J. Internet Comput. Serv.
ISSN 1598-0170 (Print) / ISSN 2287-1136 (Online)
http://www.jics.or.kr

고속 해상 객체 분류를 위한 양자화 적용 기반 CNN 딥러닝 모델 성능 비교 분석☆

Comparative Analysis of CNN Deep Learning Model Performance Based on Quantification Application for High-Speed Marine Object Classification

이 성 주[1] 이 효 찬[2] 송 현 학[2] 전 호 석[2] 임 태 호[2*]
Seong-Ju Lee Hyo-Chan Lee Hyun-Hak Song Ho-Seok Jeon Tae-ho Im

요 약

최근 급속도로 성장하고 있는 인공지능 기술이 자율운항선박과 같은 해상 환경에서도 적용되기 시작하면서 디지털 영상에 특화된 CNN 기반의 모델을 적용하는 관련 연구가 활발히 진행되고 있다. 이러한 해상 서비스의 경우 인적 과실을 줄이기 위해 충돌 위험이 있는 부유물을 감지하거나 선박 내부의 화재 등 여러 가지 기술이 접목되기에 실시간 처리가 매우 중요하다. 그러나 기능이 추가될수록 프로세서의 제품 가격이 증가하는 문제가 존재해 소형 선박의 선주들에게는 비용적인 측면에서 부담이 된다. 또한 대형 선박의 경우 자율운항선박의 시스템을 감안할 때, 연산 속도의 성능 향상을 위해 복잡도가 높은 딥러닝 모델의 성능을 개선하는 방법이 필요하다. 따라서 본 논문에서는 딥러닝 모델에 경량화 기법을 적용해 정확도를 유지하면서 고속으로 처리할 수 있는 방법에 대해 제안한다. 먼저 해상 부유물 검출에 적합한 영상 전처리를 진행하여 효율적으로 CNN 기반 신경망 모델 입력에 영상 데이터가 전달될 수 있도록 하였다. 또한, 신경망 모델의 알고리즘 경량화 기법 중 하나인 학습 후 파라미터 양자화 기법을 적용하여 모델의 메모리 용량을 줄이면서 추론 부분의 처리 속도를 증가시켰다. 양자화 기법이 적용된 모델을 저전력 임베디드 보드에 적용시켜 정확도와 처리 속도를 사용하는 임베디드 성능을 고려하여 설계하는 방법을 제안한다. 제안하는 방법 중 정확도 손실이 제일 최소화되는 모델을 활용해 저전력 임베디드 보드에 비교하여 기존보다 최대 4~5배 처리 속도를 개선할 수 있었다.

☞ 주제어 : CNN, 모델 양자화, 영상 처리, 선박 객체 분류

ABSTRACT

As artificial intelligence(AI) technologies, which have made rapid growth recently, began to be applied to the marine environment such as ships, there have been active researches on the application of CNN-based models specialized for digital videos. In E-Navigation service, which is combined with various technologies to detect floating objects of clash risk to reduce human errors and prevent fires inside ships, real-time processing is of huge importance. More functions added, however, mean a need for high-performance processes, which raises prices and poses a cost burden on shipowners. This study thus set out to propose a method capable of processing information at a high rate while maintaining the accuracy by applying Quantization techniques of a deep learning model. First, videos were pre-processed fit for the detection of floating matters in the sea to ensure the efficient transmission of video data to the deep learning entry. Secondly, the quantization technique, one of lightweight techniques for a deep learning model, was applied to reduce the usage rate of memory and increase the processing speed. Finally, the proposed deep learning model to which video pre-processing and quantization were applied was applied to various embedded boards to measure its accuracy and processing speed and test its performance. The proposed method was able to reduce the usage of memory capacity four times and improve the processing speed about four to five times while maintaining the old accuracy of recognition.

☞ keyword : CNN, Model Quantization, Image pre-processing, Ship classification

1 Department of Ocean Convergence Technology, Hoseo University., Asan, 31499, Korea.
2 Information and communication, Hoseo University., Asan, 31499, Korea.
* Corresponding author (taehoim@hoseo.edu)
[Received 13 November 2020, Reviewed 7 December 2020(R2 21 January 2021), Accepted 16 February 2021]

☆ 이 논문은 2021년 해양수산부 재원으로 해양수산과학기술진흥원의 지원을 받아 수행된 연구임(ICT기반 수산자원관리 연구센터)
☆ "본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터지원사업의 연구결과로 수행되었음" (IITP-2021-2018-0-01417)
☆ 본 논문은 2020년도 한국인터넷정보학회 춘계학술발표대회 우수논문 추천에 따라 확장 및 수정된 논문임.

1. 서 론

자율운행선박(MASS)은 무인으로 선박을 운항하기 위하여 주변 환경 탐색, 위치 정보 및 제어 시스템을 포함하여 스스로 판단하고 제어하는 통합 시스템을 의미한다. 자율운항선박 시스템의 기술 연구가 활발히 진행되면서 인간 시각 정보를 대신하여 컴퓨터가 물체를 인식하고 검출하는 인공지능 기술에 관한 연구도 활발히 진행되고 있다. 기존에는 선박 주변의 상황 인지를 위해 레이더와 같은 센서를 이용하였으나, 최근에는 카메라와 정보 융합을 통해 주변 상황을 좀 더 면밀하게 분석하여 자율운항을 정밀하게 도울 수 있다[1].

