모바일/임베디드 객체 및 장면 인식 기술 동향

Recent Trends of Object and Scene Recognition Technologies for Mobile/Embedded Devices

이수웅 (S.W. Lee, suwoong@etri.re.kr) 콘텐츠인식연구실 선임연구원
이근동 (G.D. Lee, zacurr@etri.re.kr) 콘텐츠인식연구실 선임연구원
고종국 (J.G. Ko, jgko@etri.re.kr) 콘텐츠인식연구실 책임연구원
이승재 (S.J. Lee, seungjlee@etri.re.kr) 콘텐츠인식연구실 책임연구원
유원영 (W.Y. Yoo, zero2@etri.re.kr) 콘텐츠인식연구실 책임연구원/실장

ABSTRACT

Although deep learning-based visual image recognition technology has evolved rapidly, most of the commonly used methods focus solely on recognition accuracy. However, the demand for low latency and low power consuming image recognition with an acceptable accuracy is rising for practical applications in edge devices. For example, most Internet of Things (IoT) devices have a low computing power requiring more pragmatic use of these technologies; in addition, drones or smartphones have limited battery capacity again requiring practical applications that take this into consideration. Furthermore, some people do not prefer that central servers process their private images, as is required by high performance server-based recognition technologies. To address these demands, the object and scene recognition technologies for mobile/embedded devices that enable optimized neural networks to operate in mobile and embedded environments are gaining attention. In this report, we briefly summarize the recent trends and issues of object and scene recognition technologies for mobile and embedded devices.

KEYWORDS 경량 딥러닝, 모바일 딥러닝, 객체 검출, 장면 분할

Ⅰ. 서론

최근 딥러닝 기술의 발전으로 객체 및 장면 인식 등 영상 분석 기술의 성능이 비약적으로 향상되었으며, 이를 활용한 산업적 응용이 주목받고 있다.

객체 및 장면 인식 기술은 이미지 또는 비디오에서 사람, 물체, 장면 등의 세부적 시각 정보를

* DOI: https://doi.org/10.22648/ETRI.2019.J.340612
* 이 논문은 2019년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No.2018-0-00198, 객체 추출 및 실-가상 정합 지원 모바일 AR 기술 개발).

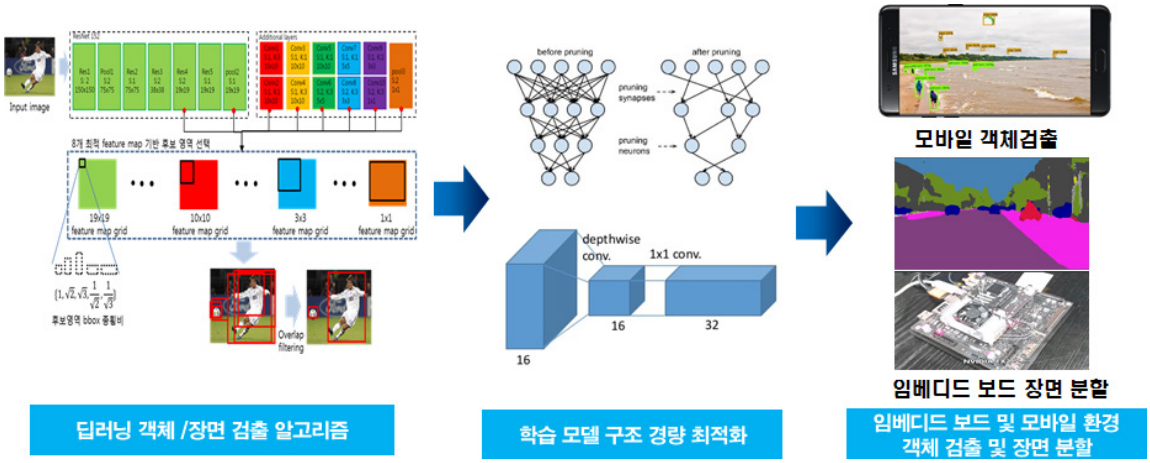


그림 1 딥러닝 기반 경량 객체 검출 및 장면 분할 개요

딥러닝 기술을 통해 마치 사람이 보는 것과 같이 추출하는 기술이다. 객체 및 장면 인식 기술은 딥러닝을 통해 비약적으로 발전하였으나, 대부분의 기술은 고성능 GPU 서버 환경을 기반으로 하고 있다.

최근에는 응용 분야 및 서비스 영역의 확장을 위해 모바일 및 임베디드의 엣지 컴퓨팅 환경에서 구동 가능한 영상 분석 기술의 수요가 증가하고 있다. 구체적으로 IoT 기기와 같은 저사양 기기, 드론이나 스마트폰 등 배터리의 제약이 있는 기기, 인터넷 연결이 지원되지 않는 서비스 환경에서 영상 분석 기술에 관한 수요가 존재하며, 개인 정보 보호나 익명화의 이슈 또한 엣지 환경에서의 영상 분석 기술을 요구한다.

이러한 요구사항들에 맞추어 최근 모바일/임베디드 환경에서 동작 가능한 경량 딥러닝 기술에 대한 개발이 수행되고 있다. 딥러닝 모델의 효율적 레이어 설계를 통한 모델 경량화와 Pruning, Quantization, Depthwise separable convolution 등을 통한 모델 최적화 등의 연구가 이루어지고 있다(그림 1 참조).

본 고에서는 최근 이슈화된 모바일/임베디드 환경에서의 객체 및 장면 인식 기술 동향 및 관련 이슈들에 대해 살펴보고자 한다.

