DOI: https://doi.org/10.13161/kibim.2021.11.2.024

이미지 기반 실시간 건설 현장 장비 및 작업자 모니터링을 위한 딥러닝 플랫폼 아키텍처 도출

Deep learning platform architecture for monitoring image-based real-time construction site equipment and worker

강태욱[1], 김병곤[2], 정유석[3]
Kang, Tae-Wook[1] · Kim, Byung-Kon[2] · Jung, Yoo-Seok[3]
Received May 14, 2021; Received June 05, 2021 / Accepted June 07, 2021

ABSTRACT: Recently, starting with smart construction research, interest in technology that automates construction site management using artificial intelligence technology is increasing. In order to automate construction site management, it is necessary to recognize objects such as construction equipment or workers, and automatically analyze the relationship between them. For example, if the relationship between workers and construction equipment at a construction site can be known, various use cases of site management such as work productivity, equipment operation status monitoring, and safety management can be implemented. This study derives a real-time object detection platform architecture that is required when performing construction site management using deep learning technology, which has recently been increasingly used. To this end, deep learning models that support real-time object detection are investigated and analyzed. Based on this, a deep learning model development process required for real-time construction site object detection is defined. Based on the defined process, a prototype that learns and detects construction site objects is developed, and then platform development considerations and architecture are derived from the results.

KEYWORDS: Deep Learning, Construction Site Management, Platform, Architecture, Consideration

키 워 드: 딥러닝, 공사 현장 관리, 플랫폼, 아키텍처, 고려사항

1. 서 론

1.1 연구의 배경 및 목적

최근 스마트 건설 연구를 시작으로 인공지능기술 기반 시공 관리 자동화 기술에 대한 관심이 높아지고 있다. 딥러닝(Deep learning)으로 대변되는 인공지능 기술은 구글, 페이스북 및 세계 선진 대학 연구소와 오픈소스 조직에서 개발한 기술이 실용화되면서, 복잡한 건설 환경에서 필요한 정보를 자동으로 모니터링할 수 있는 수준까지 발전했다.

건설 현장 관리를 자동화하기 위해서는 건설 장비나 작업자와 같은 객체를 인식하고, 서로간의 관계를 자동으로 분석할 수 있어야 한다. 예를 들어, 공사 현장에 작업자와 공사 장비 간의 관계를 알 수 있으면, 작업 생산성, 장비 운전 상황 모니터링, 안전 관리 등 현장 관리의 다양한 유스케이스(Usecase)를 구현할 수 있다. 이를 위해선, 실시간으로 건설 현장 객체를 인식할 수 있어야 한다.

본 연구는 실시간으로 건설 현장 관리를 지원하기 위해, 비전(Vision) 기반 딥러닝 플랫폼 아키텍처를 제안한다. 이를 위해, 딥러닝 기술을 이용해 공사 현장 관리 서비스를 개발할 때 필요한 고려사항을 도출한 후, 플랫폼 아키텍처를 제안하다. 이 연구는 실제 건설 현장의 대표적인 객체들을 학습하고, 객체 인식 결과를 통해, 딥러닝 모델 및 서비스 가변성을 고려한 플랫폼 아키텍처 개발 시 고려사항을 도출한다. 본 연구 범위는 딥러닝 비전 기술을 이용한 객체 탐지 내로 한정한다.

[1]정회원, 한국건설기술연구원, 연구위원 (laputa99999@gmail.com) (교신저자)
[2]정회원, 한국건설기술연구원, 연구위원 (bkkim@kict.re.kr)
[3]정회원, 한국건설기술연구원, 연구원 (yooseok@kict.re.kr)

1.2 연구의 범위 및 방법

이미지 기반 실시간 건설 현장 모니터링 서비스를 지원하는 딥러닝 플랫폼 아키텍처를 도출하기 위해, 관련 문헌조사를 통해, 딥러닝 기반 건설 현장 관리 기술 개발 사례를 조사하여 시스템 아키텍처 개발 시 필요한 요구사항을 확인한다. 아울러, 실시간 객체 인식을 위한 대표적인 딥러닝 모델을 조사하고, 분석한다. 이를 바탕으로 실시간 건설 현장 객체 탐지를 위한 딥러닝 모델 개발 프로세스를 분석한다. 분석된 결과를 바탕으로, 건설 현장 객체를 학습 및 탐지하는 프로토타입을 개발 한 후, 그 결과를 통해 고려사항 및 아키텍처를 도출한다. 이를 위해, 건설 현장 안전 관리 시나리오를 고려하고, 플랫폼 컴포넌트 정의 시 이를 일반화한다.

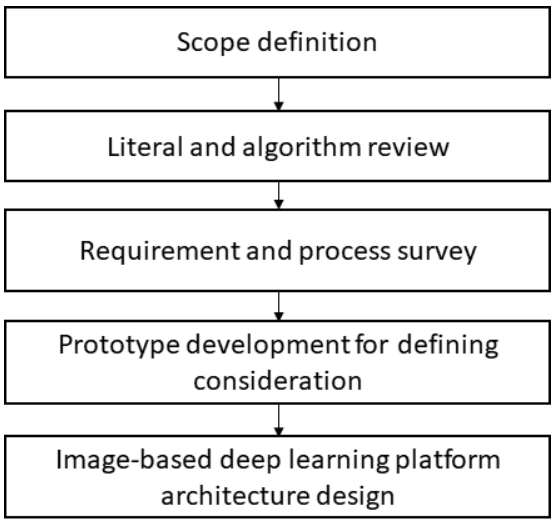

Figure 1. Study process

1.3 문헌조사

본 장에서는 건설 현장 관리를 위한 딥러닝 모델 및 아키텍처 개발과 관련된 연구 동향을 살펴보고, 플랫폼 개발에 필요한 고려사항을 확인한다.

