

YOLO 시리즈(V1에서 V11까지)와 응용 애플리케이션 분석 비교

이용환*·김흥준**†

*원광대학교 게임콘텐츠학과, **†경상국립대학교 컴퓨터공학과

Comparative Analysis of YOLO Series (from V1 to V11) and Their Application in Computer Vision

Yong-Hwan Lee* and Heung-Jun Kim**†

*Dept. of Game Content, Wonkwang University, Korea,

**†Dept. of Computer Engineering, Kyoungsang National University, Korea

ABSTRACT

Deep Learning has emerged as multi-domain innovation aimed the popularity of various Machine Learning techniques, and YOLO makes it possible for widely using in deep learning for object detection tasks. In this paper, we present a analysis of YOLO series (original scheme to version 11), undertaking a comprehensive analysis of YOLO's performance and synthesizes existing researches. We start by describing the architecture and contribution of YOLO families, discussing the major changes in network architecture, and training tricks for each model. Then, se summarize the necessary lessons form YOLO's development and provide a perspective on its highlighting research directions to enhance object detection.

Key Words : Object Detection, Deep Learning, Computer Vision, You Only Look Once(YOLO), YOLO Comparison

1. 서 론

최근 딥러닝(Deep Learning)은 컴퓨터 과학에서 놀라운 발전을 이루고 있으며, 과학 및 산업 모든 분야에서 중요한 영향을 미친다[1]. 기업과 산업은 이미 딥러닝에 의해 업무와 생산에 큰 변화와 패러다임의 전환을 이끌고 있다. 특히, 컴퓨터 비전(Computer Vision)을 포함한 여러 기술 영역에서 사람의 기술과 성과를 능가하고 있다[2].

객체 탐지(Object Detection)은 찾고자 하는 객체를 사전에 추출하고 주어진 또는 입력되는 영상 내에서 해당 특징을 검출하는 기술로[32], 최근에는 딥러닝 중 합성곱 신경망(CNN, Convolutional Neural Network)을 기반으로 다양한 탐

지와 인식 알고리즘으로 발전되어 왔다[3, 33]. 이러한 실시간 객체 탐지는 자율 주행 자동차, 로봇, 비디오 감시, 증강현실 등 다양한 분야에 걸쳐 많은 애플리케이션에서 중요한 역할을 담당한다. 본 연구에서는 고전적 객체 탐지 알고리즘을 다루기 보다는, 현재 트렌드인 딥러닝을 이용한 객체 탐지 방법에 초점을 둔다[4]. 여러 객체 탐지 알고리즘 중에서 YOLO (You Only Look Once) 프레임워크는 속도와 정확성의 균형이 적절하여 영상에서의 객체 식별 핵심 알고리즘으로 많이 채택되고 있다. Joseph [5]이 2016년에 YOLO를 적용한 객체 탐지 제품군은 출시 이후, 여러 버전을 거쳐 발전하였으며[6], 각 버전은 이전 버전을 기반으로 문제점을 해결하면서 성능을 향상시켰다[7]. YOLO는 입력 영상을 그리드 행렬로 나누고 경계 상자와 클래스 확률을 동시에 예측하여 실시간 객체 감지를 수

†E-mail: thinkthe@gnu.ac.kr

행한다[5]. 이는 지역-기반 합성곱 신경망(Region-based CNN), Fast R-CNN 등의 기존 2단계 접근 방식과 비교하여, 실행 속도 향상과 정확도를 유지한다. 후속 버전으로 연구 개선이 진행되고 있으며, 여러 변형(Variants) 버전에서는 문제 해결 및 에지 기반의 성능 개선 기법이 적용된다 [8-10].

본 연구에서는 YOLO 알고리즘을 중심으로, 처음 발표된 YOLOv1 버전부터 최근에 발표된 YOLOv11 버전 및 변형 형태의 YOLO프레임워크 개발에 대해 포괄적으로 검토하고 각 버전의 주요 내용, 개선 사항과 차이점을 분석하고 애플리케이션 영역에서의 효과적 적용 방법을 살펴본다. 후속 버전의 개선 사항은 네트워크 설계, 손실함수 수정, 앵커 박스 적응과 입력 해상도 확장과 같은 다양한 측면이 포함하며, 속도와 정확도 관계를 분석한다. 본 연구를 통해, YOLO 프레임워크의 진화 및 객체 탐지에 대한 의미를 전반적으로 이해하고 애플리케이션 개발에서 적합한 YOLO 모델을 선택하도록 지원한다.