해양수산부에서는 스마트 해양교통정책 추진전략을 통해 기존 해양사고의 80%가 넘는 인적 과실로 인해 발생하는 사고를 인공지능 기술을 활용하여 해상 내비게이션 서비스를 시행하여 선제적 예방을 하겠다고 발표하였다. 해상 네비게이션 서비스의 경우 인적 과실을 줄이기 위해 사람을 대체하는 기술을 활용하는데 충돌 위험이 있는 부유물을 감지하거나 선박 내부의 화재 등 다양한 기술이 접목되어야만 하므로 데이터가 실시간으로 처리되지 않는다면 선박 안전을 보장할 수 없다. 이를 해결하기 위해서는 프로세서의 성능을 높여 처리 속도를 높이는 방법이 있지만, 제품의 가격이 올라가는 단점이 있어 소형 선박을 운항하는 선주들에게는 비용적인 측면에서 부담이 된다. 이러한 문제를 해결하기 위해서는 프로세서의 성능을 높이지 않더라도 알고리즘의 성능은 비슷하게 유지하면서 처리 속도를 올릴 수 있는 소프트웨어를 설계하는 것이 매우 중요하다.

본 논문에서는 딥러닝 모델의 정확도를 유지하면서 고속으로 처리할 수 있는 방법에 대해 제안한다. 해상의 경우 비교적 주변 영상의 변화가 적고 대부분의 배경이 하늘 또는 바다와 같은 영역을 가지면서 객체를 검출하기 때문에 비교적 쉽게 찾을 수 있다. 이를 고려하여 해상 부유물 검출에 적합한 영상 전처리를 진행하여 효율적으로 딥러닝 입력에 영상 데이터가 전달될 수 있도록 하였다. 또한, 데이터가 고속으로 처리될 수 있도록 알고리즘 경량화 기법의 하나인 양자화 기법을 적용하여 메모리 사용률을 줄이면서 처리 속도를 증가시켰다. 마지막으로 데이터 전처리에 의해 들어가는 딥러닝 입력 채널을 줄이면서 파라미터값이 양자화 되었을 때 프로세서에서 효율적으로 소프트웨어가 동작하는지에 대한 평가를 진행하여 제시하는 연구를 진행하였다.

2. 관련 연구

2.1 데이터 셋 전처리

영상 인식에 강점을 보이는 CNN 기반의 모델에 대해 신경망을 구성하고 학습을 진행할 때 기존 연구에서는 모델의 정확도를 향상하기 위해서 데이터 셋의 양을 증가시키는 방식을 많이 사용하거나 질적으로 더 좋은 데이터를 가공하기 위하여 회전, 색상 보정과 같은 다양한 처리 기법을 적용하는 방식을 사용하였다. 하지만 학습에 사용될 데이터를 충분히 수집하지 않으면 정확도 향상이 어렵다는 단점이 존재하며 데이터의 명도 변화나 객체 간 겹침 현상과 같은 잡음으로 판단될 수 있는 환경 변화에 적응하지 못하기 때문에 고수준의 특징을 추출할 수 있는 데이터를 만들어내기 위해 모델 학습에 사용되는 데이터에 전처리를 사용하는 방식에 관한 연구가 지속해서 진행되었다.

본 논문에서 설계한 모델의 경우 해상 환경에서의 부유물을 분류해야 하므로 일반적인 카메라에서 입력으로 들어오는 데이터가 RGB의 3채널 데이터라면 비교적 불필요한 바다의 색상정보는 적게 저장하고 객체에 대한 형태를 보존하여 학습시키는 것이 정확도에 더욱 좋은 영향을 줄 수 있다. 따라서 하나의 채널로 채널 정보의 데이터를 감소시키는 방법을 적용시켜 불필요한 데이터의 양을 감소시키고 동일한 데이터 셋에 대해 각각의 전처리를 적용하였다. 먼저 24bit의 입력데이터를 8bit로 감소시키면서 비교적 해상에서 제일 많은 색상을 차지하는 바다 영역의 색상을 제거하였다. 또한 질적 향상을 위한 데이터를 생성하기 위해서 8bit color image, Gray scale을 포함하여 Histogram equalization, ZCA whitening과 같은 전처리 기법을 사용하였다.

2.2 딥러닝 모델 경량화

최근 급속도로 성장하고 있는 인공지능 기술은 실생활에 적용하려면 임베디드에서도 실시간으로 판단할 수 있는 수준에 도달하여야만 하므로 처리 속도 개선을 위한 연구가 활발히 진행되고 있다. 경량 딥러닝 연구는 크게 경량 알고리즘 연구와 알고리즘 경량화 연구로 나뉜다.

경량 알고리즘 연구의 경우는 모델 구조를 변경하는 방법과 합성곱 필터 변경, 자동 모델 탐색 등이 대표적인 기법이다. 모델 구조 변경 방식은 다양한 신규 계층 구조를 이용하여 파라미터 축소 및 모델의 성능을 개선하는

기법이며 SqueezeNet과 같은 네트워크가 이에 해당한다. 합성곱 필터를 변경하는 방식은 합성곱 신경망의 가장 큰 계산량을 요구하는 합성곱 필터의 연산을 효율적으로 줄이는 방식을 말하며 MobileNet과 같은 네트워크가 이에 해당한다. 자동 모델 탐색의 경우는 특정 요소(지연 시간, 에너지 소모 등)가 주어진 경우, 강화 학습을 통해 최적 모델을 자동 탐색하는 연구가 이에 해당한다.

알고리즘 경량화 연구의 경우는 모델 압축, 지식 증류, 하드웨어 가속화 등이 존재한다. 지식 증류는 새로운 모델 생성 시에 학습된 기본 모델을 통해서 생성된 파라미터값을 활용하여 학습 시간을 줄이는 연구를 말한다. 모델 압축의 경우는 객체를 분류하는 요소 중 하나인 가중치의 불필요한 값을 0으로 처리하는 가중치 가지치기와 비트를 제한하는 양자화 및 이마저도 0과 1로 표현하여 모델 용량을 확실하게 감소시키는 이진화 등이 이에 해당한다. 하드웨어 가속화 방식은 모바일 기기를 중심으로 뉴럴 프로세싱 유닛(NPU)을 통해 추론 속도를 향상시키는 기법을 의미한다[2].

