# YOLOv8을 활용한 디지털 문서의 핵심 객체 추출 및 분류 시스템 설계

조영래[1], 김홍준[1], 박병훈[1], 신수연[2], 이치훈[1*]
[1]티쓰리큐(주), [2]한양대학교(서울)
florenshio95@gmail.com, 96hjun@naver.com, warmpark@t3q.com,
shinsy@hanyang.ac.kr, chihoon.lee@t3q.com

# System for Extraction and Classification of Critical Objects using YOLOv8

Young-Rae Cho[1], Hong Jun Kim[1], Byung Hoon Park[1],
Sooyeon Shin[2] ,Chi hoon Lee[1]
[1]T3Q(주), [2]Center for Creative Convergence Education, Hanyang University(Seoul)

요 약

디지털 문서의 유통과정에서 발생할 수 있는 보안상의 문제를 해결하기 위해서는 파일 복사, 이동 과정에 문서의 보안 등급을 자동 검출하고 특정 문서의 유출을 방지하는 보안 솔루션이 필요하다. 따라서 본 논문에서는 이러한 보안상의 문제를 해결하기 위하여 하나의 검출 분류 시스템을 제안하고자 한다. 제안한 시스템은 디지털 문서 내용을 이용하여 핵심 정보라고 판단되는 객체를 우선 추출한 후 그 핵심 유형을 분류하는 과정을 통해서 핵심 정보를 사전에 탐지하도록 하였다. 이를 위해서 SOTA를 달성한 YOLOv8를 이용하여 디지털 문서의 핵심 객체 감지하고 또한 파인튜닝을 실시한 모델을 이용하여 그 유형을 분류하도록 설계하였다. 해당 시스템 검증을 위해서 기업에서 사용하고 있는 실제 사내 문서를 데이터셋을 이용하고 그 성능평가를 실시하였다.

## 1. 서 론

정보의 이동이 손쉬워진 디지털 정보화 시대에 정보 보안에 대한 수요는 더욱더 높아지고 있다. 특히 기업의 기밀한 내용이 담겨 있는 문서가 사외로 유출되지 않도록 관리하는 것은 매우 중요하다. 그러나 문서의 개별 내용을 파악하고 그중 핵심 정보라고 간주하는 것을 컴퓨터가 자동 검출하는 데에는 여전히 기술적인 어려움이 있다.

전통적으로 컴퓨터 비전 기술과 딥러닝을 활용하여 객체 검출 연구가 활발하다. 컴퓨터 비전은 시각 정보를 인식하고 해석할 수 있는 컴퓨터 기술이며 이미지 처리, 패턴 인식과 같은 기술을 포함한다. 그리고 사람의 인지 능력을 모방하는 딥러닝 기술을 적용하여 이미지의 특징을 자동으로 학습함으로써 이미지 분류, 객체 탐지 등의 작업을 수행할 수 있게 된다.

본 논문에서는 컴퓨터 비전 기술과 딥러닝을 이용하여 문서 내의 핵심 정보라고 간주하는 객체를 탐지하고 어떤 유형의 정보인지 유형을 분류하는 시스템을 설계하는 것에 집중한다. 사용한 데이터는 익명의 기업 사내 문서를 대상으로 처리하며, 데이터 특성을 고려해 핵심 객체의 유형은 도면과 표 객체로 한정해서 처리한다.

## 2. 관련 기술

### 2.1 객체 탐지와 인식

디지털 이미지나 비디오에서 특정 개체를 식별하고 그 위치 정보를 획득하기 위해서 객체 인식(Object Recognition) 처리와 객체 위치 결정(Object Localization)처리를 복합적으로 사용한다.

객체 인식은 이미지 내에서 특정 객체가 무엇인지 식별하는 과정이다. 객체 인식은 다음과 같은 세부 3단계를 포함한다. 1) 특징 추출: 객체를 인식하기 위해 이미지에서 유용한 정보를 추출한다. 예를 들면 색상, 텍스처, 모양 등이 객체를 인식하기 위한 유용한 특징이 될 수 있다. 2) 분류: 추출된 특징을 분석하여 정의된 범주에 따라 객체를 분류한다. 3) 정확도 및 신뢰도 평가: 분류 결과의 정확성을 평가하고 그 결과의 신뢰도를 확인한다.

