Journal of Korean Society of Industrial and Systems Engineering
https://doi.org/10.11627/jksie.2023.46.3.275

Vol. 46, No. 3 : 275-284, September 2023
ISSN : 2005-0461 / EISSN : 2287-7975

Development of Deep Learning-based Land Monitoring Web Service

In-Hak Kong* · Dong-Hoon Jeong** · Gu-Ha Jeong**†

*LX corp. Spatial Information Research Institute, Department of spatial Information Engineering, Pukyong National University
**LX corp. Spatial Information Research Institute

딥러닝 기반의 국토모니터링 웹 서비스 개발

공인학* · 정동훈** · 정구하**†

*LX 공간정보연구원, 부경대학교 지구환경시스템과학부 공간정보공학 전공
**LX 공간정보연구원

Land monitoring involves systematically understanding changes in land use, leveraging spatial information such as satellite imagery and aerial photographs. Recently, the integration of deep learning technologies, notably object detection and semantic segmentation, into land monitoring has spurred active research. This study developed a web service to facilitate such integrations, allowing users to analyze aerial and drone images using CNN models. The web service architecture comprises AI, WEB/WAS, and DB servers and employs three primary deep learning models: DeepLab V3, YOLO, and Rotated Mask R-CNN. Specifically, YOLO offers rapid detection capabilities, Rotated Mask R-CNN excels in detecting rotated objects, while DeepLab V3 provides pixel-wise image classification. The performance of these models fluctuates depending on the quantity and quality of the training data. Anticipated to be integrated into the LX Corporation's operational network and the Land-XI system, this service is expected to enhance the accuracy and efficiency of land monitoring.

Keywords : Land monitoring, Deep learning, Object detection, Web service, Spatial information

1. 서 론

국토모니터링은 다양한 공간정보를 활용해 시계열적인 국토이용변화를 체계적이고 주기적으로 관찰하며 기록하는 과정이다. 이러한 정보는 국토와 도시의 균형적인 발전 및 개발 정책을 마련하는데 필요한 자료로 활용된다. 국토의 광범위한 지역 변화를 주기적으로 파악하기 위해서는 위성영상, 항공사진 등의 영상정보가 중요하게 사용된다.

최근 기술 발전에 따라 공간정보와 원격탐사는 국토모니터링과 토지이용변화의 감지 및 관리에 필수적인 도구로 자리매김하고 있다. 그렇지만, 대량의 공간정보를 처리하고 분석하는 것은 전문 지식이 필요한 복잡한 작업으로 많은 시간이 소요된다. 이를 효율적으로 처리하기 위한 솔루션으로, 인공지능 기술 중에서도 특히 딥러닝이 주목받고 있다[9].

기존의 분석 방식은 데이터로부터 패턴을 찾아내 결과를 도출하는 것이었다. 그러나 딥러닝은 학습과정을 통해 복잡한 패턴을 자동으로 추출한다. 이러한 딥러닝 기술은 공간정보 및 원격탐사 분야에서 객체탐지(Object Detection), 분

Received 18 August 2023; Finally Revised 18 September 2023; Accepted 19 September 2023
† Corresponding Author : research@lx.or.kr

류(Classification), 분할(Segmentation) 등의 다양한 문제 해결에 활용되고 있다.

객체탐지와 분할은 모니터링 분야의 각기 다른 측면에서 중요한 역할을 한다. 객체탐지는 특정 지역의 특정 구조물, 예를 들면 건물이나 비닐하우스의 위치와 존재를 판별한다. 이는 영상에서 건물이나 구조물의 위치를 빠르게 파악해야 하는 상황에 효과적이다. 반면, 분할은 픽셀 단위의 정밀한 분석을 제공하므로, 구조물의 정확한 면적과 형태 측정에 이용된다. 이런 특성 때문에 정밀도가 요구되는 토지피복분류나 식생유형분석 등에서 사용될 수 있다[6].

객체탐지와 분할의 연산 기법에는 차이가 있어서, 목표와 응용 분야의 요구에 맞게 선택하여야 한다. 이러한 선택은 국토모니터링의 특정 목적, 정확도, 처리속도, 실시간 처리의 필요성 등 여러 요인을 고려해 결정되어야 한다. 경우에 따라 두 기법을 함께 사용해 더 좋은 결과를 얻을 수 있다[3].