본 논문에서는 모델의 크기를 압축시키면서 저전력 임베디드 실험 보드에서도 충분한 성능을 내기 위하여 기존 알고리즘 자체에서 경량화를 진행하는 방법으로 모델 양자화를 적용해 진행하였다. 알고리즘의 양자화 기술은 모델에서 객체를 분별하는 요소가 되는 파라미터 값이 객체를 분류할 수 있는 성능을 유지하는 조건에서 불필요한 가중치를 최대한 제거하는 방식이기 때문에 기존의 정확도를 위해 신경망의 깊이가 깊고 과파라미터화로 적용되어 있는 모델의 처리 속도를 개선하고 신경망 모델을 양자화를 적용해 변환하는 측면에서도 적합하다.

3. 본 론

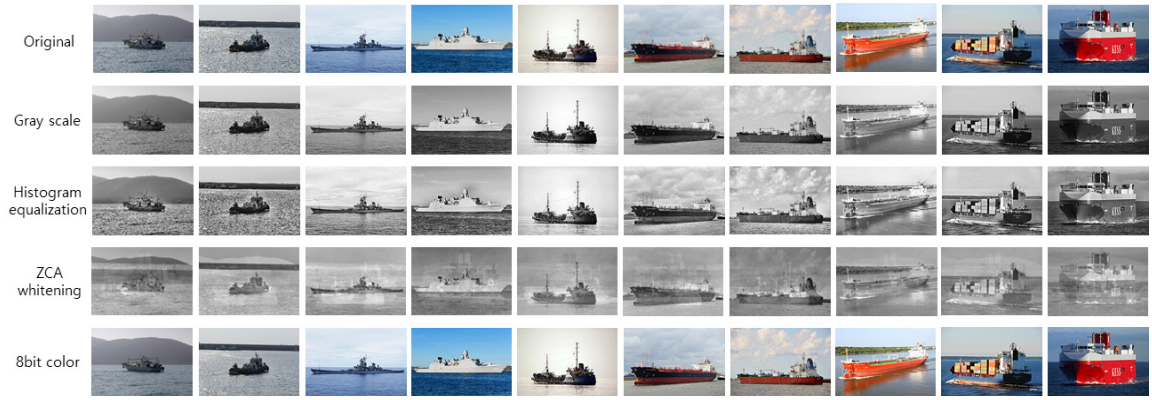
3.1 입력 데이터 전처리

CNN 모델에 대해 학습시킬 입력 데이터 전처리는 고수준의 특징을 추출하기 위해 진행된다. 그림 1에서는 전체 데이터 셋의 선박 이미지 중 10장을 선택하여 각각의 영상 처리 결과를 나타낸 것이며, Original 이미지를 제외한 전처리 데이터가 학습 데이터로 활용된다. 전처리 기법으로는 Gray scale과 이를 적용한 데이터에서 Adaptive histogram equalization, ZCA whitening 기법을 포함하여 RGB 24bit에서 8bit로 변환된 color image 4가지 방법으로 구성하였다.

gray scale 기법은 입력되는 이미지가 3채널(RGB)의 정보를 가지고 있다면, 광도의 정보만을 저장할 수 있게 처리하여 색채 정보를 제거하는 방식으로 사용한다.

histogram equalization 기법은 gray scale이 가지고 있는 각 픽셀의 광도의 정보 값을 histogram으로 나타내어 비트로 표현할 수 있는 전체 범위에 균등하게 분포시키는 방식이며 본 논문에서 사용되는 Adaptive 방식은 입력되는 이미지를 작은 블록 단위로 영역을 분할하여 histogram equalization을 적용하고, Contrast 값을 제한하여 각 영역에서의 noise 값의 증가를 제한하는 방식을 사용한다. 따라서 전체적인 이미지에서 평균을 취하는 것이 아니므로 사람의 시각으로 데이터를 확인하면 각 영역에서의 특징이 더 확연하게 보이는 효과를 가진다[3].

ZCA(Zero Component Analysis) whitening 기법은 입력되는 전체 데이터들의 모든 픽셀에서 평균을 구하여 각



(그림 1) 입력 데이터 전처리 전과 후
(Figure 1) Before & after preprocessing Input data

각의 이미지에 평균값을 제거한 필터를 생성한다[4]. 이렇게 생성되는 필터는 0 값을 중심으로 값이 생성되며, 필터에서 공분산 연산을 사용하여 인접한 픽셀에 대해 위상 스펙트럼과 이미지의 배열은 보존하는 상태로 백색화를 진행하여 주파수 스펙트럼을 평탄화 시킨다. 이러한 미백화를 적용하면 원본 이미지보다는 세부적인 부분의 데이터가 변하지만, 선박과 같은 전체적인 부분에서 경계선처럼 객체의 형태는 보존하게 된다[5].

마지막으로 8bit color image의 경우는 일반적으로 사용하는 24bit의 경우 3채널로써 RGB 각각의 채널이 8bit씩의 정보를 담고 있기 때문에 총 16,777,216가지의 색을 표현하지만 8bit color image로 변환하면 256가지의 색을 표현하는 이미지로 변환하게 된다. 이전 전처리 방식들은 8bit gary scale에서 전처리를 진행하지만 8bit color image의 방식의 경우 실제로는 8bit의 데이터로 학습이 진행되나 256가지의 색상 정보를 추출할 수 있어 바다 색상을 비교적 적게 저장하고 있을 수 있다는 장점이 있어 데이터 셋을 구성하였다. 이러한 특성을 이용하면 256가지의 색상도 사람의 눈에는 큰 차이가 없는 것처럼 보이며 원본 이미지와 유사하게 표현되므로 특징점은 살려내 CNN 기반 모델의 연산에서 좋은 결과를 얻을 수 있다.