컴퓨터 비전 기반 건설 근로자 안전모 착용 여부에 관한 딥러닝 적용 연구가 있었다(Kim and Seo, 2019). 이 연구는 R-CNN (Region-based Convolutional Neural Network)을 사용해 안전모 객체 탐지에 특화된 기술을 제안한 것이다.

딥러닝 기반 교량 구성요소 자동 분류에 관한 연구가 있었다(Lee et al, 2020). 이 연구는 교량 부재를 구분하기 위해 ResNet(Residual Neural Network), CNN을 사용하였다. 이 연구는 아키텍처 연구와는 차이가 있다.

건축물 파사드 데이터세트 구축에 관한 연구가 있었다(Gu et al, 2019). 이 연구는 딥러닝 라벨링 데이터를 신속하게 구축할 수 있는 방법을 제안한 것이다. 이를 위해, 건축물 파사드 데이터 포털사이트를 활용하였다.

도로포장 유지관리 계획 수립을 위한 딥러닝 예측 모델 개발 연구가 있었다(Lee et al, 2019). 이 연구는 도로포장 열화모델 예측을 위해 순환신경망을 사용하였다. 이 연구는 비전 기반 객체 탐지와는 거리가 있다.

딥러닝 및 에지(Edge) 컴퓨팅 기반 지능형 컨스트럭션 영상 시스템에 관한 연구가 있었다(Kim, 2019). 이 연구는 현장 영상이 에지 컴퓨터에서 오버플로우(Overflow)되는 현상을 제어하는 연구와 관련된 것이다.

딥러닝 기반 공동주택 마감공사 생산성 예측 모델 개발에 관한 연구가 있었다(Lee et al, 2019). 이 연구는 공동주택 마감공사 생산성 데이터를 활용한 예측 모델 개발 방법을 제안하였다.

딥러닝 기반 건설 차량 인식에 관한 연구가 있었다(Arabi et al, 2020). 이 연구는 객체 인식을 위해 SSD(Single Shot Detector)를 사용하였다. 이 연구는 건설 차량 인식에만 한정된다.

비전 딥러닝 모델 기반 건설 현장 다중 객체 인식에 관한 연구가 있었다(Xiao and Kang, 2021). 이 연구는 YOLO(v3)를 사용해 건설 장비 객체를 추적하는 방법을 제안하였다.

연구 동향 조사 결과, 딥러닝 모델을 이용한 건설 현장 관리 기술 연구는 초기 단계이다. 대부분의 연구는 안전모, 건설 차량 탐지 등 특정 객체 탐지 및 이 정보를 활용한 응용 연구에 초점이 맞추어져 있다.

본 연구는 실시간 건설 현장 관리를 지원하기 위한 비전 기반 딥러닝 플랫폼 아키텍처를 제안에 초점을 맞춘다.

2. 실시간 객체인식 딥러닝기술 이론고찰

2.1 개요

건설 현장 이미지에서 객체들은 다양한 영역에 분포되어 있다. 이런 이유로, 이미지에서 객체를 탐지하기 전에 객체가 어느 경계 범위 내에 있는 지를 먼저 확인해야 한다. 이후, 각 경계 범위 내 이미지를 추출하여, CNN 등의 기법을 이용해, 객체의 클래스(Class) 정확도를 계산한다. 이 장에서는 아키텍처 요구사항을 도출하기 위해, 이와 관련된 딥러닝 구조를 조사한다.

2.2 경계상자 기반 객체 예측 개념

경계상자를 식별하기 위해, 다음 같은 변수를 정의한다.

Table 1. Boundary box definition

Varaible	Definition
object.class.name	object class name definition
boundary.box.point1	left top point of boundary box
boundary.box.point2	right bottom point of boundary box

경계 상자를 예측하는 방식 중 슬라이딩 윈도우(Sliding Window) 알고리즘은 슬라이딩 윈도우 영역 이미지가 객체 클래스 예측 함수에 전달된다. 슬라이딩 윈도우는 이미지의 객체 탐색을 위해, 이미지 좌상단부터 일정 크기의 경계 상자를 만들어, 그 안에 객체를 탐색하는 과정을 반복한다. 다음 의사코드는 그 과정을 정의한다.

```
object.class predict_object_class(image, window) {
  for(x = 0; x 〈 image.width; x++) {
    for(y = 0; y 〈 image.height; y++) {
      boundary.box = create_boundary_box
                (x, y, window.width, window.height);
      window_image = create_window_image
                (image, boundary.box);
      predict_class = predict_object_class
                (window_image);
      if(predict_class 〉 0)
        object.class = predict_class;
    }
  }
  return object.class;
}
```

의사코드에서 image.width, image.height는 입력된 이미지의 크기를 정의한다. create_boundary_box 함수에서 경계상자 상단 좌측 위치를 입력받고, 윈도우 크기를 사용해 경계산자 하단 우측 위치가 포함된 boundary.box를 생성한다. 이를 이용해, create_window_image 함수에서 윈도우에 속하는 이미지인 window_image를 리턴받고, 이 이미지를 이용해, 객체 클래스를 예측한다.