* 교신저자

객체 위치 결정은 입력 이미지 내에서 탐지한 객체가 정확하게 어디에 있는지, 그 위치 정보를 찾아내는 과정이다. 이는 다음과 같은 세부 3단계를 포함한다. 1) 객체 탐지: 특정 객체가 존재하는 영역을 식별하고 가장 많은 정보를 내적한 사각형의 바운딩 박스를 사용해 수행한다. 2) 영역 제안: 객체가 존재할 가능성이 높은 후보 영역을 검색하고 그 위치를 추출한다. 3) 바운딩 박스 정의: 객체 위치, 크기를 결정하기 위해서 바운딩 박스 모서리 정보를 이용하여 좌표 정보를 획득한다.

## 2.2 컨볼루션 신경망 모델

컨볼루션 신경망(Convolutional Neural Network, CNN)[은 이미지 처리와 분석에 주로 사용되는 딥러닝의 한 형태이다. CNN은 이미지의 원시 픽셀 데이터를 입력으로 받아 자동으로 이미지의 특징을 추출하고 학습한다. 컨볼루션 계층은 필터를 통해 이미지의 특징을 추출하며, 풀링 계층은 데이터의 크기를 줄이고 중요한 정보를 유지한다. CNN은 이미지의 복잡한 패턴을 자동으로 학습하고, 위치 변화에 강건한 특성을 가지기 때문에 이미지 분류, 객체 감지 등의 작업에서 핵심적인 역할을 하게 되었다.[1] R-CNN, YOLO, SSD은 대표적인 CNN 기반의 객체 감지 딥러닝 모델이다.

1) R-CNN(Regions with CNN features)

객체 감지를 위해 지역 기반의 CNN을 사용한다. Selective Search 알고리즘을 사용해 이미지 내에서 객체 후보 영역을 생성한 후, 각각의 후보 영역은 CNN을 통과해 특징을 추출하고 각 영역에 대해 객체 분류를 수행한다. 높은 정확도를 제공하지만 느린 처리 속도가 단점이다.[2]

2) YOLO(You Only Look Once)

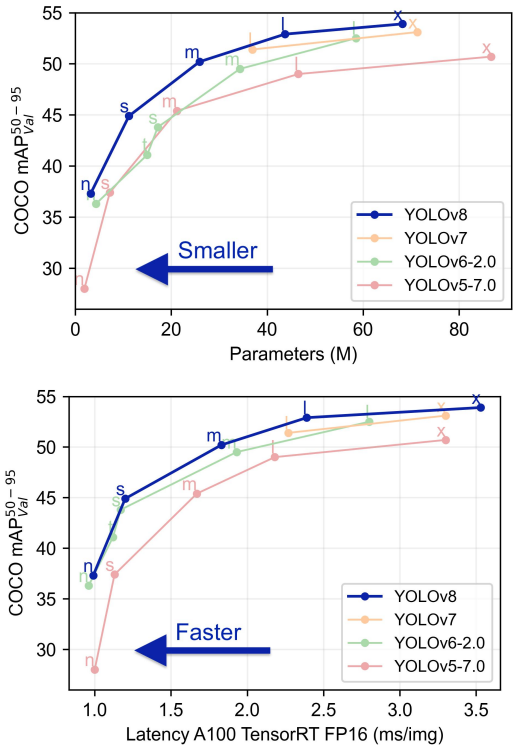
이미지 전체를 단일 신경망을 통해 한 번에 처리하는 방식이다. 이미지를 그리드로 나누고, 각 그리드 셀에서 객체의 존재 가능성과 바운딩 박스를 동시에 예측한다. 빠른 처리 속도와 좋은 실시간 성능이 장점인 모델이다.[3]

3) SSD(Single Shot MultiBox Detector)