II. 기술 현황

모바일/임베디드 환경에서 딥러닝 모델을 이용한 객체 검출과 장면 분할을 위해서는 낮은 응답 지연 시간(Low latency)과 저전력 그리고 연산량의 최적화가 필요하며, 이를 위해서는 다음의 요구사항이 만족되어야 한다.

첫째, 딥러닝 모델에서 영상 특징 추출을 위한 기본 구조로 활용되는 특징 추출 레이어(Feature extraction layer)의 경량화가 필요하다.

둘째, 객체 검출 혹은 객체/장면 분할을 위한 딥러닝 모델 구조 관점에서의 최적화가 필요하다. 예를 들어 객체 검출의 경우 객체의 위치와 종류를 결정하기 위한 구조의 경량화가 필요하며, 객체/장면 분할을 위해서는 분할을 위한 구조의 경량화가 필요하다.

본 장에서는 경량화를 위한 특징 추출 구조, 객체 검출을 위한 경량화 및 객체/장면 분할을 위한 경량화 기술에 대해 살펴본다.

1. 객체 검출 기술

딥러닝 기반의 객체 검출 기술은 그 구조에 따라 크게 Two-stage 방식과 Single-stage 방식으로 구분된다. Two-stage 방식은 Selective Search[1], Region Proposal Network[2] 등으로 이미지에서 객체 후보 영역(ROI: Region of Interest)을 찾는 단계와 찾은 후보들에 대해 클래스 분류 및 Bounding Box Regression 작업을 수행하는 단계, 총 2단계를 거쳐 객체 검출을 수행한다. 대표적인 기술로는 Faster R-CNN[2]이 있다. Single-stage 방식은 객체 후보 영역을 찾는 단계 없이 미리 정의된 Anchor Box로부터 분류와 Bounding Box Regression을 바로 수행하는 하나의 딥러닝 모델로 객체 검출을 수행한다. 모바일 및 임베디드 분야에서는 구동 속도 문제로 주로 Single-stage 방식의 방법들이 사용되며, 이 절에서도 Single-stage 방식의 방법들을 살펴본다.

가. SSD(Single-Shot multibox Detector)

SSD[3]는 Single-stage 객체 검출 기술의 일종으로 VGG[4] 등 특징 추출 레이어(Feature extraction layer)의 후속 파트에 연결되어 객체의 위치와 종류를 판별한다. SSD에서는 많이 발견되는 객체 모양과 유사한 default box를 정의하고, ground truth box와 default box와의 차이를 학습하며, 다양한 해상도를 갖는 여러 단계의 특징맵(Feature map)에 연결되어 다양한 크기의 물체를 찾아낼 수 있다.

SSDLite[5]는 구글에서 개발한 SSD의 개량형으로 SSD의 모든 conv layer를 depthwise conv와 1×1 conv로 변환한 모델이다. SSDLite는 약간의 변화로 SSD에 비해 파라미터 수는 86%가량, 계산량은 72%가량 크게 줄인다.

나. YOLO(You Only Look Once)

YOLO[6-8] 역시 Single-stage 객체 검출 기술의 일종으로 하나의 영역에 최대 두 개의 물체만 존재한다고 가정하기 때문에 SSD와 비교하여 상당히 적은 수의 박스만을 사용하고 구조 또한 더 단순하므로 계산량이 상대적으로 적고 공식 코드가 C++로 되어 있어 사용하기 간편한 점이 있어 많은 연구자가 사용하고 있다.

YOLO는 계속해서 발전하였는데, v2[7]에서는 batch norm, anchor box, Darknet19 등 다양한 방법을 적용하여 성능을 올렸고, 함께 발표한 YOLO9000에서는 9,000개의 클래스를 검출할 수 있는 객체 검출기를 발표하였다. v3[8]에서는 Bounding box regression의 성능이 높아지도록 관련 파라미터들을 개선하고, SSD와 유사하게 여러 스케일의 특징맵을 최종 검출 레이어로 연결하였으며, 특징 추출 레이어에 residual connection을 추가하여 더 성능을 높였다. YOLO는 기본 모델과 함께 성능을 조금 포기하더라도 계산량을 크게 줄인 tiny 모델을 함께 배포하여 사용자 선택의 폭을 넓혔다.

2. 경량 객체 검출 기술

가. 경량 영상 특징 추출 기술

객체 검출에서 가장 많은 계산이 소요되는 부분이 영상의 특징을 추출하는 영역이고, 따라서 이 영역의 경량화를 위한 많은 시도가 있었다. 대표적으로 depthwise separable conv 연산을 사용하는 MobileNet[5,9,10], channel shuffle을 통해 적은 계산량으로 다양한 채널의 정보를 활용하는 ShuffleNet[11,12], 텐서의 수축과 확장으로 유효한 정보를 효율적으로 골라내는 SqueezeNet[13], 그리고 단순한 shift 연산으로 spatial conv 연산을 대체하는

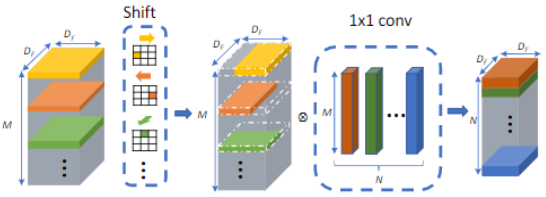
Shift[14] 등을 들 수 있다. 이들 중, 이 장에서는 텐서플로우와 함께 가장 대중적으로 활용되고 있는 MobileNet과 zero-FLOP으로 최근 주목받고 있는 Shift에 대해 중점적으로 살펴본다.