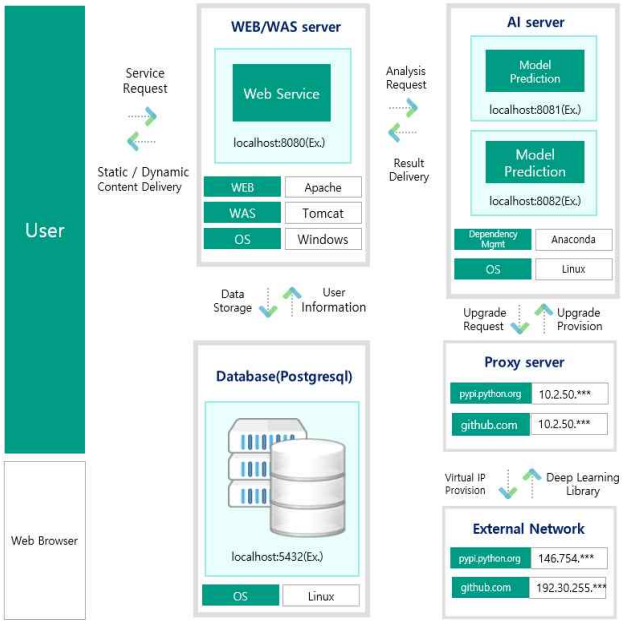
객체탐지와 분할 기술은 국토모니터링에서 중요한 역할을 하지만, 이 기능들을 이용하려면 특정 소프트웨어와 하드웨어 환경이 필요하다. 예를 들어, 딥러닝 모델로는 YOLO, Deeplab V3, Mask R-CNN 등이 있고, 이들은 파이토치, 텐서플로, 케라스와 같은 다양한 플랫폼을 필요로 한다. 또한 각 모델은 특정 라이브러리 버전을 필요로 하기 때문에 설치 시 의존성 문제가 발생할 수 있다. 이로 인해 사용자는 복잡한 설정 과정을 거쳐야 하고, 이는 AI 모델의 접근성을 제한할 수 있다. 게다가, 대부분의 인공지능 모델은 학습과 추론 과정에서 GPU 환경이 필요하며, 개별 사용자가 이 환경을 구축하는 데는 큰 노력이 필요하다[10].

국토모니터링은 효율적인 국토 관리와 계획을 위해 정확히 조사할 필요가 있다. 국토의 지역적 변화를 정확하게 파악하고 관리하는 것이 필수적이다. 과거의 국토모니터링은 육안판독을 이용한 방법이 사용되었고 관심 위치에서의 전후 시점 영상을 비교하여 변화를 탐지할 수 있었다. 하지만 육안판독은 정량적인 변화탐지 정보를 얻기 어려움이 있다. 이러한 문제점을 극복하고자 합성곱 신경망과 같은 딥러닝 기술을 기존 방법의 대안으로 활용할 수 있다[5].

본 연구에서는 시간과 비용이 많이 소요되는 현재의 국토모니터링 업무를 개선하기 위해 딥러닝 및 AI를 활용하는 사용자 시스템을 구현하는 것을 목적으로 한다. 모델만을 제공하는 것은 사용자의 접근성과 효용성에 한계가 있다. 따라서 사용자가 복잡한 시스템 환경 설정이나 분석 과정을 거치지 않고도 AI 분석 모델을 쉽게 활용할 수 있도록 웹서비스 형태로 제공하기 위한 시스템 아키텍처 및 서비스 구성을 설명하고 분석모델을 사용할 수 있는 모듈 개발에 중점을 둔다. 다양한 CNN 모델을 활용하여 항공 및 드론 영상의 분석이 이루어지며, 이를 바탕으로 사용자가 필요한 클래스 정보를 포함한 영상정보를 생성한다. 결론적으로, 본 논문에서는 직관적이고 사용자 친화적으로 접근할 수 있도록 웹서비스 형태로 구현하기 위한 딥러닝 모델, 웹서비스의 구성 및 시스템 아키텍처 설계의 방향성을 제시하고 국토모니터링의 실용성과 효율성을 높이는 방향으로 발전방안을 모색하고자 한다.

2. 웹서비스의 구성

본 연구는 영상을 이용한 육안 판독하는 국토모니터링의 시간과 노력을 절감하기 위해 LX 업무담당자가 별도의 환경설정 없이 편리하게 딥러닝을 활용할 수 있도록 웹서비스를 제공하는데 목적이 있다. 이를 위한 웹서비스의 구성은 <Figure 1>과 같이 AI 서버, WEB/WAS 서버, 그리고 DB 서버의 세 부분으로 구분하였으며, 각 서버는 고유한 기능과 역할을 수행한다.



<Figure 1> Configuration Diagram of the System

2.1. AI 서버

AI 서버의 주요 역할은 딥러닝 모델에 대한 추론을 수행하는 것이다. 이 서버는 Python 기반의 웹 프레임워크인 Flask로 구축되었다. Flask는 사용자의 요청을 효과적으로 처리하며, 해당 요청에 따른 모델 분석을 실행하고 결과를 사용자에게 반환한다.