이미지를 한 번의 단일 처리 과정(Single Shot)으로 다양한 크기의 객체를 감지한다. 그리고 여러 크기의 특징 맵을 사용해 각각의 특징 맵에서 객체의 위치를 예측한다. 이는 빠른 속도와 높은 정확도를 동시에 제공한다. YOLO에 비해 작은 객체 감지에 더 강점을 보이지만 여러 스케일의 특징 맵을 사용하기 때문에 실시간 처리에서는 YOLO가 더 빠른 성능을 보인다.[4]

## 2.3 YOLOv8

YOLOv8은 2024년 1월 기준 YOLO의 최신 버전이자 SOTA 모델이다. 해당 모델은 이미지 감지, 분류, 분할, 자세 추정을 지원한다. YOLOv8은 사전학습된 데이터양에 따라 nano, small, medium, large, extralarge 모델로 나뉜다. COCO val2017 데이터셋으로 성능 평가를 한 결과 (그림 1)과 같이 mAP50-95 성능 지표에서 YOLOv8의 모든 모델이 역대 YOLO 모델 중 최고 성능을 보였다.




(그림 1) YOLOv8 공식 문서의 성능 지표 그래프

가장 많이 사용되는 YOLOv5와 비교했을 때, 동일하게 CSPDarknet53을 백본으로 하고 NMS(Non-Maximum Suppression)으로 동일 객체에 대한 다중 감지를 억제하지만, 효율적인 합성곱 구조와 고급 정규화 기술을 통한 세부 사항 변경으로 성능 향상을 꾀했다.[5][6]

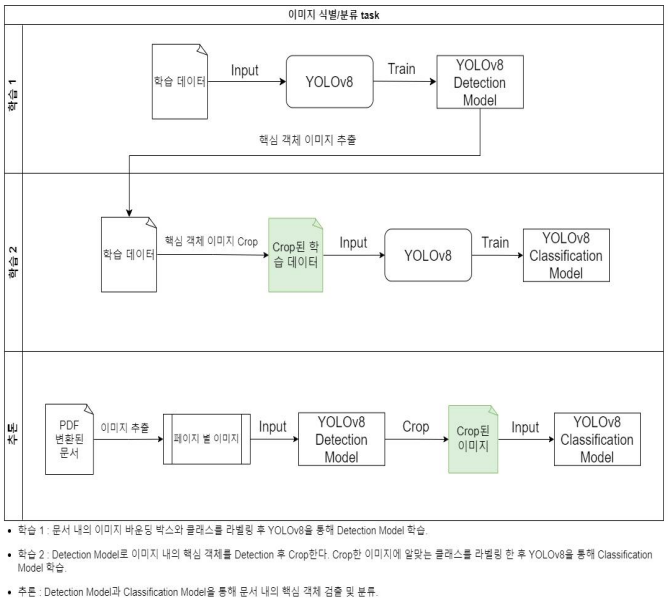
## 3. 제안한 시스템

### 3.1 실험 설계

AI 모델은 빠른 추론 속도 대비 높은 정확성이 요구된다. 따라서 싱글 스테이지 객체 탐지를 통해 실시간 이미지 처리에 적합한 YOLO모델의 최신 SOTA인 YOLOv8을 파인튜닝 하였다.

실제 추론은 문서 데이터를 pdf로 변환한 후 각 페이지를 png 포맷 파일로 변환해 YOLO모델에 입력하는 것으로 진행한다.

본 논문에서 제안하고 실험한 핵심 객체 추출 및 분류 시스템은 (그림2)와 같이 3-steps로 진행되며 요약하면 다음과 같다.




(그림 2) 시스템 전체 프로세스

첫 번째 step에서는 핵심 객체 추출을 위한 1차 모델 학습 단계이다. 원본 문서의 각 페이지 별 png 파일에서 핵심 객체를 바운딩 박스와 알맞은 분류군으로 라벨링한다. 그다음 png 파일과 라벨링 데이터가 저장된 txt파일을 Input 데이터로 하여 YOLOv8을 학습시킨다. 이때 분류 유형은 Drawing, Table, Column의 3가지다. 결과적으로 이미지상의 핵심 객체를 감지하는 YOLOv8 Detection Model을 산출한다.