이 방식은 이미지의 모든 픽셀을 다 검색해야 하므로, 많은 계산이 필요하다. 이를 해결하기 위해, 이미지 피라미드를 생성하여 문제를 해결한다. 이미지는 최소 크기에 도달 할 때까지 다운 샘플링(Down Sampling)된다. 이러한 이미지 각각에 대해 객체 검출이 실행된다.

경계상자기반 객체 탐지는 한 이미지 내에서 많은 객체가 포함되어 있을 때 핵심적인 프로세스로 정의된다. 단, 이런 방식은 정확도 높은 객체 탐색이 가능하지만, 윈도우 검출 슬라이딩에서 획득한 수많은 이미지를 CNN으로 실행하는 것은 높은 계산 성능이 필요하다. 이는 실시간 객체 탐지가 필요한 건설 현장 관리 유스케이스에는 적용하기 어려우므로, 실시간 다중 객체 탐지 딥러닝 구조 및 알고리즘을 활용한다.

2.3 실시간 다중 객체 탐지 딥러닝 구조 및 알고리즘

본 장에서는 실시간 다중 객체 탐지를 위한 대표적인 딥러닝 모델 구조를 분석하고, 고려사항 도출을 위해 필요한 내용을 분석한다.

R-CNN 알고리즘은 객체 분류에 필요한 경계 상자의 수를 줄여 계산 속도를 개선한다. 선택적으로 객체 검색에 필요한 경계 상자 수를 줄이기 위해, 이미지의 질감, 강도, 컬러 등의 이미지 정보를 사용한다. 이는 이미지를 경계상자로 예측하기 전에, 전처리를 통해 경계상자를 줄이는 방식으로 처리할 수 있다. 이를 고려한, SPP-net(Spatial Pyramid Pooling Network)은 전체 이미지에 대한 공간 피라미드를 한번에 구성해 각 경계상자 별로 CNN을 계산한다.

SSD는 객체 탐지 속도 및 정확도 사이의 균형을 고려한 알고리즘이다. SSD는 입력 이미지에 대한 CNN을 한번 계산한 후 형상맵(Feature Map)을 얻는다. 경계 상자 및 객체 분류 확률을 예측하기 위해 이 형상 맵은 3×3 크기로 정의된 후, CNN을 수행한다. 이 구조는 다양한 스케일(scale)의 객체를 탐지할 때 유리하다.

YOLO(You only Look Once)는 각 이미지를 RxC 개의 그리드(Grid)로 분할하고, 그리드의 신뢰도를 계산한다. 신뢰도는 그리드 내 객체 인식 시 정확성을 반영한다. 처음에는 객체 인식과는 동떨어진 경계 상자가 설정되지만, 신뢰도를 계산하여 경계 상자의 위치를 조정함으로써, 가장 높은 객체 인식 정확성을 가지는 경계 상자를 얻을 수 있다. 그리드에 객체 포함 여부를 계산하기 위해, 객체 클래스 점수를 계산한다. 이 결과로 총 RxCxN 객체가 예측된다.

YOLO의 학습 데이터(train.data)는 image와 여러 개의 앵커 박스(anchor.box)와 라벨로 구성된다. 학습 모델의 출력(Output)은 실제 앵커박스, 객체화 점수(Objectness Score), 클래스 점수(Class.score)로 구성된다. 그러므로, YOLO의 입출력 데이터는 다음과 같이 정의된다.

```
train.data={image, {anchor.box, label}*}
output={anchor.box, objectness.score, class.score}*
*=multiple
```

YOLO(v3)는 이미지를 격자화해, 3개의 비율(Aspect Ratio)별 anchor.box로 출력한다. 훈련 데이터셋으로 유명한 COCO 데이터셋(80개 클래스)의 경우, 각 격자마다 깊이 방향으로 255((4 + 1 + 80) x 3)개 깊이 격자가 생성된다. 이 경우, 이미지 별 격자 크기는 13x13, 26x26, 52x52인 3가지 종류로 나뉜다. 기존 방법(SSD, R-CNN)은 정해진 비율의 경계박스만 사용했다.

앞서 설명한 YOLO구조를 제외한 나머지 부분은 기존에 개발

된 CNN, FCN(Fully Convolutional Network), ResNet 등을 재활용한다.

여기서, ResNet은 네트워크의 깊은 깊이로 인한 Gradient Vanishing 및 Exploding 발생을 방지하고, Degradation되는 현상을 피하기 위한 목적으로 사용된다. 신경망 학습목적이 입력 x를 목적값 y로 맵핑하는 함수 H(x)를 탐색하는 것이라면, 학습방향은 다음 수식을 최소화하는 것이다.

$$minimize(H(x) - y)$$

ResNet에서는 관점을 바꿔 H(x) − x를 탐색하도록 수정한다. 입출력 잔차를 $F(x) = H(x)-x$로 정의를 하고 학습은 이 F(x)를 탐색한다.