1) MobileNet

MobileNet[5,9,10]은 구글에서 모바일 딥러닝 서비스를 위해 개발한 경량 딥러닝 모델이며, 2017년에 v1[9]이, 2018년에 v2[5]가, 2019년에 v3[10]가 발표되었다. MobileNet v1에서는 3×3 conv 연산을 3×3 dw-conv(depthwise conv)와 1×1 conv 연산으로 대체하여 FLOP(FLoting-point OPeration) 기준 8~9배의 계산량 감소를 끌어내면서도 기존 딥러닝 모델과 비슷한 성능을 유지하였다. 또한 정확도(Accuracy)와 FLOP의 trade-off를 조절할 수 있는 두 가지 하이퍼 파라미터를 제공하여 사용자가 목적에 맞는 딥러닝 모델을 손쉽게 구성하여 사용할 수 있도록 하였다. MobileNet v2에서는 depthwise separable conv 외에 bottleneck 형태로 딥러닝 모델을 구성하고 short-cut connection을 추가하여 성능을 높였다. MobileNet v3에서는 최근 유행하는 Platform-aware NAS(Neural Architecture Search) 방법으로 자동으로 딥러닝 모델의 구조를 찾고 그 후 NetAdapt[15] 알고리즘을 적용하여 타깃 리소스에 맞게 자동으로 압축하는 과정을 거쳐 새로운 딥러닝 모델을 구성하였다.

2) Shift

Spatial Convolution 연산은 LeNet[16], AlexNet[17] 등 초기의 영상 특징 추출 딥러닝 모델부터 지금까지 빠지지 않던 연산이지만, 계산량이 필터 크기의 제곱에 비례하기 때문에 연산량이 높은 단점이 있었다. VGG[4]에서 3×3과 1×1 conv만을 사용하여 상당한 성과를 거둔 이후 딥러닝 모델에서는 대부분 3×3과 1×1 conv를 사용하고 있으며, MobileNet에서는 3×3 conv layer를 3×3 dw-conv와 1×1 conv의 조합으로 변경하여 연산량을 크게 개선하였다. Shift[14]는 여기에서 한 걸음 더 나아가 3×3 dw-conv 레이어를 shift operation 레이어로 변경하였다. MobileNet에서 3×3 dw-conv 레이어는 이미지 공간(spatial) 방향으로 정보를 혼합하고 1×1 conv 레이어는 채널 방향으로 정보를 혼합하는데, Shift에서는 spatial 방향으로 정보를 혼합하기 위해 3×3 dw-conv 대신 FLOP이 없는 shift operation을 이용하여 zero FLOP, zero parameter의 spatial convolution의 대체재를 개발하였다(그림 2 참조).



출처 Reprinted with Permission from https://arxiv.org/abs/1711.08141

그림 2 Shift

Shift에 관한 여러 후속 연구들이 있었다. Jeon은 shift parameter를 뉴럴넷이 학습하도록 하는 Active shift layer[18]를 제안하였으며, Chen은 전체 채널 중 매우 적은 수에 대해서만 shift를 수행하여도 높은 정확도를 얻을 수 있는 Sparse shift layer[19]를 제안하였다.

나. 모바일/임베디드 경량 객체 검출 기술

이 절에서는 모바일/임베디드를 위한 경량 객체 검출 기술을 다룬다. 앞 절의 경량 영상 특징 추출 기술들과는 적은 계산량, 빠른 응답 속도를 추구한다는 점에서는 같지만 본 절에서는 모바일이나 임베디드를 타깃으로 한 기술들을 다룬다. 이 기술들은 계산량을 FLOP 대신 실제 디바이스에 대한

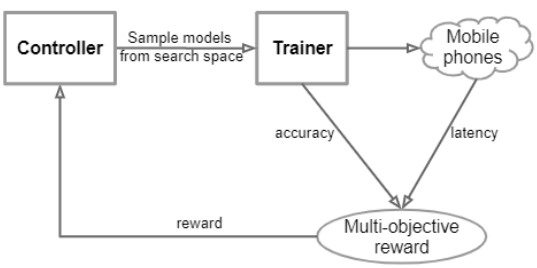
응답 지연 시간(latency)으로 주로 측정하기 때문에 platform-aware한 특징이 존재한다. 또한 많은 방법이 객체 검출 레이어보다는 앞단의 특징 추출 레이어의 계산량을 줄이는 데 집중한다.

1) MnasNet

MnasNet[20]은 Neural Architecture Search 방법의 일종으로 FLOP 대신 타깃 디바이스(예, 모바일)에서의 응답 지연 시간을 줄이기 위한 목적으로 설계되었다. MnasNet에서는 정확도와 응답 지연 시간을 모두 고려하기 위한 지수형태의 목적 함수(Objective function)를 제안하였고, 강화 학습을 통해 딥러닝 모델을 학습시켰으며, 이때 실제 응답 지연 시간을 reward로 사용하였다. MnasNet은 파라미터의 크기, 정확도와 응답 지연 시간 관점에서 기존의 manual search 및 automatic search 방법을 크게 능가하여 주목받고 있다(그림 3 참조).

