

특징 피라미드 네트워크의 특징 압축을 위한 레벨 간 특징 예측 방법

김민섭, 이종석, 심동규*

광운대학교

{minsub20, suk2080, *dgsim}@kw.ac.kr

Inter-level Feature Prediction Network for Feature Pyramid Network Feature Compression

Kim Min Sub, Lee Jong Seok, Sim Dong Gyu*

Kwangwoon Univ.

요약

본 논문은 FPN(Feature Pyramid Network) 기반 네트워크에서 추출된 특징맵을 압축하는 구조를 제안한다. FPN구조의 네트워크는 해상도별 특징맵이 추출되기 때문에 일반적으로 영상에 비해 데이터가 크기 때문에, 영상에 대한 압축에 비해 더 높은 압축율을 요구한다. 이에 본 논문은 모든 해상도의 특징맵 전체를 압축하지 않고, 심층 신경망을 통해, 낮은 해상도의 특징맵으로 높은 해상도 특징맵을 예측한 뒤 압축을 진행하는 방법을 제안한다. 제안한 방법의 결과는 압축해야 할 대상의 데이터양은 75% 감소함과 동시에, 객체 검출 성능(mAP)은 모든 특징맵을 보내는 방법 대비, 0.2% 감소했다.

I. 서론

최근 이미지, 비디오에서 객체 검출, 객체 추적, 객체 영역 분할을 수행하기 위한 심층 신경망에는 FPN[1]구조가 널리 사용되고 있다. FPN의 기본 구조는 그림 1 과 같이 영상에서 해상도별 특징맵을 추출하는 backbone을 포함한 FPN부분과 특징맵을 입력으로 넣어 객체 검출을 수행하는 detector head 부분으로 이루어져 있고, FPN의 기본 구조를 기반으로 한 객체 검출 네트워크들은 추가적인 변화를 주어 성능 향상을 위한 연구들 진행되고 있다.[2][3][4]

다른 한편으로는 심층 신경망들을 모바일 또는 에지 디바이스에서 사용하려는 시도가 있다.[5] 하지만 모바일이나 에지 디바이스에서 사용하기에는 객체 검출 신경망이 너무 많은 계산량을 필요로 한다는 문제점이 있다. 이를 해결하기 위해서 FPN 과정 까지는 높은 컴퓨팅 파워를 가지는 서버를 사용하고, FPN에서 출력된 특징맵을 에지 디바이스에 전달하여 객체 위치나 크기와 같은 객체 탐지 결과를 에지 디바이스에서 추론하는 방법에 대한 연구가 진행되고 있다. [6]

하지만, FPN에서 나온 결과물인 특징맵은 일반적으로 입력 이미지보다 더 많은 채널 수를 가지며, FPN구조상 다수개의 특징맵을 출력하기 때문에 입력대비 많은 데이터양을 가지고 있다. 큰 데이터양은 통신 네트워크 환경 부하가 생겨, 지연과 같은 문제점이 발생할 수있다. 따라서, 본 논문에서는 FPN에서 출력 된 다수개의 특징맵의 데이터양을 감소시키기 위하

여 낮은 해상도의 특징맵을 이용하여 높은 해상도의 특징맵을 예측하는, FPN의 특징 압축을 위한 레벨 간 특징 예측 방법을 제안한다. 제안하는 방법은 높은 해상도의 특징맵을 학습된 신경망을 통해 예측하여 생성함으로써 해당 해상도의 특징맵에 대한 전송을 생략하여 전체적으로 전송되는 데이터양을 75% 감소시킬 수 있으며, 객체 검출 성능은 모든 해상도의 특징맵을 보냈을 경우 대비 0.2% 감소했다. 본 논문의 구성은 다음과 같다. 2장에서는 본 논문에서 제안한 방법에 대해서 자세히 설명하고, 3장에서는 제안하는 방법의 성능을 평가하고 결론을 맺는다.