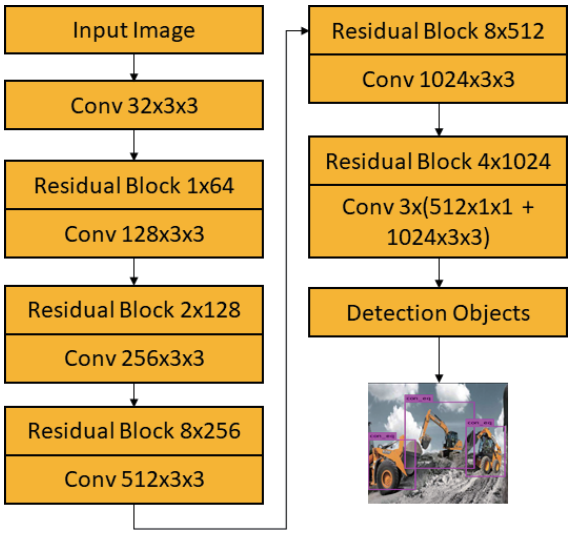

Figure 2. YOLO model architecture(Conv = Convolution Net)

이 F(x)가 잔차이므로, 이를 Residual learning, Residual mapping라 한다. 계산 상으로는 단순히 F(x) + x를 한 것으로 그 전 레이어 값을 더하고 relu(Rectified Linear Unit)연산 적용한 것 뿐이지만, 이런 문제를 개선해 높은 성능을 얻을 수 있다.

YOLO에서 Up-sampling은 저해상도에서 고해상도로 이미지 스케일 업할 때 사용된다. 이는 보간법과 유사한 방식으로, Transpose convolution, Deconvolution으로 불린다. 이런 구조를 고려해 CNN 레이어(layer)를 개발하면, Figure 2와 같은 YOLO 딥러닝 모델이 된다.

YOLO의 그리드 대부분은 낮은 신뢰도를 가질 수 있다. 신뢰도를 높이기 위해 주변의 그리드를 합칠 수 있다. 이후, 임계값을 설정해 불필요한 부분은 필터링(Low Confidence Filtering)할 수 있다(Figure 3). 이를 통해, 불필요한 이미지 예측 계산 단계를 줄일 수 있다.

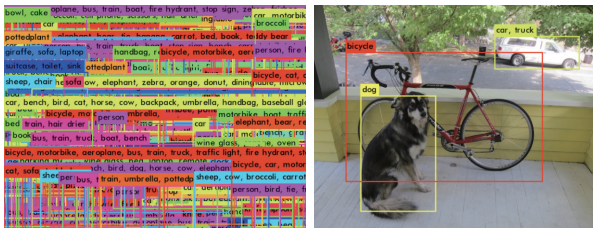

Figure 3. Low confidence filtering(YOLO)

YOLO는 단순한 처리로 속도가 매우 빠르다. 기존 다른 실시간 비전 기술과 비교할 때, 2배 이상의 높은 성능을 보인다.

본 연구에서는 건설 현장 관리 실시간성을 고려해 YOLO를 사용한다.

2.4 학습 데이터 구조

이 장은 딥러닝 학습 모델에 입력되는 대표적인 데이터 구조를 분석한다.

COCO(Common Objects in Context)는 대규모 객체 인식, 세그먼테이션, 데이터셋 캡션을 위해 고려된 학습 데이터셋이다. COCO는 330,000개 이미지와 200,000개 라벨링된 객체 정보를 제공하고, 80개 객체 카테고리를 제공한다. COCO 데이터셋은 객체 탐지 및 세그먼테이션 딥러닝 모델 학습 기본 데이터셋으로 널리 활용되고 있다. COCO구조는 JSON파일 포맷이며, 다음과 같이 경계 박스 등 정보를 포함한다.

info: dataset information
license: image license
categories: image category list
images: image information
annotations: image 개별 객체 annotation 정보

Pascal(Pattern Analysis, Statistical Modeling and Computational Learning) VOC(Visual Object Classes)구조는 객체 탐지를 위한 표준 이미지 데이터셋을 제공한다. COCO구조와 다른 점은 VOC는 XML을 사용하고, 개별 이미지별로 라벨 정보 파일이 존재하며, 경계 박스가 최상단 좌표, 최하단 좌표란 점이다.

3. 실시간 건설 현장 객체 탐지 요구사항 및 프로세스 정의

3.1 요구사항 정의

앞서 조사한 내용을 고려해, 실시간으로 건설 현장 객체를 탐지할 수 있는 플랫폼 요구사항을 정의한다. 이를 위해, 건설 현장 객체 탐지에 관련된 유스케이스를 설계한다.

건설 현장 객체 탐지와 관련된 유스케이스는 다양할 수 있다. 이 장에서는 비교적 분석하기 용이한 건설 현장 관리와 관련된 유스케이스를 건설 현장 안전 관리 5년 이상 경험을 가진 실무자 2인과 딥러닝 모델 개발 2년 이상 경험을 가진 전문가 2인의 자문을 통해 도출하였다. 질문 내용은 다음과 같다.

Q1. 건설 현장 관리 프로세스 각 단계와 요구 정보
Q2. 프로세스에서 공통적으로 포함되는 단계
Q3. 건설 현장 별로 가변적인 프로세스 단계
Q4. 딥러닝 모델 개발 프로세스와 요구 정보
Q5. 요구사항에 따라 가변적인 딥러닝 개발 단계

건설 현장의 안전 관리는 현장 객체를 실시간으로 탐지하는 작업이 필수적이다. 안전 관리 요원은 현장에 배치되어 안전 장비 미비한 작업자, 작업자와 건설 장비 간의 간섭, 위험한 지역의 작업자 및 건설 장비 진입 등을 실시간으로 모니터링한다. 이를 유스케이스로 정의하면 다음과 같다.