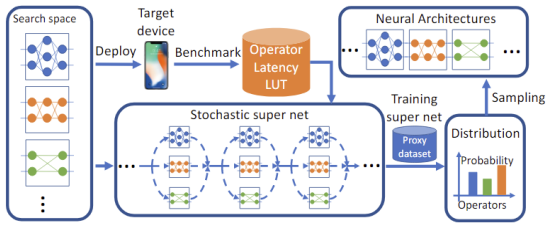
2) FBNet

FBNet[21]은 Neural Architecture Search 방법으로 찾아낸 경량 딥러닝 모델이다. Search space를 22개의 layer와 9개의 cell로 상당히 크게 구성하여 표현력을 보장하였고, 손실 함수(Loss function)에 타깃 디바이스의 응답 지연 시간을 넣어 실제 구동 시간의 최적화를 보장하였다. 또한 여러 differential re-



출처 Reprinted with Permission from https://arxiv.org/abs/1807.11626v3

그림 3 MnasNet 동작 방식



출처 Reprinted with Permission from https://arxiv.org/abs/1812.03443

그림 4 FBNet

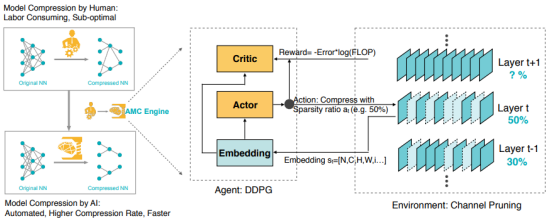
laxation 방법을 통해 강화 학습이 아닌, SGD 방법으로 학습하여 학습 시간을 강화 학습 기반의 방법에 비해 수백 배 줄이는 효과도 보였다. 실험 결과 정확도와 응답 지연 시간 모두 경쟁 딥러닝 모델에 비해 상당히 좋아진 것을 확인할 수 있었다(그림 4 참조).

3) AMC(AutoML for Model Compression)

AMC[22] 역시 최근의 흐름대로 강화 학습을 활용하여 자동으로 경량 딥러닝 모델을 찾아가는 알고리즘이다. AMC의 저자들은 경량 딥러닝 모델을 찾기 위해 많은 엔지니어와 GPU 자원을 사용한다는 것에 주목하고 이를 자동으로 해결하는 데 목표를 두었다. 제안한 agent인 DDPG가 목표한 sparsity만큼 압축을 수행하며, reward는 FLOP으로 주어진다. AMC는 범용적인 모델(MobileNet, ResNet50)과 데이터셋(ImageNet)을 실험에 사용하여 보다 상용화에 가깝다고 할 수 있다(그림 5 참조).

4) Pelee

Pelee[23]는 최근 많이 사용하는 depthwise conv layer 대신 일반 conv layer만을 사용한 구현으로 보다 더 범용적으로 사용할 수 있는 장점이 있다. Depthwise conv layer를 제거하면서도 계산량을 줄이기 위해 Pelee에서는 Two-way Dense Layer와



출처 Reprinted with Permission from https://arxiv.org/abs/1802.03494

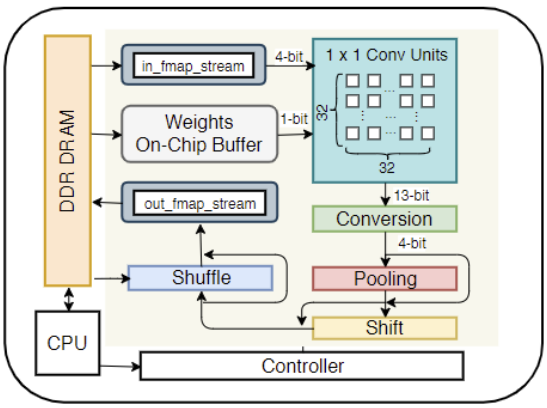
그림 5 AMC

Stem Block이라는 새로운 딥러닝 모델 구조를 제안하였다.

특히 Pelee는 SSD와 결합하여 객체 검출 모델을 구성하였는데, 이 딥러닝 모델 성능이 iPhone8 기준 23.6FPS로 실시간 성능을 내고 있다.

5) FPGA

최근 딥러닝 인식 기술을 FPGA에 구현한 사례들도 발표되고 있다[24,25]. Algorithm-Hardware Co-design이라는 컨셉으로 연구가 수행되고 있는데, 즉 기존의 연구들이 정확도에만 집중하거나 계산량에만 관심을 둔다는 문제점을 지적하며 연산이 일어날 FPGA의 특성에 맞는 효율적인 알고리즘과 하드웨어의 설계에 관한 연구를 동시에 진행하고 있다.



출처 Reprinted with Permission from https://arxiv.org/abs/1811.08634

그림 6 Synetgy Accelerator 구조

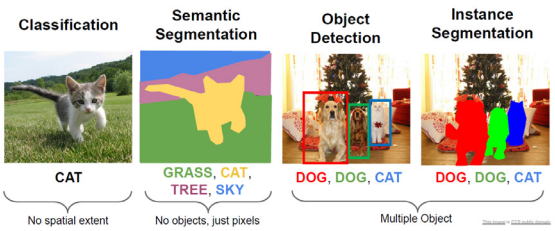
Synetgy[24]에서는 FPGA의 특성을 분석하고 하드웨어 친화적인 연산셋을 사용하며, 양자화(Quantization)가 잘 되면서 정확도가 크게 저하 되지 않는 ShuffleNet-V2[12] 기반의 DiracDeltaNet이라는 4-bit 딥러닝 모델을 제안하고 이를 ImageNet으로 학습하여 Xilinx 하드웨어에서 기존 딥러닝 모델과 유사한 정확도로 16.9배 이상으로 속도를 개선하였다(그림 6 참조).

Lightweight YOLOv2[25]에서는 FPGA를 이용한 객체 검출 모델 구현이 발표되었다. 이 연구에서는 YOLOv2를 이진화(Binarization)하여 연산을 줄이고 이를 Xilinx FPGA보드에 올리기 위한 구현 방법이 기술되어 있으며, frame rate와 소비 전력으로 측정한 efficiency가 embedded GPU 대비 42.9배 향상되었다.