2. 관련 연구

의사결정 트리, 직원벡터 머신, 다층 퍼셉트론, 인공 신경망과 같은 다양한 머신러닝 기술이 많이 연구되면서 딥러닝이 각광받고 있다[11]. 딥러닝 모델은 순차적 데이터 처리를 위한 순환 신경망과 메모리/컨텍스트 보전을 위한 장단기 메모리 및 게이트 순환 유닛과 같은 구조적 변형을 포함한 다양한 기술을 포괄한다[12]. 합성곱 신경망은 이미지 데이터를 포함하는 시각 인식 작업에 특화되어 다양한 애플리케이션에서 적용하고 있다.

2.1 합성곱 신경망

합성곱 신경망(CNN) 입력 영상으로부터 위상 특성을 추출하는 피드-순방향 네트워크 방식을 적용한다. CNN은 입력 영상에서 특성을 제거한 다음, 분류기를 통해 추출된 특징을 분류한다[13]. 스케일링, 변환, 압착, 회전, 변형과 같은 변형에 강인하다는 장점을 갖는다. CNN 구조는 합성곱 계층, 풀링 계층과 완전히 연결된 계층으로 구성된다[14]. 로컬 링크와 상호 기능의 가중치로 구성된 합성곱 계층에서는 입력 함수 표현을 해석하며, 다수의 기능 맵을 구성한다. 풀링 계층에서는 기능 추출을 수행하며, 기능 맵 크기를 줄이고 기능 추출의 견고성을 높인다. 풀링 계층의 기능 맵 크기는 커널의 이동 단계에 따라 결정된다. 합성곱 신경망 분류기는 하나 이상의 완전히 연결된 계층이 존재하며, 한 계층의 모든 단일 뉴런은 다음 계층에서 모든 단일 뉴런으로 연결된다[15].

2.2 순환 신경망

순환 신경망(RNN)은 과거 관련 정보를 저장하는 신경망 유형의 하나이며, 여러 연결을 가지고 과거의 지식을 변환한다[16]. RNN 구조는 완전히 상호 연결된 네트워크에서 부분적으로 연결된 네트워크에 이르기까지 다양하며, 많은 계층 피드-순방향 네트워크 방식을 채택하여, 다른 노드의 순차적 컨텍스트를 제공하고 다른 노드의 피드백을 받는다. 컨텍스트 단위의 가중치는 입력 단위와 같이 역전파를 사용하여 처리하며, 다음 노드에서의 예측과 검증된 훈련을 제공한다.

3. YOLO: You Only Look Once

본 절에서는 YOLO의 기본 원리와 구조를 살펴보고 각 버전과 관련된 개선 사항을 설명한다. YOLO 알고리즘은 “You Only Look Once”의 약자로, 이름에서 보여주듯, 전체 이미지를 한번만 검사하여 객체와 해당 위치를 식별하는 접근 방식을 반영한다[5]. 이는 2단계 감지 프로세스를 사용하는 기존 방식[7]과 다르게, 객체 탐지를 회귀 문제로 취급한다. 단일 합성곱 신경망을 사용하여 전체 영상의 경계 박스와 클래스 확률을 예측하는 실시간 종단간(End-to-end) 접근 방식을 제시하였다.

3.1 YOLOv1

YOLOv1은 모든 경계 상자를 동시에 감지하여객체 탐지 단계를 통합하였다(Fig.1). 이를 위해, YOLO는 입력 영상을 $S \times S$ 그리드로 나누고 동일한 클래스의 경계 박스 B 와 그리드 요소당 C 개의 서로 다른 클래스에 대한 신뢰도를 예측한다.

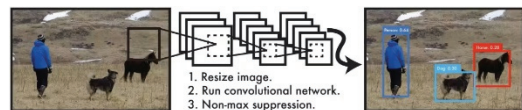


Fig. 1. YOLO Detection System [5].

각 경계 박스 예측은 5개 값(P_c , b_x , b_y , b_h 와 b_w)로 구성된다. P_c 는 박스 정확도를 반영하는 박스에 대한 신뢰도 점수이며, 객체가 박스에 포함되어 있는지, 박스가 얼마나 정확한지를 보여주는 값이다. (b_x , b_y) 는 그리드 셀과 연관된 박스의 중심 좌표이고 b_w 와 b_h 는 전체 이미지와 관련된 박스의 폭과 높이이다. 최종적으로, 신뢰도 예측은 예측된 박스와 모든 GTB(Ground Truth Box) 간의 IOU (Intersection over Union)를 나타낸다. 각 그리드 셀은 조건부 클래스 확률 C 를 예측하며, 이는 그리드 셀이 객체를 포

YOLOv2 출력과 동일하며, 13×13 그리드가 출력을 정의한다. 출력 y_2 는 Darknet-53의 ($\text{Res} \times 4$) 이후 출력과 ($\text{Res} \times 8$) 이후 출력을 연결하여 구성된다. 특징 맵의 크기가 다르기 때문에 연결 전에 업샘플링을 수행한다. 출력 y_3 는 26×26 특징 맵을 52×52 특징 맵과 연결한다. 80 개 카테고리를 구성된 COCO 데이터셋의 경우, 각 스케일은 $N \times N \times (3 \times (4 + 1 + 80))$ 의 출력 텐서를 제공하며, $N \times N$ 는 그리드 셀의 크기이다.