S1. 건설 프로젝트 관리 정보 정의
S2. 안전 관리 일정 정의
S3. 안전 펜스 영역 정의
S4. 안전 장비 착용 확인
S5. 안전 펜스 영역 내 작업자 진입 여부 확인
S6. 작업자와 건설장비 충돌 여부 확인
S7. 안전 이슈 리포트
S8. 현장 내 객체들 탐지
S9. 탐지된 객체들 저장
S10. 탐지된 객체들 로딩

각 유스케이스는 다음 질문을 통해 도출된 것이다.

Q1 = {S1, S2, S3, S4, S5, S6, S7}
Q2 = {S6, S8, S9, S10}
Q3 = {S1, S2, S3, S4, S5, S6}
Q4 = {S8, S9, S10}
Q5 = {S6, S8}

유스케이스와 액터(Actor)간의 관계를 확인하기 위해, 유스케이스 다이어그램을 Figure 4과 같이 정의하였다.

이를 통해, S8, S9, S10은 다른 유스케이스의 공통 유스케이스로 특정 사용 상황에 가변적이지 않으므로, 플랫폼에서 재활용 가능한 컴포넌트로 구현할 수 있음을 알 수 있다. 이외 유스케이스들은 서비스 목적에 따라 데이터 형태 및 처리구조가 변하므로 가변성을 고려할 필요가 있다.

유스케이스를 고려한, 플랫폼 아키텍처 설계에 필요한 핵심 요구사항을 정의하면 다음과 같다.

R1. 비전(Vision) 기반 실시간 객체 탐지
R2. 객체 탐지 결과 통보
R3. 건설 현장 관리 서비스 정의
R4. 건설 현장 관리 서비스 관리자
R5. 건설 현장 관리 데쉬보드
R6. 건설 현장 관리 데이터베이스

3.2 딥러닝 모델 개발 프로세스

앞서 조사한 딥러닝 모델 알고리즘, 유스케이스와 요구사항을 고려해 이미지 기반 실시간 건설 현장 객체 탐지를 위한 모델 학습 및 예측 프로세스를 Figure 5와 같이 정의한다.

프로세스는 딥러닝 모델을 생성하고 사용하는 부분으로 구성된다. 딥러닝 모델 생성 전에 서비스에 필요한 정보 요구사항이 무엇인지 규정되어야 한다. 그리고, 이 요구사항에 부합되는 딥러닝 학습 데이터를 수집하고, 학습을 통해 모델을 생성해야 한다.

딥러닝 모델 사용은 생성된 모델을 로딩해, 현장 데이터를 모델에 입력하고, 출력 결과를 데이터베이스에 저장한다. 각 프로세스 별 기능의 역할은 Table 2에 상세화하였다.

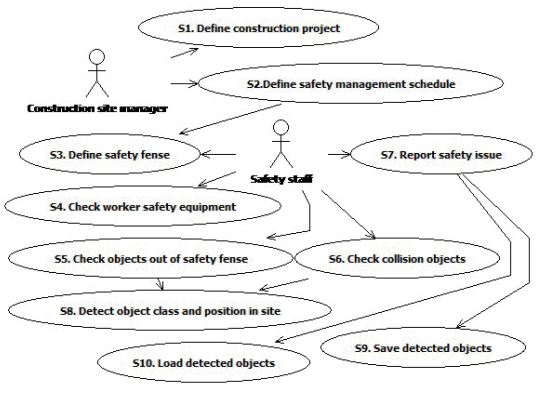

Figure 4. Construction site management use cases (safety)

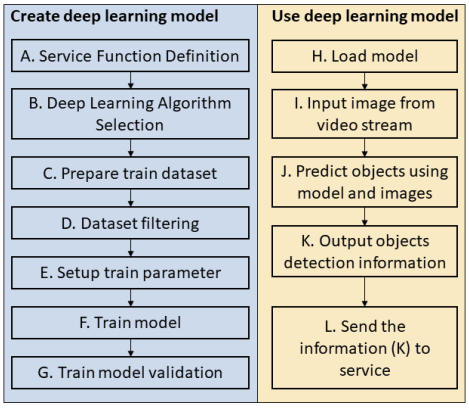

Figure 5. Deep learning creation and usage process

Table 2. Process step function description

Step	Function description
A. Service Function Definition	For construction site object detection, service function definition such as real-time design, object classes, object detection environment, data aquisition sensor etc.
B. Deep Learning Algorithm Selection	To detect object from image dataset, selecting deep learning algorithm
C. Prepare train dataset	Preparing train dataset to build deep learning model
D. Dataset filtering	Filtering dataset to remove noise data
E. Labeling train dataset	Labeling train dataset considering object classes
F. Train model using parameters	Setup train parameter to fit the model and Training model using dataset and parameters.
G. Train model validation	Validating train model and update it
H. Load model	Loading model for using it in service
I. Input image from video stream	Inputting image dataset to model for prediction
J. Predict objects using model and images	Predicting object classes including information
K. Output objects detection information	Outputting predicted object classes and information
L. Send the information (K) to service	Sending the information to service

딥러닝의 가장 큰 장애물은 대용량 학습 데이터를 준비하는 것이다. 이는 노동집약적인 많은 비용이 소모되는 작업이다. 이를 고려해, 딥러닝 모델 개발 시 유사 객체가 학습된 모델을 사용하는 전이학습(transfer learning)을 사용하도록 한다. 전이학습은 새로운 모델로 학습할 때보다 이미지 분포가 더 크기 때문에 효율적 학습이 가능하다. 전이학습은 풀고자 하는 문제와 비슷하면서 사이즈가 큰 데이터로 이미 학습이 되어 있는 모델이다. 이를 통해, 훈련 모델 데이터셋 수집 및 라벨링 작업 비용을 줄인다.

