Core ARM A7 @ 1.43 GHz
GPU	128-Core NVIDIA Maxwell, 472 GFOPS
Memory	4GB 64-bit LPDDR4 25.6 GB/s

4. 모델 경량화 및 연산 가속화 기반 추론 평가

본 장에서는 자원 제약적인 Jetson Nano 엣지 기기에서 추론 실험을 진행하고, 모델 경량화 및 연산 가속화가 추론 성능 개선에 미치는 영향을 평가한다.

4.1. TensorRT 기반 모델 최적 변환

TensorRT 는 학습된 딥러닝 모델을 NVIDIA GPU 아키텍처에 맞게 최적화하여 추론 속도를 향상하는 모델 최적화 엔진이다. 저자는 PyTorch 로 구현한 YOLOv5, YOLOX 모델을 TensorRT 로 최적화하여 Engine (변환 모델)으로 구성하고, 두 Engine 의 출력 Logit 을 TensorRT 가 이해할 수 있는 형상으로 구성하기 위해 별도의 C++ Parser 를 구현하여 적용하였다.

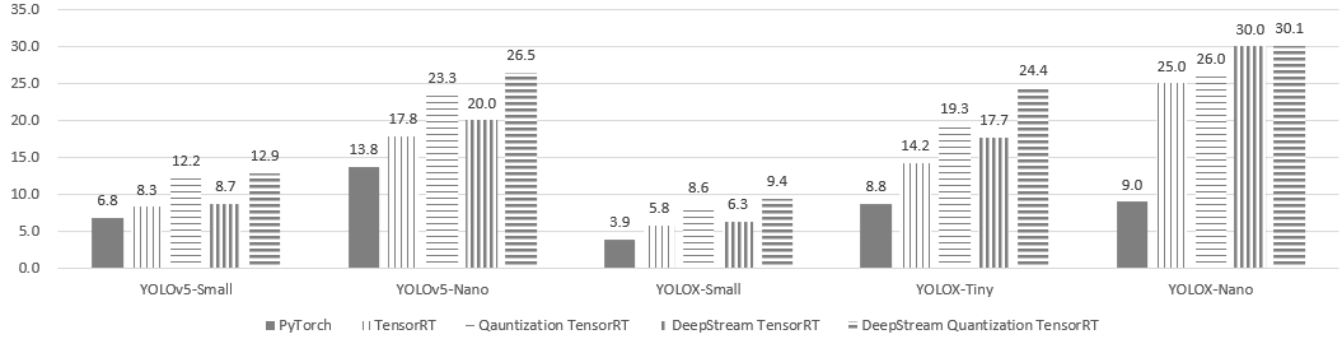
4.2. Quantization 기반의 모델 경량화

대부분의 모델은 정확도 극대화를 위해 FP(Full Precision) 32bits 규격으로 가중치를 학습하고 저장한다. 본 논문에서는 양자화(Quantization)을 통한 모델 경량화가 직접적으로 추론 연산 속도 개선에 영향을 미칠 수 있다는 점에 착안하여 TensorRT 의 Symmetric Linear Quantization 을 적용하여 네트워크 가중치를 FP16 으로 양자화 하였다. 동시에 원본 모델과의 정확도 손실 차이 역시 실험을 통해 측정한다.

4.3. 이미지 데이터를 활용한 추론 실험 평가

표 1 은 PyTorch, TensorRT, TensorRT with Quantization -FP16 세 가지 추론 엔진 구성에 따른 YOLOv5, YOLOX 모델의 크기 별 추론 정확도 및 속도를 보인다. 먼저 모델 변환 및 경량화로 인한 정확도 손실은 거의 없는 것으로 확인하였다. 또한 PyTorch 원본 모델의 경우 YOLOv5-small 의 매개변수가 1.9 백만개 더 적음에도 불구하고 YOLOX-small 대비 더 높은 정확도와 빠른 추론을 제공한다. 이와 유사하게 mAP@0.5, mAR 성능 측면에서 Anchor-free YOLOX-nano 의 성능이 Anchor-based YOLOv5-nano 와 비교하여 절반 수준에 그치고 있는 반면에, 상대적으로 추론 지연 시간이 우수한 것으로 확인하였다. 특히 TensorRT with Quantization-FP16 이 적용된 모델은 PyTorch 원본 모델에 비해 약 1.5 ~ 2 배의 추론 지연 시간 축소를 달성할 수 있어 본 실험에서 모델 최적 변환 및 경량화의 필요성을 명확히 확인할 수 있다.