본 연구에서는 이러한 전처리 방법을 사용해 데이터를 기존 한 픽셀당 24bit에서 8bit로 감소시키기 때문에 모델의 정확도에 큰 영향을 미칠 수 있는 입력 데이터에서 비교적 불필요한 데이터를 제거하여 데이터 용량이 1/3로 감소하며 신경망 모델을 학습시킬 경우에도 파라미터의 수가 줄어들기 때문에 연산량도 비례적으로 감소한다.

3.2 신경망 모델 구성 및 학습

CNN(Convolutional Neural Network)은 영상과 이미지처럼 화면을 구성하는 것에 가장 기본이 되는 단위인 픽셀에 대한 데이터를 처리하여 인식하는데 특화된 신경망으로써, 연산 구조 자체가 같지는 않으나 사람의 눈으로 들어오는 시각 정보를 처리하는 방식과 유사한 딥러닝 기반 네트워크이다.

모델 구성으로는 합성곱 신경망 모델을 사용한다. 구성하려는 모델은 해상 환경에서 선박과 같은 부유물을 검출하기에 적합한 모델이어야 한다. CNN 모델의 경우 모델의 깊이가 깊어질수록 정확도가 높아지는 것이 연구 결과로 이미 상당히 검증되었다. 하지만 학습에 사용되는 전처리 과정이 진행된 데이터 셋이 부유물을 검출하기 적합한 전처리 과정인지 검증하기 위해서 비교적 CNN 모델의 의존성이 낮을 수 있도록 LeNet과 같은 네트워크처럼 신경망을 설계해 실험에 적용하였다[6].

CNN 기반 모델의 정확도를 높게 가져가기 위해서는 학습을 진행할 모델 구성도 상당히 중요한 요소 중 하나가 되는데, 본 논문에서 구현한 신경망은 선박과 같은 해상환경에서 존재하는 부유물 분류에 사용될 계층들에 대해 각 계층의 파라미터 수를 포함하여 모델의 흐름대로 다음 표 1에 세부적으로 기술하였다.

(표 1) 신경망 모델 구성
(Table 1) Neural-Network model configuration

Layer	Output shape (Col, Row, Number of Channels)	Number of filter	Parameter
Input shape	(60, 90, 1)	-	0
Conv2D	(58, 88)	32	320
Conv2D	(56, 86)	32	9,248
Max-Pooling	(28, 43)	32	0
Conv2D	(26, 41)	64	18,496
Conv2D	(24, 39)	64	36,928
Max-Pooling	(12, 19)	64	0
flatten	14592	-	0
FC-layer	128	-	1,867,904
FC-layer	2	-	258
Total Parameter : 1,933,154			

해당 모델에서는 Convolution layer가 4개, Max-pooling layer가 2개, FC-layer의 개수를 2개로 구성하였다. 모델의 깊이가 깊지는 않으나 각 Convolution layer에서 합성곱 연산을 통해 도출되는 필터의 개수를 늘려 입력되는 데이터를 통해 특징점을 더 많이 저장할 수 있도록 조정하였다. 합성곱 필터 개수를 증가시킬 경우, Convolution layer에서는 특징점을 저장할 공간도 증가하지만, 모델의 총 연산량이 증가하고 파라미터의 수가 증가하게 된다.

특징점을 추출하는데 사용되는 필터 크기는 3x3이며 다음 계층에 전달되는 입력 데이터의 크기는 줄어들고 깊이가 깊어질수록 필터의 개수는 증가하게끔 진행하였다. 각각의 전처리 방법을 통하여 학습시키는 모델들은 입력 데이터가 8bit로 표현되어 기존의 RGB 3채널이 아닌 Input shape를 조정해 하나의 채널로 학습을 진행하였으며 이에 따라 총 파라미터의 수도 1/3로 줄어든다.

또한, 이에 따라 해당 모델에서의 필터를 통해 나오는 데이터를 1차원으로 나열한 후에 출력으로 연결해주는 FC-layer의 수를 2개 계층으로 지정하였다. 따라서 구성된 모델에서 파라미터가 존재하는 계층은 총 6개 계층이며, 하나의 파라미터마다 float32 값으로 학습이 진행되었다.

3.3 구성 모델 양자화

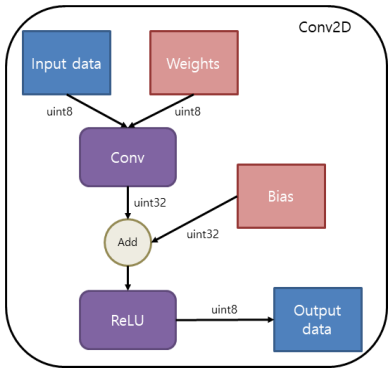
양자화는 샘플링에서 연속적인 데이터를 이산적인 데이터로 표현하는 것처럼 파라미터값을 설계하고자 하는 비트로 압축하는 것으로, 일반적인 CNN 기반의 모델은 가중치와 같은 파라미터가 부동 소수점으로 구성이 되어 있다. 양자화 기법은 모델의 이러한 부동 소수점의 비트를 줄여서 연산량을 감소시키는 방식이다. 일반적인 합성곱 신경망의 경우에는 Convolution layer, FC layer의 계층처럼 가중치와 편향 값을 연산해야 하는 계층에서 파라미터값을 가지게 되는데 이러한 파라미터의 값들은 일반적인 모델에서 float32의 자료형에 저장되게 된다. 부동 소수점으로 구성되는 파라미터들은 일반적으로 모델이 가지고 있는 가중치에서 그 값이 작을 경우에는 모델에 정확도에 대해서 큰 영향을 주지는 않는다. 따라서 이러한 양자화 기법을 신경망 모델에 적용할 때는 정확도에 큰 영향을 주지 않는 선에서 모델의 경량화를 진행하여야 저전력 임베디드에 적용하는 데 의미가 있다.