4. 프로토타입 개발 및 고려사항 도출

4.1 프로토타입 개발

앞서 정의된 프로세스에 따라 개발된 프로토타입 개발 과정을 플랫폼 아키텍처 설계에 반영한다. 프로토타입 개발 결과를 바탕으로, 건설 현장 작업자 및 장비 모니터링에 필요한 실시간 객체 탐지용 딥러닝 플랫폼 아키텍처를 정의한다.

딥러닝 모델 개발을 위해, 건설 현장에서 주로 많이 확인할 수 있는 객체들인 건설 중장비와 작업자를 학습 대상으로 한다. 학습 알고리즘은 실시간 객체 탐지에 높은 성능을 보이는 YOLO(v3)를 적용하였다.

학습 데이터는 전이학습을 위해 구글에서 '건설 중장비', '건설 작업자' 키워드를 통해 각각 446개, 450개 데이터셋을 검색하여 생성하였다. 생성된 데이터셋에는 이미지가 없는 노이즈 데이터 등이 포함되어 있다. 이런 데이터들은 필터링 코드를 작성해 삭제되었다. 이후, 육안 검사를 통해, 키워드와 전혀 다른 데이터들도 삭제되었다. 이 과정은 전체 20분이 소요되었다. 이후, 학습용 데이터셋 구조로 변환하기 위해, 데이터셋 네이밍(naming)과 폴더 구조를 생성한다(Figure 6).

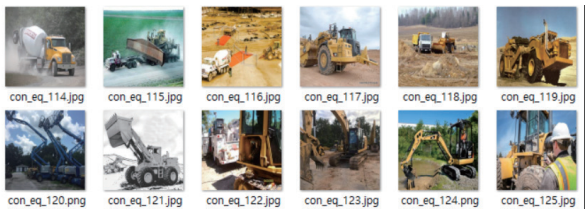

Figure 6. Preparing train dataset (Google)

학습 데이터를 준비한 후, 데이터파일은 train, test, validate로 분류하였다. 본 경우에는 각각 train에 70%, validate에 20%, test에 10%를 할당한다.

학습 데이터를 준비한 후, 데이터 라벨링(labeling)을 수행하였다(Figure 6). 이 도구를 이용해 각 이미지 별 경계 박스 설정을 하였으며, 평균 작업 시간은 이미지 내에 존재하는 객체 개수에 따라 1~10초 정도 소요되었다. 본 학습 데이터셋의 경우, 한 장당 평균 2~5개의 경계박스가 선택되었다. 실제, 중장비 이미지 데이터 라벨링 작업은 51분 소요되었다.

중장비 라벨링 속도(Labeling Speed)는 평균 10장/분이고, 6초당 1장 작업 처리 속도이다. 건설 작업자 라벨링 작업은 60분이 소모되었다. 라벨링 속도는 7.5장/분 속도이다. 참고로, 이미지에 포함된 객체 클래스 개수, 라벨링 난이도 및 라벨링 도구 사용 편의성에 따라 이 작업 속도는 달라질 수 있다.

Figure 7. Labeling process results (Equipments and workers, Google)

모델 학습은 전체 데이터셋 크기가 1,000개 미만이므로 전이 학습 기법을 사용하였다.

4.2 결과 분석

학습 모델 실행 결과는 학습 객체 마스크(mask) 크기, IoU(Intersectioin over Union), 클래스 값으로 출력된다. IoU 는 실제와 예측된 경계박스의 교차율을 뜻하며, 1에 가까울수록 좋다.

$$IoU = Aarea\ of\ Overlap\ /\ Area\ of\ Union \quad (1)$$

클래스 값은 1에 가까울 수록 학습이 잘되고 있다는 의미이다. 본 사례에서 학습이 완료되었을 때, 클래스 값은 1.0에 수렴하였고, IoU값은 0.79 ~ 0.85정도를 나타내었다(Figure 8).

```
Region 82 Avg IOU: 0.798036, Class: 0.999875
5R: 1.000000, .75R: 0.875000,  count: 8
Region 94 Avg IOU: 0.783650, Class: 0.996331
5R: 1.000000, .75R: 0.600000,  count: 5
Region 106 Avg IOU: -nan, Class: -nan, Obj:
75R: -nan,  count: 0
Region 82 Avg IOU: 0.850576, Class: 0.999021
5R: 1.000000, .75R: 1.000000,  count: 4
```

Figure 8. Dataset train results

이 사례에서 학습 시간은 12시간 30분 소모되었으며(Intel i7, GPU 1070 8GB), 생성된 학습 모델 크기는 346.3MB였다.