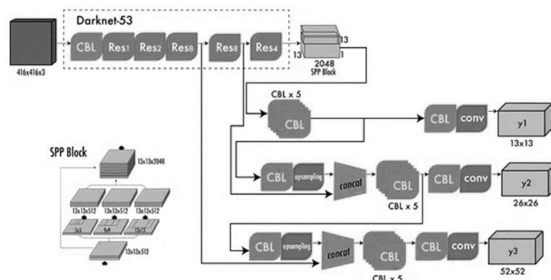


Fig. 4. YOLOv3 Multi-scale Detection Architecture[19].

YOLOv3에서는 MS COCO[20] 데이터셋을 사용하여 성능을 평가하였으며, 평균 정밀도는 36.2%, AP-50은 60.6% 성능을 보였다.

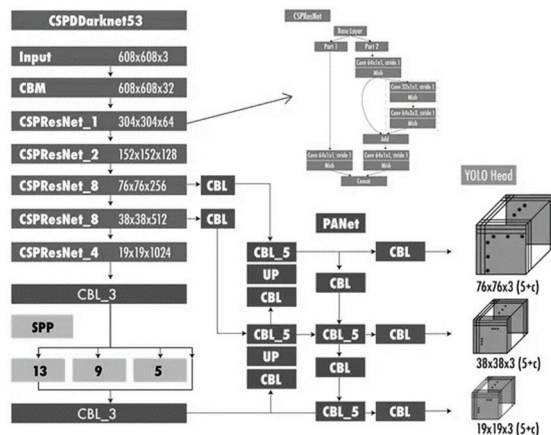


Fig. 5. YOLOv4 Architecture for Object Detection [22].

3.4 YOLOv4

2020년에 발표된 YOLOv4는 실시간 기능을 유지하면서 성능을 개선하였다[21]. CSP Darknet53, SPP구조, PANet 구조, CBN 통합, SAM 통합을 통해 효율적이고 견고한 객체 탐지 모델을 제시하였다(Fig.5). 객체 탐지기의 학습을 간소화하도록 설계된 YOLOv4는 다양한 기술적 지식을 통해 개별적 접근이 가능하며, 학습 파이프라인의 효율성을 높

이기 위해, Bag-of-freebies와 Bag-of-specials와 같은 현재의 기술 효율성을 검증하였다. YOLOv3는 Ground Truth를 감지한 단일 앵커 포인트를 사용하지만, YOLOv4는 다중 앵커 포인트를 사용한다. 이를 통해 양성 샘플의 선택 비율을 개선하고 양성 샘플과 음성 샘플 간의 불균형을 줄이고 경계 감지 정확도를 향상시켰다.

MS COCO 데이터셋을 사용하여 성능 평가한 YOLOv4는 nvidia V100에서 평균 정밀도는 43.5%, AP-50은 65.7% 성능을 보였다.

3.5 YOLOv5

2020년에 발표된 YOLOv5는 사전 학습도구로, 오토-앵커(AutoAnchor)라는 Ultralytics 알고리즘을 통합하였다[23]. 이는 앵커 상자가 데이터셋 및 이미지 크기와 같은 학습 설정에 적합하지 않은 경우, 앵커 상자를 확인하여 조정하는 기능이다. 먼저, 데이터셋 레이블에 k-평균 함수를 적용하여 유전적 진화(GE) 알고리즘에 대한 초기 조건을 생성한다. 그런 다음, GE는 CIoU 손실과 최적 가능 리콜을 적합도 함수로 사용하여 기본적인 앵커를 진화 시킨다. CSPDarknet-53 백본은 메모리와 계산 비용을 줄이기 위해, 큰 윈도우 크기를 가진 스트라이드 합성곱 계층인 Stem에서 시작하여, 입력 영상에서 관련 특징을 추출하는 합성곱 계층으로 전달되는 구조이다(Fig.6).