6) eSSD(efficient SSD)

ETRI에서도 모바일/임베디드를 위한 경량 객체 검출 기술 연구를 지속하고 있으며, low latency, low power를 지향하는 MobileNet-v1 SSD의 개량형인 eSSD를 개발하였다[26]. eSSD는 객체 검출 구조의 최적화, MobileNet v1을 개량한 영상 특징 추출 레이어의 최적화, 그리고 NMS(Non-Maximum Suppression), batch 등 시스템 레벨의 최적화를 통해 에너지와 성능이 밸런스를 갖는 효율적인 딥러닝 모델을 구현하였다. eSSD는 SSDLite에 비해 추가 영상 특징 추출 레이어(additional feature extraction layer)의 병목(bottleneck) 구조를 제거하고 prediction layer를 1×1 conv만을 이용하여 효율적으로 구성한 것이 특징으로 SSDLite 대비 모델 사이즈가 조금 늘어나지만 응답 지연 시간이 줄어들어 전체적 성능을 높였다.

그림 7 ETRI DeepMobileAR 객체 검출



출처 CC0 public domain. http://cs231n.stanford.edu/slides/2019/cs231n_2019_lecture12.pdf

그림 8 의미론적 분할 및 인스턴스 분할 비교

또한, ETRI에서는 개발된 경량 객체 검출 모델을 모바일폰에 포팅하여 모바일 환경에서 객체 검출을 수행하는 DeepMobileAR이라는 서비스를 개발하였다. 그림 7은 휴대폰에서 ETRI 경량 객체 검출 모델의 동작 결과를 보여준다(그림 7 참조).

또한 이 결과물로 ETRI에서는 LPIRC2018(Low Power Image Recognition Challenge)라는 국제 경쟁 대회에 참가하여 track 3의 우승을 차지한 바 있다[27]. 최종 스코어는 전년 1위와 동일한 하드웨어를 사용하였음에도 4배가량 상승하였다.

LPIRC는 기존 챌린지에서 중시했던 검출 정확도 성능 외에 소비 전력을 함께 측정하여 스코어를 매기기 때문에 저전력과 고성능 두 가지 모두를 달성해야 하는 챌린지이면서 실제 산업 적용에 한 발 더 다가선 챌린지라고도 볼 수 있다.

3. 객체/장면 분할 기술

영상 내의 객체 및 장면 분할 기술은 분할 대상과 기술에 따라 분류된다. 먼저, 분할 대상은 객체와 stuff 영역으로 구분되며, 객체는 사람, 자동차, 동물 등 셀 수 있는 물체를 의미하며, stuff 영역은 하늘, 바다, 도로 등 형태가 정해지지 않고 셀 수 없는 영역을 의미한다[28].

객체 및 장면 분할 기술은 기술적으로 의미론적 분할(Semantic Segmentation), 인스턴스 단위 분할(Instance-level Segmentation), Panoptic 분할(Panoptic Segmentation)[29]로 분류된다.

의미론적 분할은 이미지 내의 각 픽셀을 클래스 레이블 단위로 분류하는 것으로, 인스턴스(Instance) 단위로 구분 짓지 않는다. 데이터셋에 따라 객체만을 대상으로 분할하거나[30,31], 객체와 stuff 영역을 포함한 장면 전체에 대해 모두 분할하기도 한다[31-34].

인스턴스 단위 분할은 이미지 내 각 객체를 인스턴스 단위로 구분지어 분할하는 것으로, 의미론적 분할과 달리 동일한 클래스에 속한 객체일지라도 서로 구분되어야 한다(그림 8 참조).

Panoptic 분할은 의미론적 분할과 인스턴스 단위 분할이 결합된 형태로, 기본적으로 이미지 내의 각 픽셀을 클래스 레이블 단위로 분류하면서, 객체에 대해서는 인스턴스 단위로 구분 짓는다.

가. 의미론적 분할 기술

의미론적 분할 기술은 픽셀의 클래스 레이블링 문제이기 때문에 영역 단위로 처리하는 객체 검출, 인스턴스 단위 분할과 달리 모든 픽셀을 한 번에 예측할 수 있는 FCN(Fully Conv Net) 기반 모델들이 주류를 이룬다. FCN 기반 모델의 경우 pooling

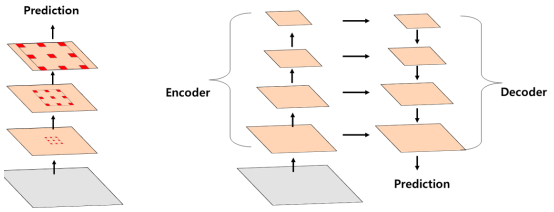


그림 9 의미론적 분할 모델 구조 비교

및 convolution stride로 인해 특징맵의 공간 해상도가 저하되는 문제가 있는데, 이는 입력 영상의 각 픽셀 단위로 클래스를 레이블링해야 하는 의미론적 분할 모델의 성능을 크게 좌우한다. 이를 개선하기 위해 skip layer, deconvolution, unpooling 기반 upsampling 등을 적용하여 저하된 특징맵의 공간 해상도를 다시 확대시키는 모델들이[35,36] 제안되었다. 최근에는 특징맵의 공간 해상도 문제를 해결하는 동시에 multi-scale context를 활용하여 성능을 개선하는 dilated FCN과 인코더-디코더 네트워크 두 가지 방향으로 주로 연구가 이루어지고 있다(그림 9 참조).