딥러닝 분야에서는 대부분의 모델이 Python으로 구현되어 있다. 또한, 데이터 분석, 가공, 그리고 모델 학습에 필

요한 라이브러리들도 Python 환경에서 풍부하게 제공된다. 이러한 이유로, Flask와 Python 환경을 기반으로 하는 AI 서버는 딥러닝 개발의 유연성과 확장성을 강화할 수 있다[7].

2.2 WEB/WAS 서버

WEB/WAS 서버는 사용자에게 인터페이스를 제공하며, AI 및 DB 서버와의 연동을 담당한다. 이 서버는 Java 기반의 오픈소스 프레임워크인 Spring Boot로 구현되었다. Spring Boot는 IoC(Inversion of Control)와 AOP(Aspect-Oriented Programming)와 같은 프로그래밍 패러다임에 기반을 둔다. 이런 패러다임을 사용함으로써, 시스템의 유연성을 향상시키며, 서로 다른 모듈 간의 결합도를 낮출 수 있다. 또한, 공통기능의 모듈화를 통해 코드 중복을 최소화할 수 있다[15].

2.3 DB 서버

데이터베이스 관리의 효과성을 높이기 위해 관계형 DB에서 JPA(Java Persistence API)를 도입하였다. JPA는 ORM(Object Relational Mapping) 기능을 통해 Java의 Model 객체와 데이터베이스 간의 매핑 작업을 간소화한다. 이러한 기능 덕분에 개발자는 복잡한 SQL 쿼리를 직접 작성할 필요 없이, Java 클래스와 메서드를 활용하여 데이터베이스 작업을 수행할 수 있다. 더욱이, JPA의 API는 다양한 데이터베이스 간의 마이그레이션을 지원하며, 이로 인해 데이터베이스의 독립성도 확보할 수 있다[14].

2.4 내·외부망 연계

정보보안을 위해 내·외부망 연계 시 내부망과 외부망 사이에 프록시 서버를 배치한다. 이 프록시 서버는 실제 내부망의 IP 주소를 숨기고 자신의 IP 주소를 대신 사용해 요청을 전송한다. 이 방식으로 외부망은 내부망의 실제 IP 주소를 알 수 없게 되므로 직접적인 공격을 방지할 수 있다.

3. 국토 모니터링을 위한 딥러닝 모델

본 연구에서는 국토의 지형 변화와 구조물의 위치 및 분포를 딥러닝을 통해 정확하게 파악하고, 그 결과를 시각적으로 표현하는 시스템을 구현하는 것이다. 연구를 진행하는 과정에서 다양한 딥러닝 모델들을 검토하였으며, 세 가지 주요 모델을 최종적으로 선정하고 적용하였다.

첫 번째로, 지형의 세부적인 특징과 영역 분류를 위해 DeepLab V3를 사용하였다. 이 모델은 영상 분할에 특화되어 있어, 영상의 각 픽셀이 어느 범주에 속하는지 정밀하게 분류한다.

두 번째로, 국토 내 다양한 객체를 실시간으로 탐지하기 위해 YOLO(You Only Look Once) 모델을 채택하였다. 이 모델은 영상을 한 번만 스캔하여 빠른 시간 내에 다양한 객체를 탐지하는 능력이 있다.

세 번째로, 회전된 객체나 복잡한 형태의 객체탐지를 위해 Rotated Mask R-CNN을 선택하였다. 이는 기존의 Mask R-CNN을 기반으로 하면서, 회전된 객체탐지 성능을 강화한 버전이다.

이렇게 선정된 모델들은 각각 고유한 특징을 가지며, 국토모니터링 구현에 중요한 역할을 한다.

3.1 YOLO(You Only Look Once)

YOLO는 실시간 객체탐지를 위한 딥러닝 모델이다. 전통적인 방식에서는 후보영역 추출 후 객체인식을 별도로 수행했으나, YOLO는 영상을 그리드로 분할한 뒤 각 그리드에서 바운딩 박스와 클래스 확률을 동시에 예측한다. 이 방식은 영상을 한 번만 훑어보기 때문에 처리 속도가 빠르다는 장점을 가진다[12].