모델에서 생성되는 이러한 파라미터 값은 0에 근사하게 정규화 과정을 거쳐 실질적인 객체를 분류하기 위한 값으로 데이터가 저장되게 되는데, 모델의 계층이 깊어지고 각 계층에서 파라미터의 비트가 많아질수록 연산량이 급격하게 늘어나게 된다. 즉, float32라는 자료형에 weights와 bias 값이 저장되는데 이러한 값들의 bit를 조절함으로써 메모리 용량을 절약하고 연산속도는 증가시킬 수 있다는 것이 양자화의 장점이다. 특히 영상과 같은 픽셀 데이터 처리 기반의 CNN과 같은 신경망 모델은 입력 데이터의 크기가 크거나 필터의 수가 많을수록 합성곱 계층에서의 데이터 연산량이 상당히 높아 임베디드 실험 보드에서 실시간적인 처리가 어려운데 양자화를 적용한다면 한 파라미터의 값마다 데이터 용량이 감소함으로써 저전력 임베디드와 같은 하드웨어에서 속도와 메모리 용량에서 이점을 볼 수 있게 된다.

본 연구에서는 학습을 진행하는 것은 float32에서 진행하고 결과값으로 추출되는 파라미터 값들에 대해 양자화를 적용하는 학습 후 양자화 기법으로 진행하였으며, 이러한 방법은 파라미터의 수가 많을수록 비교적 정확도 하락 폭이 작아지는 특성이 있다.

양자화는 좀 더 세밀하게 표현할 수 있는 float32의 값들을 비교적 일정한 크기의 단위로 재배치하는 것이므로, 8bit로 양자화를 진행할 때 overflow 현상이 발생하면 객체를 정확하게 인식하지 못할 수 있으므로 이를 고려해 범위를 조절하여 압축을 진행하여야 한다.

float32에서 uint8의 형태로 양자화를 적용하는 경우에서 Convolution layer 하나의 계층에서 보면 다음 계층으로 데이터를 전달하기까지의 전체적인 흐름은 그림 3과 같다.



(그림 3) uint8의 양자화 흐름도
(Figure 3) Quantization technique of uint8

실질적인 합성곱 연산에 사용되는 입력 데이터와 weights 값들이 0~255의 수로 표현되는 uint8로 데이터가 변환되고, 결과 값은 uint32에 저장되게 된다. 그 후 bias 값이 더해지게 된다. 이 bias 값의 경우 너무 작은 비트의 값으로 압축을 시킬 경우 정확도 하락에 큰 영향을 미칠 수 있다. 그 이유는 다음 계층에 출력값을 전달해줘야 하는데 객체의 클래스를 보다 정확하게 맞추기 위해서 활성화 함수를 사용하여 다음 계층에게 전달해줘야 한다. 이 과정에서 bias 값에 대한 정보는 weights의 데이터보다는 비교적 세밀하게 표현이 유지되어야만 활성화 함수의 그래프 이동이 가능해 객체를 분류하는 데 중요한 역할을 하므로 bias 값은 32bit로 유지하되 weights와 입력 데이터의 결과값과 연산을 위해 uint32로 저장한다.

이렇게 더해진 weights와 bias 값은 다시 uint8로 변환하여 Convolution layer 계층의 마지막 단계인 활성화 함수를 통해 출력되는 데이터가 다음 계층에게 입력 데이터로 전달되게 된다[8].

이렇게 출력되는 데이터는 각 계층에서의 피처 맵(feature map)으로 정의되며 필터 연산을 통해 추출된 특징점들을 저장할 수 있는 공간이 된다.

아래 표 2는 각 모델을 비교하기 위해 실험에 사용될 자료형들로 기존 모델의 자료형과 모델의 총 크기를 포함하고 각 전처리 방법으로 학습시킨 모델의 크기를 모델의 각 비트에 따라 평균화하여 간략히 나타낸 것이다.

각 파라미터마다 float32로 저장되었던 모델의 크기는 22,715kb였으며, 해당 신경망 모델에서 weights만 8bit로 고정 소수점 동적 범위 양자화를 진행하면 7,557kb의 용량으로 진행된다. 그리고 나머지 값도 8bit의 완전한 정수 값으로 양자화하면 메모리 공간을 절약할 수 있기에 보다 효율적인 모델을 설계할 수 있다. 따라서 32bit에서 8bit까지 압축을 진행한 경우에는 약 4배까지 용량을 압축시킬 수 있었다.

(표 2) 비트별 학습된 모델의 크기
(Table 2) Size of trained model by bit type

모 델	평균 모델 크기 (kb)
float32	7,557
float16	3,783
int8	1,902
uint8	1,902

3.4 실험 결과

3.4.1 실험 환경

본 논문에서 제안한 전처리 및 양자화를 적용하기 위해 사용된 임베디드 보드는 Raspberry 3 & 4, Coral dev board를 활용하였다. Raspberry Pi는 영국에서 개발한 소형 임베디드 플랫폼으로 1GHz 이상의 64bit 기반 쿼드코어 ARM 프로세서를 탑재하였다. Raspberry의 경우는 3b+ 모델과 4b를 사용하였으며, Raspberry에서는 Coral usb accelerator가 함께 사용되었다. Coral dev board는 구글에서 2019년 초에 새롭게 출시한 모델로, Edge 단에서 AI를 실행하는데 최적화 설계가 되어 있는 저전력 임베디드 실험 보드이다. Edge TPU는 Dev board에 내장되어 있으며 Coral usb accelerator와 함께 고정 소수점 연산에 특화되어 있어 비교적 적은 비트의 처리 속도를 증가시킬 수 있는 장점이 있다. 표 3은 본 연구에서 실험에 사용된 임베디드 보드의 사양을 간략하게 나타낸 것이다.