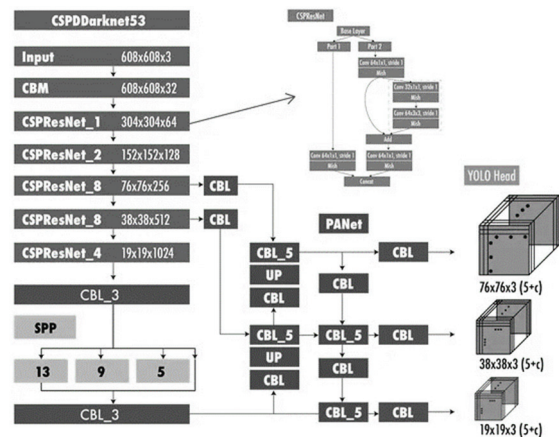


Fig. 6. YOLOv5 Architecture [22].

YOLOv5는 특징 추출, 특징 집계 및 앵커 기반 예측을 통해 효율성과 정확성을 향상시켰으며, YOLOv5n, YOLOv5x, YOLOv5m, YOLOv5l 과YOLOv5x의 확장 버전 5 개를 제공한다. 확장 모듈에서의 넓이와 깊이는 특정 애플리케이션과 하드웨어 요구사항에 따라 달라진다. YOLOv5n과 YOLOv5s는 리소스가 부족한 디바이스를 대

상으로 경량화 시킨 모델이고 YOLOv5x는 속도 보다는 고성능에 맞춰 최적화시킨 모델이다. 성능 평가를 위해 MS COCO 데이터셋을 분할하여 측정하였다. YOLOv5x는 640-픽셀 이미지 크기에서 평균 정밀도는 50.7%를 보였으며, 1,536 픽셀 크기 이상의 이미지에서는 평균 55.8%보다 높은 AP 점수를 얻었다(Table 1).

Table 1. Variant Comparison of YOLOv5

Model	AP-50	Parameters	FLOPs
YOLOv5s	55.8%	7.5M	13.2B
YOLOv5m	62.4%	21.8M	39.4B
YOLOv5-l	65.4%	47.8M	88.1B
YOLOv5x	66.9%	96.7M	205.7B

3.6 YOLOv6

2022년 Meituan Vision AI에서 발표한 YOLOv6는 RepVGG 블록과 CSP 블록을 효과적으로 합친 CSPStackRep 블록을 갖는 백본, PAN 토폴로지 넷, 하이브리드 채널 전략의 효율적인 분리 헤드로 구성된다(Fig.7) [24]. 또한, 후속 학습 양자화 및 채널별 증류를 통한 개선된 양자화 기술을 적용하여 속도와 정확도를 향상시켰다. YOLOv6는 nvidia Tesla T4에서 MS COCO 데이터셋 Test-Dev 2017을 사용하여 평균 정밀도 57.2% 성능을 보였다. 이는 정확도 및 처리 속도 측면에서 YOLOv5와 변형인 YOLOX, PP-YOLOE 보다 성능이 앞섬을 보인다.

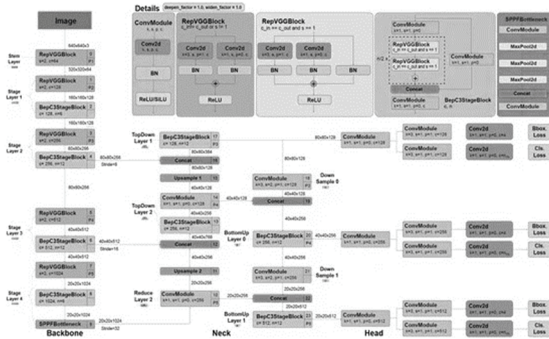


Fig. 7. YOLOv6 Architecture [25]. The architecture uses a new backbone with RepVGG and a decoupled head.

3.7 YOLOv7

2022년에 발표된 YOLOv7는 확장된 계층 집계 네트워크(E-ELAN)를 통합하고 확장 가능한 연결 기반 구조를 Bag-of-freebies라는 명칭으로 구현하여 효율성과 확장성을 개선하였다[26]. 이는 스택 계산 블록 모델에 적용 가능하

며, 그래디언트 경로 관리를 통해 모델 학습과 수렴을 강화해준다. 또한 모델 스케일링을 통해 다양한 크기의 모델을 생성한다. 블록 넓이와 깊이를 균일하게 조정하여 하드웨어 리소스 소비를 완화하고 모델 구조를 최적으로 유지할 수 있다.