Dilated FCN[37-40]은 해상도의 손실, 파라미터의 증가 없이 특징의 수용영역(Receptive field)을 확장시킬 수 있는 dilated(atrous) convolution을 FCN에 적용한 모델이다. PSPNet(Pyramid Scene Parsing Net)[39]과 DeepLabv2[40]는 dilated FCN을 기반으로 Spatial Pyramid Pooling[41]을 적용하여 다양한 스케일의 맥락(Context) 정보를 추가함으로써 정확도를 개선하였다.

Dilated FCN은 정확도 개선에는 효과적이지만, dilated convolution으로 인해 특징맵의 크기가 기존 FCN에 비해 커지면서 메모리와 연산량이 늘어나는 단점이 있다.

인코더-디코더 네트워크의 인코더 파트는 FC(Fully Connected) 레이어가 제거된 ImageNet 분류 모델을 사용하여 점진적으로 특징맵을 down-sampling하면서 고수준의 의미론적 특징을 추출한다. 디코더 파트는 인코더의 중간 레이어의 특징을 이용하여 점진적으로 특징맵을 upsampling하면서 정교한 분할 외곽선을 얻을 수 있게 한다. 대표적인 기술로는 U-Net[42], SegNet[43], RefineNet[44] 등이 있다. 인코더-디코더 네트워크는 dilated convolution이 필요 없기 때문에 빠른 연산에 적합하며, Ⅱ장 5절에서 소개될 ENet[45], DeepLabv3+[46] 등 경량 모델에 대한 연구가 진행되고 있다.

나. 인스턴스 단위 분할 기술

객체를 인스턴스 단위로 구분해야 하기 때문에 인스턴스 단위 분할 기술은 객체 검출 기술을 기반으로 발전되었다.

Mask R-CNN[47]는 Faster R-CNN[2]의 기존 객체 검출 레이어에 객체 분할 레이어를 병렬 구조로 추가하여 객체 검출 모델을 객체 분할 모델로 확장시켰다. 객체 분할 레이어는 단순한 fully conv net으로 구성되어 있기 때문에 연산량은 크게 증가되지 않았다. 또한 기존 classification 및 bounding box regression loss에 mask loss를 추가하여 multi-task learning을 하고, ROIPool 레이어를 Feature Pooling 과정의 양자화 에러 문제를 개선한 ROIAlign 레이어로 대체하여 객체 검출 및 분할 성능을 개선시켰다. 이 모델은 객체 검출 성능 개선을 위해 객체 검출 레이어를 cascade 구조로 구성한 Cascade R-CNN[48]과 결합되어 Cascade Mask R-CNN으로 발전되었다.

HTC(Hybrid Task Cascade for Instance Segmentation)[49] 모델은 Cascade Mask R-CNN을 기반으로 병렬 연결되었던 객체 검출 레이어와 객체 분할 레이어를 교차 배열 구조로 변경하고, 각 stage

에서 추출된 Mask Feature를 연결하여 information flow를 개선하였다. 또한 의미론적 분할 레이어를 객체 검출 레이어와 객체 분할 레이어에 연결하여 COCO-stuff[28] 데이터셋을 이용하여 모델을 훈련시켰다. 이로 인해 객체와 stuff 영역을 구분하는 공간적 맥락(Spatial Context)이 특징에 인코딩되어 객체 검출 및 분할 성능이 개선되었다. 이 HTC 모델을 기반으로 홍콩중문대, Sensetime 등으로 구성된 MMdet 팀은 2018 COCO Challenge[50]의 인스턴스 분할 트랙에서 우승을 차지하였다.

다. Panoptic 분할 기술

Panoptic 분할 기술은 2018년 COCO, Mapillary Challenge[50]가 열리면서 연구가 진행되었다. 현재까지는 의미론적 분할 모델과 인스턴스 단위 분할 모델의 결과를 결합하는 형태로 연구가 진행되고 있다.

4. 경량 객체/장면 의미론적 분할 기술

객체 및 장면 분할 기술 중 인스턴스 단위 분할과 Panoptic 분할 기술은 현재까지는 모델의 경량화보다는 정확도를 개선시키는 방향으로 연구가 진행되고 있다. RetinaMask[51] 모델은 경량 객체 검출 모델인 RetinaNet[52]에 객체 분할 레이어를 추가하였으나, 이는 Multitask learning을 통한 객체 검출 성능 개선을 위한 것으로, 인스턴스 단위 분할에 대해서는 자세히 다루지 않았다.

반면 의미론적 분할 기술의 경우, 모바일 환경에서의 응용을 고려한 고속, 경량화에 대한 연구가 진행되고 있다. 특히 dilated FCN에 비해 연산량이 상대적으로 적은 인코더-디코더 네트워크 기반 모델의 인코더 파트에 경량 특징 추출 모델을 사용하여 연산량을 효율화하는 방법들이 제안되었다.

가. ENet

ENet[45]는 크게 입력영상에서 정보를 축약해 가는 Encoder 부분과 축약된 정보에서 다시 Upsample 과정을 통해 Decoder하는 부분들로 구성된다. 기존 모델보다 모델구조를 최적화하였는데, Bottlenet(1×1, 3×3, 1×1)으로 구성되는 모델구조에서 3×3의 채널수를 32개 이하로 최소화하여 처리량을 줄였다.