학습 모델을 통한 객체 탐지 성능을 확인해 보기 위해, 건설 현장 영상을 구한 후, 학습 모델에 입력해 보았다. 그 결과는 Figure 9와 같다.

중장비 객체 탐지의 경우 정확도는 각 객체별로 89%, 87%, 78%이다. 작업자의 경우, 93~62% 정확도를 보이며, 평균 정확도는 81.1%이다. 참고로, 정확도가 낮게 인식되는 객체는 사진에서 일부만 보이는 작업자이다.

객체 탐지 속도는 초당 평균 2.6 FPS(Frame Per Second)이다.

Figure 9. Predicted object classes results (Google)

결과를 확인해 보면, 다음을 확인할 수 있다.

1. 학습 데이터 준비 과정 및 도구 정의 필요

라벨링은 라벨링 개발자에 따라 그 결과가 달라질 가능성이 있다. 가급적 일관된 결과물을 위해, 목적에 맞는 객체 라벨링 지침 및 도구가 적절히 제공될 필요가 있다.

2. 모델 예측 결과 신뢰성 관리 필요

모델 예측 결과의 신뢰성을 확보하기 위해, 평가 메트릭스를 관리할 필요가 있다.

3. 실시간성과 정확도간 트레이드오프(tradeoff) 고려

실시간성과 정확도는 딥러닝 모델과 학습 데이터양에 의존된다.

4. 서비스에 따른 적절한 실시간성 고려

실시간성은 딥러닝 모델 종류, 하드웨어 지원 사항 등에 따라 달라질 수 있다. 불필요하게 높은 실시간성은 많은 비용을 요구한다. 요구사항에 맞는 수준의 실시간성을 고려할 필요가 있다.

4.3 플랫폼 개발 고려사항 도출

앞서, 정의한 유스케이스, 요구사항을 일반화하고, 딥러닝 알고리즘 분석, 구현 결과를 바탕으로 건설 현장 관리 실시간 객체 탐지를 위한 플랫폼 개발 시 고려사항을 다음과 같이 도출한다.

C1. 서비스 가변성

정의한 유스케이스는 공사현장 안전관리에 대한 것이고, 이 유스케이스만 해도 공사현장 및 안전관리 요구사항에 따라 검출해야 하는 객체, 정보, 서비스 흐름이 다르다. 그러므로, 서비스 요구사항에 대한 가변성을 고려해야 한다.

C2. 딥러닝 모델 가변성

C1에 따라 필요한 딥러닝 모델도 가변적일 수도 있다. 딥러닝 모델 알고리즘은 서비스 요구사항에 따라 손쉽게 선택 및 관리될 수 있어야 한다.

C3. 딥러닝 프로세스 지원

딥러닝 모델 선택, 학습, 실행, 평가, 저장 및 로딩을 지원하며, 학습 데이터의 라벨링, 필터링을 건설 현장 객체 데이터셋 특성을 고려하여 지원될 수 있어야 한다.

C4. 실시간 성능

실시간성에 대한 정의는 서비스 요구사항에 따라 달라진다. 실시간성이 초당 혹은 분당 객체 예측 속도로 정의되느냐에 따라 입력되는 데이터셋의 크기, 사용되는 GPU(Graphic Process Unit) 및 메모리 등 스펙이 달라질 수 있다.

C5. 서비스와 딥러닝 모델간 상호운용성

서비스는 딥러닝 모델 추론 시 필요한 입력이나 출력 데이터를 사용한다. 만약, 상호간 입출력 형식이 각각에 따라 다르다면, 데이터 상호교환이 어렵고, 딥러닝 모델을 재활용하기도 어려워진다. 그러므로, 이에 대한 표준적인 입출력 형식 및 구조가 필요하다.

C6. 서비스 지원 API

입출력 형식 뿐 아니라, 서비스에서 호출하는 함수인 API(Application Program Interface)를 제공할 수 있어야, 서비스가 딥러닝 모델 및 서비스 운영에 필요한 플랫폼 기능을 사용할 수 있다.

C7. 클라우드 지원

서비스와 딥러닝 모델은 클라우드 기반으로 지원되어, 하드웨어와 독립되어 있어야 한다. 하드웨어는 서비스 요구사항에 따라 변경될 수 있어야 한다.

4.4 실시간 건설 현장 객체 탐지용 딥러닝 플랫폼 아키텍처 도출

앞서 도출한 플랫폼 개발 고려사항을 바탕으로 가변성 및 확장성을 고려한 딥러닝 플랫폼 아키텍처를 Figure 10과 같이 정의한다.

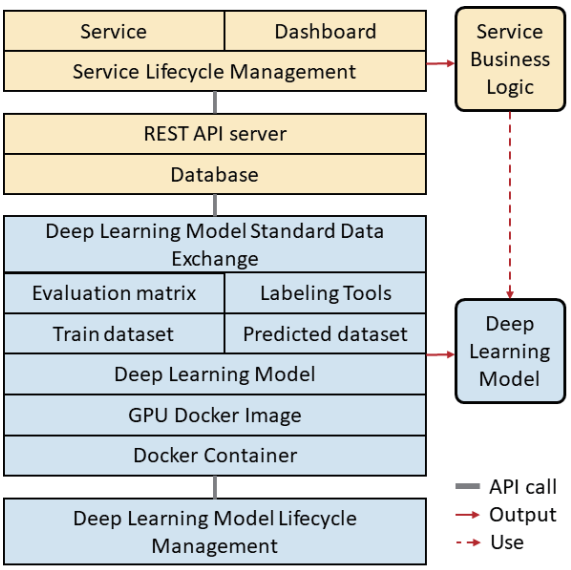

Figure 10. Platform architecture

제안된 플랫폼 아키텍처 구성요소 역할은 다음 표와 같다.