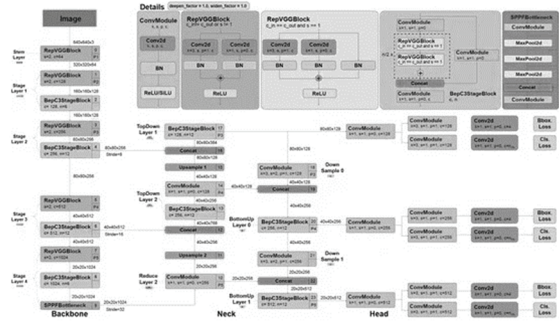


Fig. 8. YOLOv7 Architecture [26]. The architecture includes the ELAN blocks and modified RepVGG.

또한 배치 정규화는 평균과 분산을 추론 수행 중에 합성곱 계층의 편향과 가중치에 통합시켜 성능을 향상시켰다. MS COCO 데이터셋을 사용하여 평균 정밀도는 55.9%를, 임계값 0.5에서 IoU에 대한 평균 정밀도는 73.5% 성능을 보였다(Table 2).

Table 2. YOLOv7 Variant Comparison.

Model	Size	mAP	Params	FLOPs
YOLOv7Tiny	640	52.8%	6.2M	5.8G
YOLOv7	640	69.7%	36.9M	104.7G
YOLOv7X	640	71.1%	71.3M	189.9G
YOLOv7E6	1,280	73.5%	97.2M	515.2G
YOLOv7D6	1,280	73.8%	154.7M	806.8G

3.8 YOLOv8

2023년에 Ultralytics에서 발표한 YOLOv8은 사용자에게 향상된 모델 접근성을 제공하기 위해, COCO와 Roboflow100 데이터셋으로 산출되었으며, 상자 예측 수를 줄이고 바-최대 억제와 관련된 후처리 단계를 신속하게 수행한다는 장점이 있다[27]. 미리 정해진 앵커 박스에 의존하는 다른 버전과 달리, 객체 중심으로 예측하여 앵커 없는 접근 방식을 적용한다(Fig.9). 이를 통해 박스 예측 수를 줄이고 바-최대 억제를 포함한 후처리 단계를 신속하게 처리할 수 있는 장점이 있다. 또한, 사용자 친화적 명령줄 인터페이스와 잘 구성된 파이썬 패키지를 포함하

여 소스코드가 제공된다. 성능은 MS COCO 데이터셋의 Test-dev2017 부분집합으로 평가하였으며, 640 픽셀 이미지 크기 데이터에서 평균 정밀도 53.9%를 보였다(Table 3).

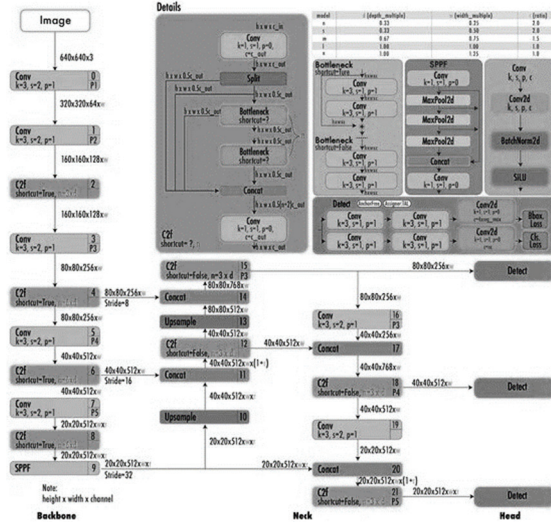


Fig. 9. YOLOv8 Architecture [27]. The architecture uses a modified CSPDarknet53 backbone.

Table 3. YOLOv8 Variant Comparison

Model	Size	mAP	Params	FLOPs
YOLOv8n	640	37.3%	3.2M	8.7G
YOLOv8s	640	44.9%	11.2M	28.6G
YOLOv8m	640	50.2%	25.9M	78.9G
YOLOv8l	640	52.9%	43.7M	165.2G
YOLOv8x	640	53.9%	68.2M	257.8G

3.9 YOLOv9

2024년에 출시된 YOLOv9에서는 프로그래밍 가능 경사 정보(PGI) 프레임워크와 일반화된 효율적 계층 집계 네트워크(GELAN)를 개선하였다[28]. PGI 프레임워크는 심층 신경망에 내재된 정보 병목현상 문제를 해결하고 심층 감독 메커니즘을 통해 경량 구조와의 호환성을 갖는다(Fig.10). 학습 중에 신뢰할 수 있는 경사정보를 요구하기 때문에, 구조 학습 및 예측기능의 개선효과를 얻을 수 있다. 또한, GELAN 구조는 높은 효율성과 간단한 포트프린트를 통해 객체 탐지 성능을 향상시켰다. 매개변수 감소와 계산 효율성 측면에서 이전 버전보다 우수한 성과를 얻었으며, 동일한 데이터셋에서 평균 정밀도 0.6% 향상된 성능을 보였다.