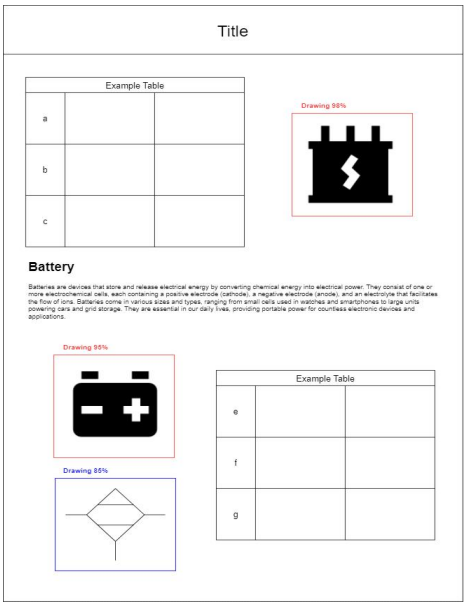
두 번째 step에서는 1차 모델에서 검출된 핵심 객체의 상세한 유형 분류를 수행하는 2차 모델을 학습한다. 단, 2차 유형 분류를 하는 대상은 1차 모델 추론 결과에서 Drawing에 해당하는 결과값으로 한정했다. 1차 모델을 이용해 1차 모델에 사용했던 데이터를 추론해 Drawing에 해당하는 핵심 객체만을 Crop한다. Crop된 이미지를 알맞은 2차 분류 유형으로 라벨링해 Crop 이미지와 라벨링 데이터를 YOLOv8에 Input하여 학습시킨다. 결과적으로 2단계 유형 분류를 하는 YOLOv8 Classification Model을 산출한다.

세 번째 step에서는 실제 추론 서비스 단계이다. 원본 문서가 들어오면 PDF로 변환한 뒤 페이지별로 png 파일로 변환한다. 변환된 png 파일에서 1차 모델로 핵심 객체를 검출하고, 검출된 객체가 Drawing 유형일 경우 Crop한 뒤 Crop된 이미지를 2차 모델로 유형 분류를 진행한다.

### 3.2 실험 데이터셋

1차 모델 학습에 사용된 데이터는 사내 제품에 대한 도면 png 파일 400건을 사용하였다. 일반 비정형 형식의 문서 안에 도면이 삽입된 페이지에 대한 png 파일은 500건이다. 비정형 문서란, 구조와 형식이 미리 정해진 정형 문서와 반대로, 구조와 형식이 정해지지 않아 쉽게 알아보기 힘든 구조를 지닌 문서를 뜻한다. 실험에서 사용한 훈련 문서는 (그림 3)과 같이 도형의 형태로 요약해서 설명할 수 있다. 단, 데이터 보안 문제로 실제 문서와 다르게 샘플 형태로 제시하였다.

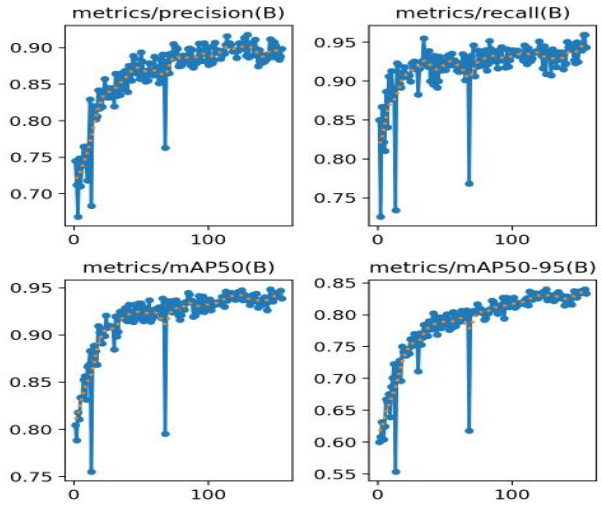
2차 모델 학습 데이터는 1차 모델 학습에 사용된 데이터에서 Crop된 png 이미지 400건이다.