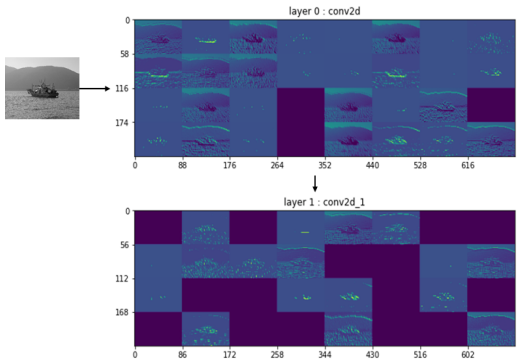
(표 3) 임베디드 모듈 사양
(Table 3) Embedded module specifications

종 류	CPU	GPU/TPU	RAM
Raspberry 3 b+	1.4GHz ARM Cortex-A53	Broadcom VideoCore IV	1GB LPDDR2
Raspberry 4 b	1.5GHz ARM Cortex-A72	Broadcom VideoCore IV	4GB LPDDR4
Coral Dev board	NXP i.MX 8M SoC (quad Cortex-A53, Cortex-M4F)	Integrated GC7000 Lite Graphics	1GB LPDDR4

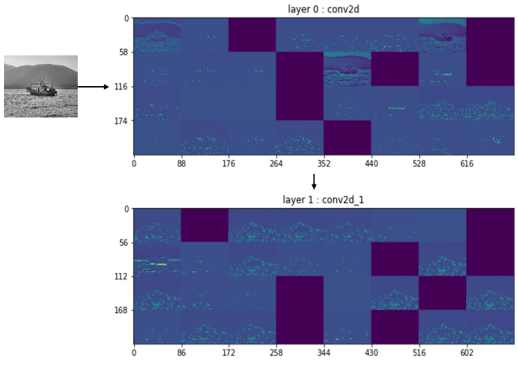
3.4.2 영상 데이터 전처리 및 CNN 학습

CNN 기반 딥러닝 모델에서의 필터 개수를 증가시켜 특징점을 저장할 공간을 확보하여 영상 데이터의 전처리 기법을 비교하였으며 이를 합성곱 계층의 연산 결과인 feature map을 시각화하여 확인하였다.

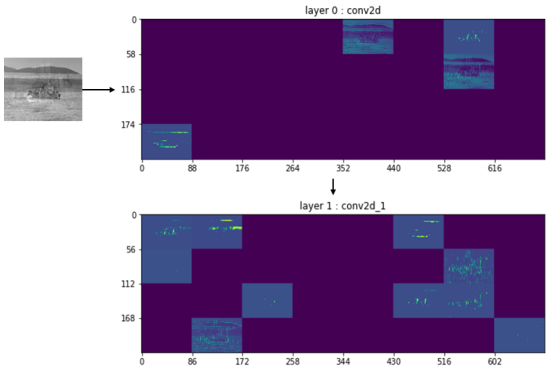
그림 4~7의 경우는 전처리가 진행된 각각의 데이터 셋을 학습시킨 모델들을 사용하여 추론을 진행하는 과정을 나타낸 것이며 1~2번째 합성곱 연산이 이뤄지는 Conv2D 계층의 파라미터 값들의 결과를 선박 이미지 한 장을 선정하여 입력하였을 때의 feature map을 시각화한 것이다. x축은 너비 y축은 높이를 의미하며 첫 번째 입력 데이터의 크기가 가로 90픽셀 세로 60픽셀일 경우에 합성곱 연산을 3x3 크기의 필터를 사용하여 진행하고 난 후 가장자리에 있는 1픽셀 크기가 감소해 하나의 다음 계층에 입력 데이터로 활용될 feature map의 크기가 88x58로 표현된다.



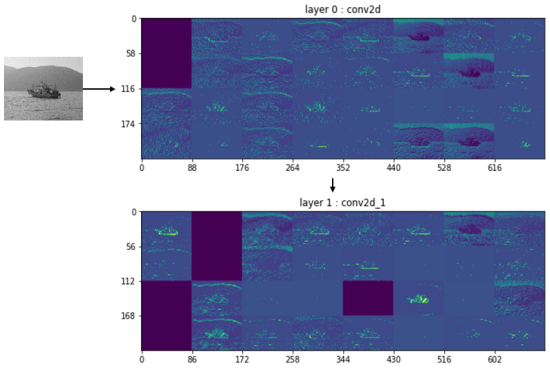
(그림 4) grayscale 모델의 피처맵
(Figure 4) Feature maps in grayscale



(그림 5) Histogram 모델의 피처맵
(Figure 5) Feature maps in Histogram equalization



(그림 6) ZCA 모델의 피처맵
(Figure 6) Feature maps in ZCA whitening



(그림 7) 8bit color 모델의 피처맵
(Figure 7) Feature maps in 8bit color image

1~2번째 합성곱 계층의 필터 개수는 32개이므로, 그에 해당하는 필터의 값을 확인할 수 있다. 다른 feature map에 비해 비교적 어둡고 형태가 나타나지 않은 맵들은 입력 데이터에 대해서 필터값 추출이 되지 않은 feature map들이다. 즉, 해당 맵이 반응하지 않았다는 것을 의미한다. 그림 6의 ZCA의 경우 첫 번째 계층에서의 feature map이 다른 전처리 기법으로 학습한 모델에 비해 활성화가 되지 않은 것으로 보이며 8bit image가 제일 많은 필터가 활성화된 것으로 확인하였다. 이러한 feature map을 정확도에 좋은 영향을 미쳤는지에 대한 결과를 유추하는 데 활용하였다. 학습에서의 실험 데이터는 약 1만 장으로, 이 중 약 75%는 CNN 모델의 학습용 데이터로 사용하고 나머지 25%는 검증용 데이터로 사용하였다. 학습을 진행할 때 각 모델의 Input size는 90x60으로 진행하였으며, epoch는 50번 학습시켰다.