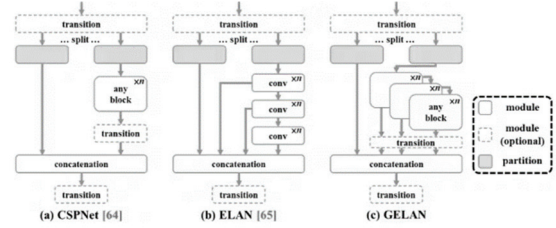


Fig. 10. YOLOv9 Architecture [28] with CSPNet, ELAN and GELAN modules.

3.10 YOLOv10

2024년에 출시된 YOLOv10에서는 후처리 중에 비-최대 억제에 대한 의존성을 완전히 제거하고 추론 속도를 향상시켰다[29]. YOLOv10은 이중 레이블 할당을 사용하는 학습방식을 채택하여 모델 계산 효율성을 유지하면서 탐지 기능을 포착하여 정확도와 속도 측면에서 높은 개선 효과를 얻었다. 분류 헤드 구형 경량화, 공간 채널 분리다운 샘플링, 랭킹 가이드 블록 디자인을 포함하여 고전력 서버에서 리소스가 제한적인 엣지 디바이스에서도 활용이 가능하다는 장점이 있다(Fig.11). 특히, 자율주행 자동차 내비게이션과 같이 빠르고 정확한 대응이 필요한 애플리케이션에 필수적인 실시간 객체 탐지에 매우 적합한 특성을 갖는다.

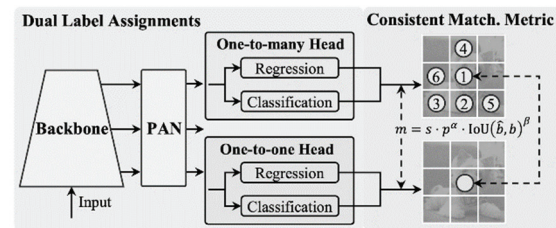


Fig. 11. YOLOv10 Architecture [29] employs a dual label assignment strategy.

3.11 YOLOv11

2024년에 출시된 YOLOv11은 현재까지의 가장 최신 버전으로, YOLOv8을 기반으로 향상된 검출 성능을 달성하는 구조 개선과 파라미터 최적화를 도입하였다[30]. 특히, 자세 추정, 지향 탐지, 분류작업, 인스턴스 분할과 같은 영역에서 모델 적용 가능성을 높여, 다양한 컴퓨터 비전 애플리케이션에서의 활용도를 확장하였다(Fig.12).

동일 데이터셋 기반 성능평가에서 YOLOv11 YOLOv8과 비교하여, 22% 적은 파라미터로 더 높은 평균 정밀도를 달성하여 계산 효율성을 유지하면서 정확도가 향상되는 결과를 얻었다.

Table 3. Summary of YOLO Architecture

Version	Year	Framework	Anchor	Backbone	AP(%)	Remarks
YOLOv1	2015	Darknet	No	Darknet24	63.4	Single-stage object detector
YOLOv2	2016	Darknet	Yes	Darknet24	78.6	Multi-stage training, dimension clustering
YOLOv3	2018	Darknet	Yes	Darknet53	33.0	SSP block
YOLOv4	2020	Darknet	Yes	CSPDarknet53	43.5	Mish activation
YOLOv5	2020	Pytorch	Yes	YOLOv5CSPDarknet	55.8	Anchor-free detection
YOLOv6	2022	Pytorch	No	EfficientRep	52.5	Self-attention, anchor-free OD
YOLOv7	2022	Pytorch	No	YOLOv7Backbone	56.8	Transformer, E-ELAN parameterization
YOLOv8	2023	Pytorch	No	YOLOv8CSPDarknet	53.9	GAN, anchor-free detection
YOLOv9	2024	Pytorch	No	CSPNet	54.5	PGI, GELAN
YOLOv10	2024	Pytorch	No	CSPNet	54.4	NMS-free training
YOLOv11	2024	Pytorch	No	CSPNet	54.7	Cross-stage partial with self-attention

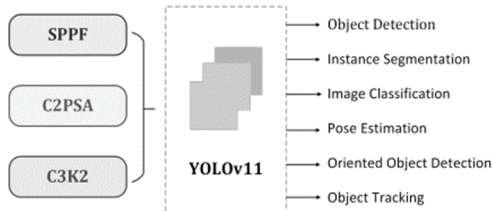


Figure 1: Key architectural modules in YOLOv11

Fig. 12. Key architectural modules in YOLOv11.