Table 3. Architecture component role definition

No	Component	Consideration	Role
1	Service	C1	Service business logic
2	Dashboard	C1	Dashboard for service information idenfication and management
3	Service Lifecycle Management	C1	Management component for service definition, execution, deletion
4	REST API server	C2, C5, C6	REST API for communication with Deep learning data
5	Database	C1	Database to manage service data
6	Deep Learning Model Standard Exchange	C3, C5	Standard data exchange module to input and output data with service which supports JSON, XML, VOC, COCO format etc.
7	Evaluation matrix	C3	Deep learning model evaluation matrix management module
8	Labeling Tools	C3	Labeling tools for defining labels
9	Train dataset	C3	Train dataset to build deep learning model
10	Predicted dataset	C3	Predicted dataset for evaluating deep learning model
11	Deep Learning Model	C4	Trained deep learning model
12	GPU Docker Image	C4, C7	GPU Docker for supporting GPU hardware
13	Docker Container	C7	Docker container to support and manage various deep learning models
14	Deep Learning Model Lifecycle Management	C2, C3	Deep learning model lifecycle management to support model load, save

제안된 플랫폼은 C1–7 요구사항을 구현하기 위해, Table 3에 정의된 컴포넌트를 하부 구조로 가진다. 건설 현장 관리 서비스 개발을 위해서는 딥러닝 모델 및 서비스 가변성을 지원하는 기능이 필요하므로, 모델과 서비스를 로딩, 저장, 실행 기능을 갖는 생애주기 관리 컴포넌트가 필요하다. 서비스 요구사항에 맞는 딥러닝 모델 및 하드웨어 선택을 위해 가상 이미지 관리자인 도커(Docker)를 사용한다. 딥러닝 모델과 서비스 간 입출력은 표준화된 REST API와 JSON 같은 산업 데이터 교환 표준을 사용한다.

5. 결론

본 연구는 딥러닝 기술을 이용해, 공사 현장 관리 수행 시 필요한 실시간 객체 탐지 플랫폼 아키텍처를 도출하였다. 이를 위해, 실시간 객체 탐지를 지원하는 딥러닝 모델을 조사 및 분석하였고, 이를 바탕으로 실시간 건설 현장 객체 탐지에 필요한 딥러닝 모델 개발 프로세스를 정의하였다. 정의된 프로세스를 바탕으로, 건설 현장 객체를 학습 및 탐지하는 프로토타입을 개발 한 후, 그 결과를 통해 플랫폼 개발 고려사항과 아키텍처를 도출하였다. 도출 결과 C1~7의 고려사항과 14개의 핵심 컴포넌트를 정의하였다.

건설 분야에서 딥러닝 모델을 사용한 비전 기반 건설 현장 관리는 아직 초기 단계이다. 향후, 연구 내용을 바탕으로 이미지 기반 건설 현장 관리 딥러닝 플랫폼과 서비스를 개발하여 효과를 검증하고, 건설 현장 관리에 적합하도록 학습 데이터 취득 및 모델 개발 프로세스를 개선할 계획이다.

감사의 글

This research was supported by a grant (21AUDP-B127891-05) from the Architecture & Urban Development Research Program funded by the Ministry of Land, Infrastructure and Transport of the Korean government.

References

Arabi, S., Haghighat, A., Sharma, A. (2020). A deep-learning based computer vision solution for construction vehicle detection. Computer Aided Civil and Infrastructure Engineering, 35(7), pp. 753-767.

Gu, H. M., Seo, J. H., Choo, S. Y. (2019) A Development of Facade Dataset Construction Technology Using Deep Learning-based Automatic Image Labeling, Journal of the Architectural Institute of Korea Planning & Design, 35(12), pp. 43-54.

Journal of Advanced Information Technology and Convergence, 17(7), pp. 55-63.

Kim, G. H., Seo, B. W. (2019) Intelligent construction video management system based on edge computing using deep learning,

Kim, M. H. (2019). Application of Deep Learning Algorithm for Detecting Construction Workers Wearing Safety Helmet Using Computer Vision, The Korean Society of Safety, 34(6), pp. 29-37.

Korea Journal of Construction Engineering and Management, 20(2), pp. 3-12.

Lee, G. R., Han, C. H., Lee, J. B. (2019) The Development of Productivity Prediction Model for Interior Finishes of Apartment using Deep Learning Techniques.

Lee, J. H., Park, J. J., Yoon, H. C. (2020). Automatic Classification of Bridge Component based on Deep Learning, Journal of The Korean Society of Civil Engineers, 40(2), pp. 239-245.

Lee, Y. G., Sun, J. W., Lee, M. J. (2019). Development of Deep Learning Based Deterioration Prediction Model for the Maintenance.

Planning of Highway Pavement, Korea Journal of Construction Engineering and Management, 20(6), pp. 34-43.

Xiao, B., Kang, S. C. (2021). Vision-Based Method Integrating Deep Learning Detection for Tracking Multiple Construction Machines. Journal of Computing in Civil Engineering, 35(2)