II. 본론

본 논문에서는 FPN구조의 네트워크에서 출력 된 특징맵 중, 하위 해상도의 특징맵을 입력으로 상위 해상도의 특징맵을 예측한 뒤 압축을 진행하는 방법을 제안한다. 본 논문에서 사용한 FPN의 구조의 네트워크에서 출력된 특징맵은 P₂, P₃, P₄, P₅와 P₅를 max pooling으로 down sampling한 P₆로 이루어져있다. 그림 2는 본 논문에서 제안하는 구조로, Encoder와 Decoder로 구성되어있다. Encoder는 FPN에서 출력 된 특징맵에서 가장 높은 해상도의 특징맵인 P₂를 제외한 P₃, P₄, P₅, P₆를 압축해서 Decoder로 전송한다. Decoder는 전송받은 압축된 특징맵을 복원 후, 가장 낮은 해상도 특징맵은 P₆를 제외한 P₃, P₄, P₅를 본 논문에서 제안한 네트워크에 입력하여 예측된 P₂인 P₂'를 생성한다. 마지막으로, P₂', P₃', P₄', P₅', P₆'를 detector head에 입력하여 객체 검출을 수행한다. 제안한 네트워크가 P₂를 예측하는 과정은 다음과 같다. 입력받은 특징맵들을 P₂의 해상도(H,W,C)에 맞춰 보간 한 후, 채널 축 방향으로 이어붙인다. 이때 보간방법은 최단인접 보간법(Nearest neighbor interpolation)을 사용한다. 이후, 합성곱 층 과 GDN(Generalized Divisive Normalization)[7] 층을 통해 예측한 P₂'가 출력된다. GDN의 하이퍼 파라미터로 beta를 1e-6, gamma를 0.1, offset을 2의 -18승을 사용했다.