나. DeepLab

구글에서 2015년에 atrous conv 및 VGG16 기반의 DeepLabv1[37]를 발표하고, 이어서 개선된 버전인 DeeLabv2[40]를 발표하였는데, ResNet-101 기반 특징맵에 다른 rate를 갖는 atrous conv를 병렬연결한 ASPP(Atrous Spatial Pyramid Pooling) 모듈과 Bi-linear Interpolation 기반 특징맵 확장을 사용하여 성능을 향상시켰다. DeepLabv3[53]는 ASPP 모듈에 batch normalization을 추가하고, 글로벌 컨텍스트를 포함하는 이미지레벨 특징과 융합하여 정확도를 향상시켰다. 최근에는 dilated

표 1 ENet 구조

Layer	Type	Output size
Initial		16x256x356
bottleneck1.0	downsampling	64x128x128
bottleneck2.0~ bottleneck2.8	downsampling dilated 2, 4, 8, 16	128x64x64
Repeat section 2, without bottleneck2.0		
bottleneck4.0~ bottleneck4.2	upsampling	64x128x128
bottleneck5.0 bottleneck5.1	upsampling	16x256x256
fullconv		Cx512x512

FCN과 인코더-디코더 네트워크 구조를 결합한 DeepLabv3+[46]을 발표하였다. Dilated FCN인 DeepLabv3를 인코더로 사용하여 다양한 스케일의 컨텍스트 정보를 포함하도록 개선하고, 간단한 디코더 모듈을 사용하여 정교한 분할 외곽선을 얻어 정확도를 개선시켰다. 또한 특징 추출 모델을 ResNet에서 Xception으로 교체하고, atrous separable convolution을 ASPP와 디코더 모듈에 적용하여 연산량을 감소시켰다. 최근에는 경량 특징 추출 모델인 MobileNet과 결합하여 모바일 디바이스에서 구동할 수 있는 분할 모델들[54,55]이 공개되었다.

다. RENet

ETRI는 ENet을 기반으로 conv3 레이어들의 특징맵 다운사이징을 통해 더 견고한 함축적 특징정보 사용 및 conv5 채널수 변경을 통한 처리 속도 향상을 고려한 Robust & Efficient 모델을 제안하였는데, 기존 ENet 모델 대비 20% 처리속도 향상을 보였다. 휴대폰에서 실내 장면 영역(바닥, 벽, 테이블 등)과 실외 장면 영역(도로, 나무, 건물, 하늘 등) 분할을 처리 수행한다.

그림 10 실내외 장면영역 분할(ETRI)

III. 결론

기존 고성능 GPU 서버 환경에서의 객체 검출 및 장면 분할의 영상 분석 기술 개발에서 최근 모바일/임베디드 환경의 영상 분석 기술의 요구사항이 증가하고 있어, 이를 위해 저사양의 환경에서 동작 가능한 다양한 형태의 딥러닝 모델 개발 연구가 활발히 이루어지고 있다.

향후 영상 분석 인식 성능 향상 개발과 함께 저사양 시스템 환경에서 서비스 및 응용을 위한 지속적인 모델 경량화 및 처리 고속화의 문제를 해결해 나가며 지속적 연구개발이 수행될 것으로 예상된다.

용어해설

객체 검출(Object Detection) 이미지에서 사람, 동물, 물체 등 정형의 객체를 찾아내는 행위. 객체의 종류와 객체를 감싸는 박스의 좌표로 표현됨

신경망 구조 탐색(Neural Architecture Search) 신경망 학습은 사람이 정의한 신경망 구조에 들어갈 파라미터들을 탐색하는 과정을 말하는데, 신경망 구조 탐색은 여기서 더 나아가 인공 신경망을 이용하여 새로운 인공 신경망 구조를 찾는 과정을 의미함

객체 분할(Object Segmentation) 이미지에서 사람, 동물, 물체 등 객체의 영역을 분할하는 행위. 객체의 종류와 픽셀 단위의 객체의 영역으로 표현됨

의미론적 분할(Semantic Segmentation) 이미지 내의 객체 또는 객체와 Stuff 영역을 포함한 장면 전체에 대해 각 픽셀을 클래스 레이블 단위로 분류하는 행위

인스턴스 단위 분할(Instance Segmentation) 이미지에서 객체 검출 영역별로 클래스 카테고리를 구분하고 객체 영역을 분할하는 행위

Panoptic 분할(Panoptic Segmentation) 이미지에서 픽셀별로 객체 및 stuff의 클래스 분류, 인스턴스 단위로 객체를 분할하는 행위

약어 정리

CNN	Convolutional Neural Network
FLOP	FLoating OPerations
FPGA	Field Programmable Gate Array

GPU	Graphics Processing Unit
NAS	Neural Architecture Search
NMS	Non−Maximum Suppression
SGD	Stochastic Gradient Descent

참고문헌

[1] Uijlings, Jasper RR et al., "Selective search for object recognition," International journal of computer vision 104.2 (2013): 154-171.

[2] Ren, Shaoqing et al., "Faster r-cnn: Towards real-time object detection with region proposal networks," Advances in neural information processing systems. 2015.

[3] Liu, Wei et al., "Ssd: Single shot multibox detector," European conference on computer vision. Springer, Cham, 2016.

[4] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556 (2014).

[5] Sandler, Mark et al., "Mobilenetv2: Inverted residuals and linear bottlenecks," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[6] Redmon, Joseph et al., "You only look once: Unified, real-time object detection," Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[7] Redmon, Joseph, and Ali Farhadi. "YOLO9000: better, faster, stronger," Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[8] Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement," arXiv preprint arXiv:1804.02767 (2018).

[9] Howard, Andrew G. et al., "Mobilenets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint arXiv:1704.04861 (2017).

[10] Howard, Andrew et al., "Searching for mobilenetv3," arXiv preprint arXiv:1905.02244 (2019).

[11] Zhang, Xiangyu et al., "Shufflenet: An extremely efficient convolutional neural network for mobile devices," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[12] Ma, Ningning et al., "Shufflenet v2: Practical guidelines for efficient cnn architecture design," Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[13] Iandola, Forrest N. et al., "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size," arXiv preprint arXiv:1602.07360 (2016).