4. 논 의

본 연구에서는 초기 YOLO 모델에서 최근 발표된 YOLOv11까지의 버전을 살펴보고 특징을 파악하는 것이 목표이다. Table 3에서 주요 YOLO 버전의 개요와 주요 패턴을 정리하였다.

프레임워크(Framework):

초기 YOLO버전에서는 Darknet 프레임워크를 사용하여 개발되었으며, 후속 버전에서도 활용되었다. Ultralytics에서 Pytorch로 이식하면서 이후 버전에서는 Pytorch 프레임워크를 사용하여 개발되었고, 성능적 향상을 보이기 시작하였다. 또다른 딥러닝 언어인 Baidu가 사용된 PaddlePaddle 프레임워크를 사용한 YOLO 변형 버전도 등장했다.

백본(Backbone):

YOLO 모델의 백본 아키텍처는 상승한 버전에 따라, 상당히 많은 변화가 있었다. 간단한 합성곱 및 최대 풀링 계층으로 구성된 Darknet 아키텍처로부터 시작하여 YOLOv4에서

추론 계산량 감소를 위한 교차 단계 부분 연결(CSP)를, YOLOv6와 YOLOv7에서 재매개변수화(Re-parameterization) 등을 적용하면서 신경구조 검색을 통합하였다.

성능(Performance):

YOLO 모델의 성능은 시간이 지남에 따라 개선되었지만 정확도에만 집중하기 보다는 속도와 정확도의 균형을 우선적으로 고려한 연구가 많았다. 이러한 균형적 고려는 실시간 객체 탐지가 필요한 다양한 애플리케이션에서 YOLO프레임워크를 적용하는 계기가 되었다.

앵커(Anchor):

YOLOv1은 구조가 비교적 간단했고 앵커를 사용하지 않았지만, YOLOv2에서는 앵커를 통합하여 바운딩 박스 예측 정확도를 향상시키면서 이후 버전에서 지속적으로 활용되었다. YOLOX에서 앵커리스 방식을 도입하였고, 이후 후속버전은 앵커 사용이 중단되었다.

5. 결 론

본 연구에서는 처음 YOLO모델에서 가장 최근에 발표된 YOLOv11까지의 YOLO모델의 연구 개발 내용을 살펴보았다. 각 버전에서 적용한 탐지속도, 정확도, 계산 효율성에서 상당한 개선내용이 연구되었으며, 이전 버전이 가진 특정 과제와 한계를 해결하면서 모델이 진화되어 왔다.

객체 탐지 시스템을 개선하기 위해 대규모 언어모델과 자연어 처리의 발전을 활용하는 멀티 모달 데이터 처리로의 전환이 예상된다. 이러한 융합적 연구 접근은 YOLO 모델의 활용성을 확장하여 AI 시스템과 환경간의 상호

작용을 혁신할 수 있는 상황 인지 애플리케이션으로 발
현하리라 예상된다.