<그림 2> 실시간 스트림 기반의 경량화 기법 별 초당 프레임 수 (FPS: Frames Per Second)



4.4. Deepstream 기반의 스트림 데이터 처리

본 논문에서는 카메라를 통해 실시간으로 유입되는 스트리밍 영상 데이터를 대상으로 모델의 추론 속도를 개선하는 것에 초점을 맞추고 있다. 추론 가속화를 위한 세 번째 기법으로 비디오 분석 응용 프로그램의 고성능 개발을 손쉽게 할 수 있도록 만든 Deepstream 라이브러리를 활용한다. 공통적으로 VisDrone2019 Test 이미지 데이터 1,610 개를 스트림 동영상으로 구성하여 카메라를 통해 실제 촬영 하는 방식을 채택하였다. 이를 통해 모델 변환 및 경량화를 통해 달성할 수 있는 Model Throughput 향상과 스트림 데이터 전/후처리 가속화를 통한 추론 속도(FPS: Frames Per Second) 개선을 통합하여 전체 성능 개선을 비교 분석할 수 있다.

4.5. 실시간 스트림 데이터의 추론 실험 결과

그림 2 는 1080p/30fps 해상도를 지원하는 카메라를 Jetson Nano 에 장착하고, 4.1, 4.2 및 4.4 절에서 논의된 세 가지 기법을 모두 적용하였을 때 실시간 스트림에 대한 모델 추론 속도(FPS)를 비교한 결과다.

전체적으로 PyTorch 프레임워크 대비 TensorRT with Quantization-FP16 and Deepstream 을 적용한 모델에서 약 2 ~ 3.3 배의 FPS 성능을 달성하였다. 또한, 4.3 절의 실험 결과와 동일하게 Deepstream 을 추가로 도입하여도 Anchor-free YOLOX 계열 모델에서 보다 빠른 추론 연산 속도 향상이 확인되었다. 다른 결과 특성은 TensorRT with Deepstream 이 모델 기법 및 경량화 정도와 무관하게 TensorRT with Quantization-FP16 보다 FPS 성능이 공통적으로 낮게 도출된다는 것이다. 본 결과는 30fps 수준의 스트림 입력과 1.7 ~ 9 백만개 수준의 매개변수 조건에서는 양자화를 통해 모델의 Feed-forwarding 연산 시간을 줄이는 것이 전체 추론 속도 개선에 보다 중요하다는 것을 의미한다.

특히 세 가지 기법을 모두 적용한 YOLOX-nano 에서 카메라 입력 스트림 처리의 실시간성을 만족하였는데 더 높은 초당 프레임 수 입력을 지원하는 카메라를 적용할 경우 Deepstream 활용 시 보다 높은 FPS 달성이 가능할 것으로 예상된다.

5. 결론 및 향후 연구 계획

본 논문에서는 드론에 장착된 엣지 기기에서 관측될 수 있는 교통 이미지와 실시간 스트림 데이터를 대상으로 YOLO 계열 두 가지 대표 모델을 활용하여, 1) Anchor Box 기반 학습 유무, 2) TensorRT 최적 변환, 3) Quantization-FP16 경량화, 4) Deepstream 기반 실시간 스트림 데이터 처리 기법을 단독 또는 결합 적용하여 객체 식별 모델의 정확도 및 속도를 비교 평가하였다. 향후에는 Pruning 과 Knowledge Distillation 기법을 활용하여 작은 모델에서 큰 정확도 손실 없이 연산 속도를 개선할 수 있는 연구를 AGX Xavier, Jetson Nano, Raspberry Pi 4 등과 같은 다양한 엣지 기기 상에서 모색하고 신규 기법을 제안하고자 한다.

Acknowledgement

이 논문은 2022 년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. 2021-0-00907, 능동적 즉시 대응 및 빠른 학습이 가능한 적응형 경량 엣지 연동분석 기술개발)

참고문헌

[1] NVIDIA Jetson Nano, [Online] Available at https://developer.nvidia.com/embedded/jetson-nano-developer-kit.

[2] YOLOv5, [Online] Available at https://github.com/ultralytics/yolov5.

[3] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun, “YOLOX: Exceeding YOLO Series in 2021,” https://arxiv.org/abs/2107.08430.

[4] NVIDIA TensorRT, [Online] Available at https://developer.nvidia.com/tensorrt.

[5] NVIDIA Deepstream, [Online] Available at https://developer.nvidia.com/Deepstream-sdk.

[6] Xingyi Zhou, Dequan Wang, Philipp Krähenbühl, “Object as Points,” [Online] Available at https://arxiv.org/abs/1904.07850.

[7] VisDrone2019-DET Dataset, [Online] Available at http://aiskyeye.com/download/.