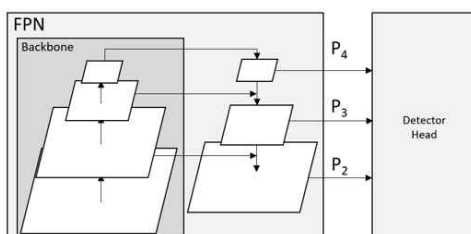


그림 1. FPN기반 객체 검출 네트워크의 기본구조

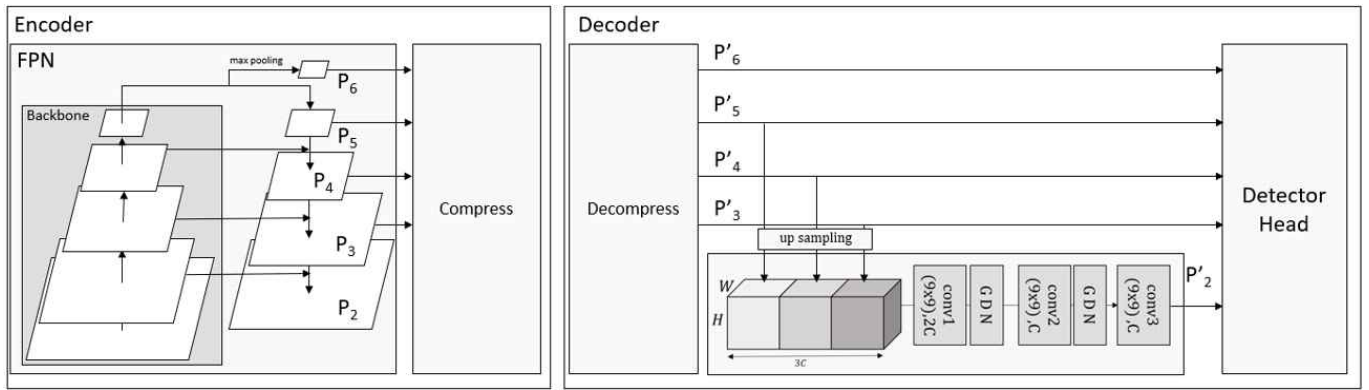


그림 2. 제안하는 특징맵 압축 구조

III. 실험환경 및 실험결과

본 논문에서 제안하는 방법의 성능을 평가하기 위하여 제안하는 특징 피라미드 네트워크의 특징 압축을 위한 레벨 간 특징 예측 방법을 python 3.7.9기반의 pytorch 1.7.1 딥러닝 프레임 워크를 이용하여 구현했다. 제안하는 방법의 객관적 압축 성능 평가를 위해 FPN에서 나온 특징맵을 선형 양자화 및 SBMP 기반 CABAC[8]을 사용하여 압축한 뒤, 압축된 데이터의 크기 및 복원된 특징맵을 사용한 객체 검출 정확도를 모든 특징맵을 보내는 경우(*Anchor*), 그리고 P3를 P2의 해상도로 최단입점 보간법을 적용한 것을 P2로 사용하는 경우(*Nearest neighbor interpolation*)와 비교했다. 객체 검출 정확도는 *AP-S*, *AP-M*, *AP-L*, *mAP*에 대해서 측정하며 이는 각각 소형, 중형, 대형에 대한 객체 검출 성능과 평균적인 객체 검출 성능을 의미한다. 객체 검출 정확도 비교를 위한 네트워크는 detectron2[9]를 사용하여 구현했다. 구현한 네트워크는 ResNeXt-101(32x8d)[10]을 FPN의 backbone으로 사용했으며, 특징맵을 입력받아 객체 검출을 수행하는 detector head는 Faster R-CNN[11]의 detector head 부분을 사용했다.

제안하는 네트워크를 학습시키기 위하여 COCO data set[12]의 train2017에서 10000장을 training set으로 사용하고, val2017에서 5000장을 validation set으로 사용했다. 성능 비교를 위한 Test set으로는 train2017에서 training set을 제외한 5000장을 사용했다.

표 1. 제안한 방법의 성능 비교

	data size(bit)	mAP(%)	AP-S(%)	AP-M(%)	AP-L(%)
<i>Anchor</i>	3773 k	57.595	37.755	64.244	73.022
<i>Nearest neighbor interpolation</i>	848 k	52.073	25.242	63.272	73.018
<i>Proposed</i>	848 k	57.387	36.429	63.961	73.023

표1에서 제안한 방법은 모든 특징맵을 보내는 방법(*Anchor*) 대비 75% 정도의 bit 감소를 보임과 동시에, 객체 검출 성능(*mAP*(mean Average Precision))은 0.2% 하락했다. 특히, 소형 물체 탐지 성능을 나타내는 *AP-S*에서의 성능 하락 정도가 크게 줄었음을 확인 할 수 있다.

IV. 결론

본 논문에서는 FPN 구조의 네트워크에서 하위 해상도의 특징맵을 입력으로 상위 해상도의 특징맵을 예측한 뒤 압축을 진행하는 방법을 제안했다. 본 논문에서 제안한 방법은 압축 할 대상의 총 데이터양을 75% 감소 시킴과 동시에, 객체 검출 성능(*mAP*)은 모든 특징맵을 보내는 방법 대비, 0.2% 감소했다.

ACKNOWLEDGMENT

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학 ICT 연구센터지업사원(IITP-2021-2016-0-00288)의 연구결과로 수행되었음

참고 문헌

- [1] Lin, T. Y. "Feature Pyramid Networks for Object Detection," Proc.CVPR, pp. 2117-2125, Jul. 2017.
- [2] Ghiasi G. "Nas-fpn: Learning scalable feature pyramid architecture for object detection," Proc.CVPR, pp. 7036-7045, Jun. 2019.
- [3] Li, Zeming "Detnet: A backbone network for object detection," arXiv preprint arXiv:1804.06215, 2018.
- [4] Gong, Y. "Effective fusion factor in FPN for tiny object detection," Proc.CVPR, pp. 1160-1168, Jun. 2021.
- [5] Robert J. Wang "PELEE: A Real-Time Object Detection System on Mobile Devices," arXiv preprint arXiv:1804.06882, 2018.
- [6] Ren, J. "Distributed and efficient object detection in edge computing: Challenges and solutions," IEEE Network 32.6, pp. 137-143, 2018.
- [7] Balle J. "End-to-End Optimized Image Compression," Proc.ICLR, Apr. 2017.
- [8] Haase P. "State-based multi-parameter probability estimation for context-based adaptive binary arithmetic coding," Proc.DCC, pp. 163-172, Mar. 2020.
- [9] Wu Y. "Detectron2," 2019, (<https://github.com/facebookresearch/detectron2>).
- [10] Saining, X. "Aggregated Residual Transformations for Deep Neural Networks," Proc.CVPR, pp. 1492-1500, Jul. 2017.
- [11] Shaoqing, R. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," NIPS, pp. 91-99, Dec. 2015.
- [12] Lin, T. Y. "Microsoft COCO: Common Objects in Context," ECCV, pp. 740-755, 2014.