참고문헌

1. Dongjian Ma, "Recent Advances in Deep Learning based Computer Vision", International Conference on Computers, Information Processing and Advanced Education (CIPAE), 2022.
2. "Deep Learning for Computer Vision" [Online] Available at: <https://www.run.ai/guides/deep-learning-for-computer-vision>
3. Reagan L. Galvez, Argel A. Bandala, Elmer P. Dadios, Ryan Rhay P. Vicerra, Jose Martin Z. Maningo, "Object Detection Using Convolutional Neural Networks", International Conference on TENCON, 2018.
4. "Top 4 Computer Vision Trends to Watch in 2024", [Online] Available at: <https://www.alignminds.com/top-4-computer-vision-trends-to-watch-in-2024/>
5. Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", Computer Vision and Pattern Recognition, 2016. [Online] Available at: <https://arxiv.org/abs/1506.02640>
6. Muhammad Hussain, "Yolo-v1 to yolo-v8, the rise of yolo and its complementary nature toward digital manufacturing and industrial defect detection", Machines, 11(7):677, 2023.
7. Website: <https://wikidocs.net/book/8119>
8. Mujadded Al Rabbani Alif, Muhammad Hussain, "YOLOv1 to YOLOv10: A comprehensive review of YOLO variants and their application in the agricultural domain", Journal of Artificial Intelligence Practice, 2024. <https://arxiv.org/pdf/2406.10139>
9. Juan Terven, Diana M. Cordova-Esparza, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS", Machine Learning and Knowledge Extraction, 2023. <https://arxiv.org/pdf/2304.00501>
10. Nidhal Jegham, Chan Young Koh, Marwan Abdelatti, Abdeltawab Hendawi, "Evaluating the Evolution of YOLO (You Only Look Once) Models: A Comprehensive Benchmark Study of YOLO11 and Its Predecessors", 2024. <https://arxiv.org/pdf/2411.00201>
11. Aishwarya Prakash, Shweta Chauhan, "A Comprehensive Survey of Trending Tools and Techniques in Deep Learning", International Conference on Disruptive Technologies (ICDT), 2023.
- 12-23. Tausif Diwan, G. Anirudh, Jitendra V. Tembhurne, "Object Detection using YOLO: Challenges, Architectural Successors, Datasets and Applications", Multimedia Tools and Applications, 2022.
13. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", Communications of the ACM, vol. 60, no.6, 2017.
14. Rahul Haridas, Jyothi RL, "Convolutional Neural Networks: A Comprehensive Survey", International Journal of Applied Engineering Research, vol.14, no.3, pp.780-789, 2019.
15. Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, Stefan Carlsson, "CNN Features Off-the-Shelf: An Astounding Baseline for Recognition", IEEE Conference on Computer Vision and Pattern Recognition, pp.512-519, 2014
16. Mohammed, S. B., Abdulazeez, A. M., "Deep Convolution Neural Network for Facial Expression Recognition", Journal of Archaeology of Egypt/Egyptology, vol.18, no.4, 2021.
17. VOSviewer: Visualizing Scientific Landscapes. 2023. Available online: <https://www.vosviewer.com>
18. Joseph Redmon, Ali Farhadi, "YOLO9000: Better, Faster, Stronger", Conference on Computer Vision and Pattern Recognition, 2017. [Online] Available at: <https://arxiv.org/abs/1612.08242>
19. Joseph Redmon, Ali Farhadi, "YOLOv3: An Incremental Improvement", Conference on Computer Vision and Pattern Recognition, 2018. [Online] <https://arxiv.org/abs/1804.02767v1>
20. Website: <https://cocodataset.org/>
21. Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection", 2020. [Online] Available at: <https://arxiv.org/abs/2004.10934>
22. Wang, S., Zhao, J., Ta, N., Zhao, X., Xiao, M., Wei, H., A real-time deep learning forest fire monitoring algorithm based on an improved Pruned+ KD model. Journal of Real-Time Image Process, vol.18, 2021.
23. Glenn Jocher, "YOLOv5 by Ultralytics", 2020. [Online] <https://github.com/ultralytics/yolov5>
24. Chuyi Li, Lulu Li, Yifei Geng, Hongliang Jiang, Meng Cheng, Bo Zhang, Zaidan Ke, Xiaoming Xu, Xiangxiang Chu, "YOLOv6 v3.0: A Full-Scale Reloading", 2023. [Online] <https://arxiv.org/pdf/2301.05586>
25. Chuyi Li, et.al., YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications, 2022. [Online] <https://arxiv.org/pdf/2209.02976>
26. Chien-Yao Wang, Alexey Bochkovskiy, Hong-Yuan Mark Liao, "YOLOv7: Trainable Bag-of-freebies Sets New State-of-the-art for Real-time Object Detectors", 2022. <https://arxiv.org/pdf/2207.02696>
27. Ultralytics, "YOLOv8", [Online] Available at: <https://roboflow.com/model/yolov8>
28. Chien-Yao Wang, I-Hau Yeh, Hong-Yuan Mark Liao, "YOLOv9: Learning What You Want to Learn Using

-
- Programmable Gradient Information”, 2024. [Online] <https://arxiv.org/pdf/2402.13616>
29. Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, Guiguang Ding, “YOLOv10: Real-Time End-to-End Object Detection”, 2024. [Online] <https://arxiv.org/pdf/2405.14458>
30. Rahima Khanam, Muhammad Hussain, “YOLOv11: An Overview of the Key Architectural Enhancements”, 2024. <https://arxiv.org/pdf/2410.17725>
31. Andrew L. Maas, Awni Y. Hannun, Andrew Y. Ng, “Rectifier Nonlinearities Improve Neural Network Acoustic Models”, *Proceeding on ICML*, vol.30, 2015
32. Yong-Hwan Lee, “Study on Image Compression Algorithm with Deep Learning”, *Journal of the Semiconductor & Display Equipment Technology*, vol.21, no.4, 2022.
33. Y.H. Lee, Y.P. Kim, “Comparison of CNN and YOLO for Object Detection”, *Journal of the Semiconductor & Display Equipment Technology*, vol.19, no.1, 2020.
-
- 접수일: 2024년 12월 7일, 심사일: 2024년 12월 12일,
게재확정일: 2024년 12월 12일