3.4.3 CNN 추론 및 양자화 결과

제안하는 전처리 기술을 적용한 CNN 기반의 딥러닝 모델을 양자화 진행하였을 때 정확도를 유지하는지 확인하기 위하여 각 임베디드 실험 보드에서 float32부터 uint8까지의 양자화된 모델을 평가하였다. 또한 실시간으로 연산 처리 속도를 평가하기 위해 카메라로 영상 데이터를 입력받아 높이, 너비, 채널 형식의 3차원 배열로 재조합하여 추론에 사용되는 모델의 첫 번째 계층의 입력으로 연결하고 전반적인 처리 속도를 확인하였다.

표 4의 경우는 비트별로 양자화를 적용하였을 때, 검증용 데이터를 사용하여 각 모델에 나타난 정확도를 기존 신경망 모델인 float32를 포함하여 나타낸 것이며, 데이터 셋에 HIstogram equalization을 적용한 모델이 고정 소수점으로 양자화를 적용한 경우에서도 정확도의 손실이 거의 없이 가장 높은 성능을 나타냈다. ZCA 처리 기법을 int8로 양자화를 진행하였을 때에는 정확도가 상당히 낮아지는 현상이 보였는데 이는 모델이 학습할 때 float32에서는 0을 중심으로 부동 소수점을 정규화하고 필터 연산을 통해 그림 6에서와 같이 feature map의 활성화가 되지 않은 이미지들의 데이터가 음수 표현이 가능한 int8 양자화를 진행하게 되면서 파라미터를 표현이 유사한 상태로 압축되어 정확도 하락이 높았을 것으로 예상된다.

표 5의 경우는 Coral Dev Board CPU에서 전처리 기법의 연산 부분을 제외하고 각 모델의 추론 부분에서 이미

(표 4) CNN 기반 딥러닝 모델의 정확도(%)
(Table 4) Accuracy of CNN-based deep learning models (%)

모 델	Gray scale	Histogram	ZCA	8bit Color
float32	96.46	99.88	96.26	97.16
float16	96.46	99.88	96.22	97.16
int8	91.21	83.54	36.36	96.44
uint8	96.62	99.88	96.22	97.16

(표 5) Coral Dev Board CPU 환경에서의 딥러닝 모델 처리 속도 (sec)
(Table 5) Deep Learning Model Processing Speed in Coral Dev Board CPU Environment (sec)

모 델	Gray scale	Histogram	ZCA	8bit Color
float32/16	0.024	0.024	0.024	0.024
int8/uint8	0.033	0.033	0.033	0.033

지 한 장의 연산 처리 속도를 비교한 표이다. 비트별로 비교해보았을 때, 파라미터의 개수가 동일해 비트마다 압축되는 수도 같았기 때문에 추론 속도에서도 비트에 따라 동일한 성능을 보여주는 것을 확인할 수 있다.

표 6에서는 정확도가 제일 높았던 Histogram의 전처리 방법을 사용하여 uint8로 변환한 딥러닝 모델을 고정 소수점을 연산하기에 최적화되어 있는 전용 하드웨어 가속기를 적용하였을 때 각 임베디드 실험 보드에서 카메라를 통해 입력된 데이터를 한 프레임마다 전처리 기법을 적용하고 신경망 모델에 입력 데이터로 전달하여 추론하는 부분까지의 전반적인 시스템 처리 속도를 비교한 표이다. 각 임베디드 실험 보드에서 연산 처리 속도는 4~5배의 향상된 성능을 보였으며 정확도는 비트별 모델 정확도와 동일하였다. CNN 기반 딥러닝 모델에서 필터 개수가 많아 파라미터의 수가 약 190만 개가 존재하지만 상당히 높은 처리 속도를 보이는 것을 확인할 수 있었다.

(표 6) 각 임베디드 보드의 처리 속도 가속기 적용 비교 (sec)
(Table 6) Comparison of processing speed accelerator application of each Embedded board (sec)

모 델	Raspberry 3 b+	Raspberry 4 b	Coral Dev Board
CPU int8/uint8	0.098	0.048	0.033
TPU int8/uint8	0.057	0.008	0.006

4. 결 론

본 논문은 딥러닝 CNN 모델을 해상 환경에서 실시간으로 객체 인식을 하기 위해 전처리 및 양자화를 적용하였는데 전처리를 통해 입력 채널을 줄이고 양자화를 적용하면서 정확도 손실을 최소화하는 방법을 제안하였다.

입력되는 전처리 방법으로는 Gray Scale, Histogram equalization, ZCA whitening, 8bit color image 등 다양하게 적용하였고 그 중 Histogram equalization 방식이 정확도가 가장 높았다. 해상의 경우 대부분의 배경이 하늘 또는 바다와 같은 저주파 영역을 가지면서 Histogram equalization 방식의 특징을 적용하면 해상 배경과 부유물의 경계선을 더 명확하게 분리하기 때문에 더욱 강인한 입력 데이터로 전처리가 된 것을 실험으로 확인할 수 있었다. 해상 부유물을 더욱 정확하게 인식하기 위해서는 CNN 모델 층을 깊게 쌓아야 한다. 하지만 임베디드 보드에서 실시간으로 처리하기에는 메모리 용량 문제 및 처리속도의 문제점이 있다. 따라서 본 논문에서는 양자화를 적용하여 정확도의 손실이 거의 없이 처리하면서 기존에 양자화를 적용하지 않은 상태인 float32로 연산이 진행되는 CNN 모델보다 처리 속도를 최대 4~5배 정도 개선할 수 있었다.