(그림 3) 입력 데이터: 문서 샘플

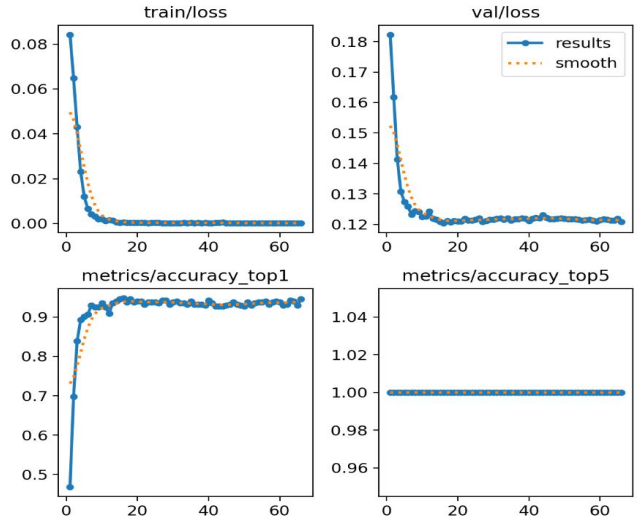
### 3.3 실험 결과

1차 모델(YOLOv8 Detection model) 학습은 약 16시간 소요되었다. 최대 1000 에포크 설정으로 학습을 시작해 성능 향상이 50 에포크를 넘어도 나타나지 않으면 학습이 자동으로 종료된다. 배치 사이즈는 4, 최고 성능이자 최종 에포크는 150이다. 낮은 해상도의 데이터가 섞여 있는 것을 고려해 멀티 스케일 데이터 증강 기법을 활용했다. 최고 mAP50은 0.9425, 최고 mAP50-95는 0.8368이며 다른 학습 결과 지표와 함께 (그림4)에서 확인할 수 있다.




(그림 4) 1차 모델 학습 결과 그래프

2차 모델(YOLOv8 Classification model) 학습은 약 2시간 소요되었으며 학습 하이퍼 파라미터는 1차 모델과 같다. 최고 성능 에포크는 66이며 최고 accuracy_top1은 0.9453이며 (그림 5)는 학습 결과 성능 그래프이다.




(그림 5) 2차 모델 학습 결과 그래프

### 4. 결론 및 향후 계획

본 논문에서는 본안 솔루션을 개발하기 위해서 딥러닝 기술을 이용한 디지털 문서의 핵심 객체 감지 및 유형 분류 시스템을 설계하고 실험을 진행하였다.

핵심 객체 검출은 사내 문서를 png 이미지로 변환하고 핵심 객체를 추출하는 YOLOv8 Detection Model을 파인튜닝함으로써 구현하였다. 핵심 객체가 검출되면 도면 정보만 이용해서 파인튜닝한 YOLOv8을 이용하여 2차 유형 분류를 진행한다. 1차 모델과 2차 모델은 각각 학습 결과 mAP50는 0.9425, accuracy_top1은 0.9453로 측정되었으며 높은 성능을 보여줌을 확인할 수 있었다.

성능 지표는 학습 과정에서 계산된 결과이기 때문에 실제 테스트 데이터셋을 통한 성능 평가가 필요하다. 또한 비슷한 성격의 사내 문서만을 학습 데이터로 사용했기 때문에 과적합에 대한 가능성도 여전히 고려되어야 한다. 따라서 향후 계획으로 테스트 데이터셋을 구축해 더 자세한 모델의 성능 평가를 시도하고 범용성을 확인하고자 한다.

### 참고문헌

[1] Yann LeCun et al, "Gradient-Based Learning Applied to Document Recognition", IEEE, 1998.

[2] Ross Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation Tech report (v5)", arXiv, 2014.

[3] Joseph Redmon et al, "You Only Look Once: Unified, Real-Time Object Detection", arXiv, 2016.

[4] Wei Liu et al, "SSD: Single Shot MultiBox Detector", arXiv, 2016.

[5] YOLOv8 공식문서, "https://docs.ultralytics.com/",

[6] YOLOv8 github, "https://github.com/ultralytics/ultralytics"