[14] Wu, Bichen et al., "Shift: A zero flop, zero parameter alternative to spatial convolutions," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[15] Yang, Tien-Ju et al., "Netadapt: Platform-aware neural network adaptation for mobile applications," Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[16] LeCun, Yann et al., "Gradient-based learning applied to document recognition," Proceedings of the IEEE 86.11 (1998): 2278-2324.

[17] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks," Advances in neural information processing systems. 2012.

[18] Jeon, Yunho, and Junmo Kim. "Constructing fast network through deconstruction of convolution," Advances in Neural Information Processing Systems. 2018.

[19] Chen, Weijie et al., "All You Need is a Few Shifts: Designing Efficient Convolutional Neural Networks for Image Classification," arXiv preprint arXiv:1903.05285 (2019).

[20] Tan, Mingxing et al., "Mnasnet: Platform-aware neural architecture search for mobile," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[21] Wu, Bichen et al., "Fbnet: Hardware-aware efficient convnet design via differentiable neural architecture search," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[22] He, Yihui et al., "Amc: Automl for model compression and acceleration on mobile devices," Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[23] Wang, Robert J., Xiang Li, and Charles X. Ling. "Pelee: A real-time object detection system on mobile devices," Advances in Neural Information Processing Systems. 2018.

[24] Yang, Yifan et al., "Synetgy: Algorithm-hardware co-design for convnet accelerators on embedded fpgas," Proceedings of the 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. ACM, 2019.

[25] Nakahara, Hiroki et al., "A lightweight yolov2: A binarized cnn with a parallel support vector regression for an fpga," Proceedings of the 2018 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. ACM, 2018.

[26] Alyamkin, Sergei et al., "Low-Power Computer Vision: Status, Challenges, Opportunities," IEEE Journal on Emerging and Selected Topics in Circuits and Systems (2019).

[27] https://rebootingcomputing.ieee.org/lpirc/2018

[28] Caesar, Holger, Jasper Uijlings, and Vittorio Ferrari. "Coco-stuff: Thing and stuff classes in context," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[29] Kirillov, Alexander et al., "Panoptic segmentation," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[30] Everingham, Mark et al., "The pascal visual object classes (voc) challenge," International journal of computer vision 88.2 (2010): 303-338.

[31] Lin, Tsung-Yi et al., "Microsoft coco: Common objects in context," European conference on computer vision. Springer,

Cham, 2014.

[32] Cordts, Marius et al., "The cityscapes dataset for semantic urban scene understanding," Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[33] Zhou, Bolei et al., "Scene parsing through ade20k dataset," Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[34] Neuhold, Gerhard et al., "The mapillary vistas dataset for semantic understanding of street scenes," Proceedings of the IEEE International Conference on Computer Vision. 2017.

[35] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation," Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[36] Noh, Hyeonwoo, Seunghoon Hong, and Bohyung Han. "Learning deconvolution network for semantic segmentation," Proceedings of the IEEE international conference on computer vision. 2015.

[37] Chen, Liang-Chieh et al., "Semantic image segmentation with deep convolutional nets and fully connected crfs," arXiv preprint arXiv:1412.7062 (2014).

[38] Yu, Fisher, and Vladlen Koltun. "Multi-scale context aggregation by dilated convolutions," arXiv preprint arXiv:1511.07122 (2015).

[39] Zhao, Hengshuang et al., "Pyramid scene parsing network," Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[40] Chen, Liang-Chieh et al., "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," IEEE transactions on pattern analysis and machine intelligence 40.4 (2017): 834-848.

[41] Lazebnik, Svetlana, Cordelia Schmid, and Jean Ponce. "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). Vol. 2. IEEE, 2006.

[42] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation," International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.

[43] Badrinarayanan, Vijay, Alex Kendall, and Roberto Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," IEEE transactions on pattern analysis and machine intelligence 39.12 (2017): 2481-2495.

[44] Lin, Guosheng et al., "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation," Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[45] Paszke, Adam et al., "Enet: A deep neural network architecture for real-time semantic segmentation," arXiv preprint arXiv:1606.02147 (2016).

[46] Chen, Liang-Chieh et al., "Encoder-decoder with atrous separable convolution for semantic image segmentation," Proceedings of the European conference on computer vision (ECCV). 2018.

[47] He, Kaiming et al., "Mask r-cnn," Proceedings of the IEEE international conference on computer vision. 2017.

[48] Cai, Zhaowei, and Nuno Vasconcelos. "Cascade r-cnn: Delving into high quality object detection," Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[49] Chen, Kai et al., "Hybrid task cascade for instance segmentation," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[50] COCO+Mapillary Joint Recognition Challenge Workshop at ECCV 2018, http://cocodataset.org/workshop/coco-mapillary-eccv-2018.html

[51] Fu, Cheng-Yang, Mykhailo Shvets, and Alexander C. Berg. "RetinaMask: Learning to predict masks improves state-of-the-art single-shot detection for free," arXiv preprint arXiv:1901.03353 (2019).

[52] Lin, Tsung-Yi et al., "Focal loss for dense object detection," Proceedings of the IEEE international conference on computer vision. 2017.

[53] Chen, Liang-Chieh et al., "Rethinking atrous convolution for semantic image segmentation," arXiv preprint arXiv:1706.05587 (2017).

[54] Apple Core ML Models: DeeplabV3, https://developer.apple.com/machine-learning/models/#image

[55] Mobile Deeplab-V3+ model for Segmentation, https://github.com/nolanliou/mobile-deeplab-v3-plus