현재 연구에서는 Tensorflow를 활용하여 양자화를 진행하였는데 지원되는 타입은 float 32,16으로 소프트웨어를 지원하고 Coral Accelerator는 (u)int 8비트로 고정소수점 형태로 가속을 진행하였다. 향후 FPGA에서 8비트 이하로 고정소수점을 지원할 수 있도록 하드웨어를 설계하여 임베디드 보드에서 저전력 및 고성능이 지원되는 해상 전용 딥러닝 모델을 개발할 예정이다.

참고문헌(Reference)

[1] S. Li and K. S. Fung, “Maritime autonomous surface ships (MASS): implementation and legal issues,” Maritime Business Review, Vol. 4, No. 4, pp. 330-339, Nov. 2019.
https://doi.org/10.1108/MABR-01-2019-0006

[2] Y. J. Lee, Y. H. Moon, J. Y. Park, and O. G. Min, “Recent R&D Trends for Lightweight Deep Learning” Electronics and Telecommunications Trends, Vol. 34, No. 2, pp. 40-50, Apr. 2019.
https://www.doi.org/10.22648/ETRI.2019.J.340205

[3] C. Cui, X. Wang, and H. Shen, “Improving the face recognition system by hybrid image preprocessing,” in 2016 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2016.
https://www.doi.org/10.1109/CYBER.2016.7574866

[4] K. K. Pal and K. S. Sudeep, ″Preprocessing for Image Classification by Convolutional Neural Networks″, in 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2016.
https://www.doi.org/10.1109/RTEICT.2016.7808140

[5] A. J. Bell and T. J. Sejnowski, “The ‘independent components’ of natural scenes are edge filters,” Vision Research, Vol. 37, No. 23, pp. 3327-3338, Dec. 1997.
https://doi.org/10.1016/S0042-6989(97)00121-1

[6] J. S. Lee, S. K. Lee, D. W. Kim, S. J. Hong, and S. I. Yang, “Trends on Object Detection Techniques Based on Deep Learning”, Electronics and Telecommunications

Trends, Vol. 33, No. 4, pp. 23-32, Aug. 2018.
https://www.doi.org/10.22648/ETRI.2018.J.330403

[7] S. W. Lee, G. D. Lee, J. G. Ko, S. J. Lee, and W. Y. Yoo, “Recent Trends of Object and Scene Recognition Technologies for Mobile/Embedded Devices,” Electronics and Telecommunications Trends, Vol. 34, No. 6, pp. 133-144, Dec. 2019.
https://doi.org/10.22648/ETRI.2019.J.340612

[8] B. Jacob et al., “Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference,” in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.
https://www.doi.org/10.1109/CVPR.2018.00286

[9] H. H. Song, H. C. Lee, S. J. Lee, H. S. Jeon and T. h. Im, “Design of Video Pre-processing Algorithm for High-speed Processing of Maritime Object Detection System and Deep Learning based Integrated System”, Journal of Internet Computing and Services (JICS), Vol. 21, No. 4, pp. 117-126, 2020.
http://dx.doi.org/10.7472/jksii.2020.21.4.117

[10] J. h. Chae, H. Y. Ko, B. B. Lee and N. G. Kim, “A Study on the Pipe Position Estimation in GPR Images Using Deep Learning Based Convolutional Neural Network”, Journal of Internet Computing and Services (JICS), Vol. 20, No. 4, pp. 39-46, 2020.
https://dx.doi.org/10.7472/jksii.2019.20.4.39

[11] S. Y. Kahu and K. M. Bhurchandi, ″JPEG-based Variable Block-Size Image Compression using CIE La*b* Color Space″, KSII Transactions on Internet and Information Systems, Vol. 12, No. 10, pp. 5056-5078, 2018.
http://doi.org/10.3837/tiis.2018.10.023

[12] L. Tan, D. Xuan, J. Xia and C. Wang, ″Weather Recognition Based on 3C-CNN,″ KSII Transactions on Internet and Information Systems, vol. 14, no. 8, pp. 3567-3582, 2020.
http://doi.org/10.3837/tiis.2020.08.024.

◐ 저 자 소 개 ◑

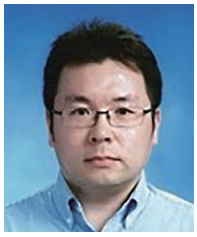
이 성 주(Seong-ju Lee)
2019년 호서대학교 정보통신공학과(공학사)
2019년~현재 호서대학교 정보통신공학과(공학석사)
관심분야 : 딥러닝, 컴퓨터 비전
E-mail : sjlee3416@naver.com

이 효 찬(Hyo-Chan Lee)
2014년 호서대학교 정보통신공학과(공학사)
2016년 호서대학교 대학원 정보통신공학과(공학석사)
2019년 호서대학교 대학원 정보통신공학과(공학박사)
2019~현재 호서대학교 해양IT융합기술연구소 연구원
관심분야 : 임베디드 시스템 설계, 영상 신호 처리, 딥러닝
E-mail : lhc_104@naver.com

◐ 저 자 소 개 ◑

송 현 학(Hyun-Hak Song)
2019년 호서대학교 정보통신공학과(공학사)
2019년~현재 호서대학교 정보통신공학과(공학석사)
관심분야 : 컴퓨터 비전, 인공지능
E-mail : rainy_930@naver.com

전 호 석(Ho-Seok Jeon)
2020년 호서대학교 정보통신공학과(공학사)
2020년~현재 호서대학교 정보통신공학과(공학석사)
관심분야 : 하드웨어, 영상 인식
E-mail : wjsghtjr33@naver.com



임 태 호(Tae-Ho Im)
2012년 중앙대학교 대학원 전자전기공학과(공학박사)
2012년~2015 삼성전자 DMC연구소 책임연구원
2015년~현재 호서대학교 해양IT융합기술연구소 조교수
2019년~현재 호서대학교 정보통신공학과 조교수
관심분야 : 5G 이동통신, LPWAN, 수중통신, 딥러닝
E-mail : taehoim@hoseo.edu