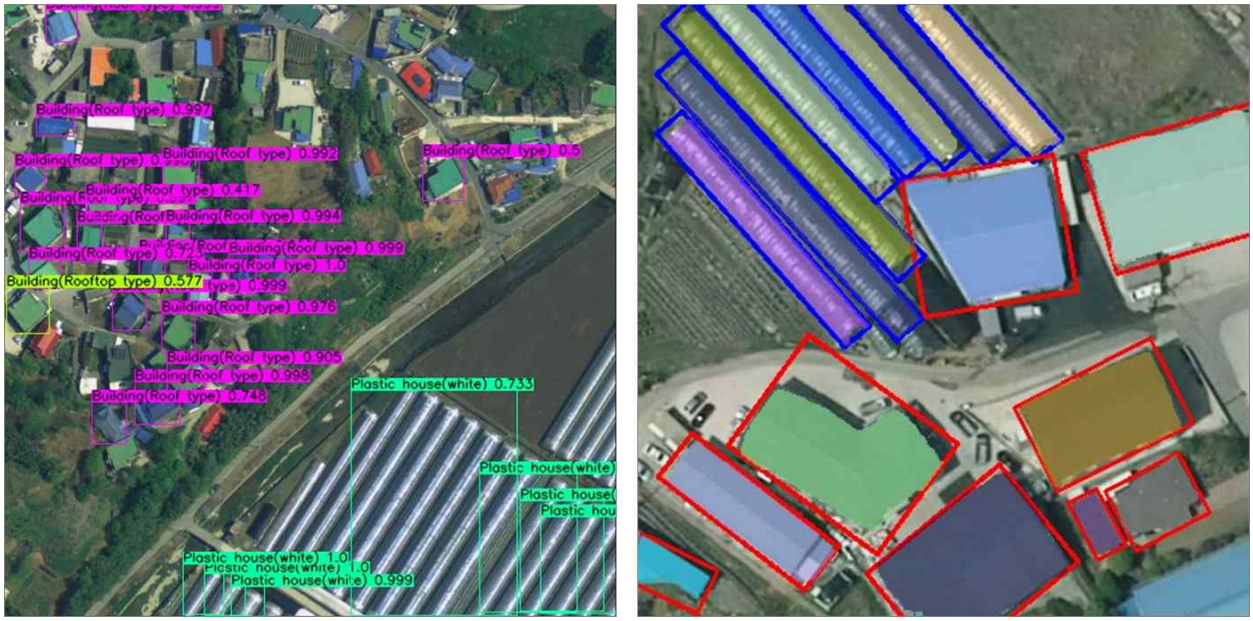
국토 모니터링은 지리적 및 환경적 변화를 실시간으로 파악하는 작업이다. YOLO의 빠른 처리 속도는 산사태, 화재, 도로 상황과 같은 다양한 객체와 현상의 실시간 탐지에 적합하다. 특히, 드론이나 위성 영상에서의 변화를 실시간으로 탐지하는 데 있어 YOLO는 널리 활용된다.

그러나 YOLO는 작은 객체의 탐지성능이 떨어질 수 있으며, 서로 겹치는 객체나 특정 각도에서의 객체탐지에 어려움을 겪을 수 있다. 이러한 한계점을 인지하고 있어야 하며, 특정 상황에서는 다른 모델과의 조합을 고려하는 것이 중요하다.

3.2 Rotated Mask R-CNN

국토 모니터링은 효과적인 토지 사용 관리, 자연 재해 대응, 그리고 환경보호를 위한 필수적인 작업이다. 최근의 기술 발전으로 인해 딥러닝 기반 객체탐지 기술이 모니터링 업무에 큰 도움을 주고 있다. Rotated Mask R-CNN은 Mask R-CNN을 기반으로 개발되어 회전된 바운딩 박스를 통해 회전된 객체의 픽셀 수준에서 분할이 가능하다. 이런 기능은 위성 영상처럼 회전된 객체가 많은 복잡한 영상의 객체 탐지와 분류에 특히 유용하다[13].

Rotated Mask R-CNN은 국토모니터링에서 뛰어난 정확도를 제공하는 큰 장점을 가지고 있다. 그렇지만, 이런 성능을 유지하려면 많은 양의 고품질 훈련 데이터와 상당한



<Figure 2> Object Detection Results Using Different Models (Left: YOLO. Right: Rotated Mask R-CNN)

계산 자원이 필요하다는 조건이 있다.

<Figure 2>에서는 YOLO 및 Rotated Mask R-CNN 모델을 이용하여 항공영상에서 건물과 비닐하우스를 추론한 결과를 보여준다. YOLO 모델은 바운딩 박스 형태로 객체를 구분하고 Rotated Mask R-CNN은 바운딩 박스 뿐만 아니라 의미론적 분할을 수행하여 각 객체의 경계를 픽셀 수준으로 분류한 결과를 나타낸다.

3.3 DeepLab V3

DeepLab V3는 영상 분할을 위한 딥러닝 모델이며, 각 영상 픽셀이 어떤 카테고리에 속하는지 예측하는 데 특화되어 있다. ASPP(Atrous Spatial Pyramid Pooling)는 이 모델의 주요 특성으로, 다양한 스케일의 정보를 캡처하기 위해 서로 다른 딜레이션 비율을 가진 컨볼루션을 동시에 사용한다. 먼저, 기본 CNN 구조를 통해 영상에서 피처를 추출하고, 그 다음에 추출된 피처 맵에 ASPP를 적용하여 마침내 세그멘테이션 마스크를 만든다[2].

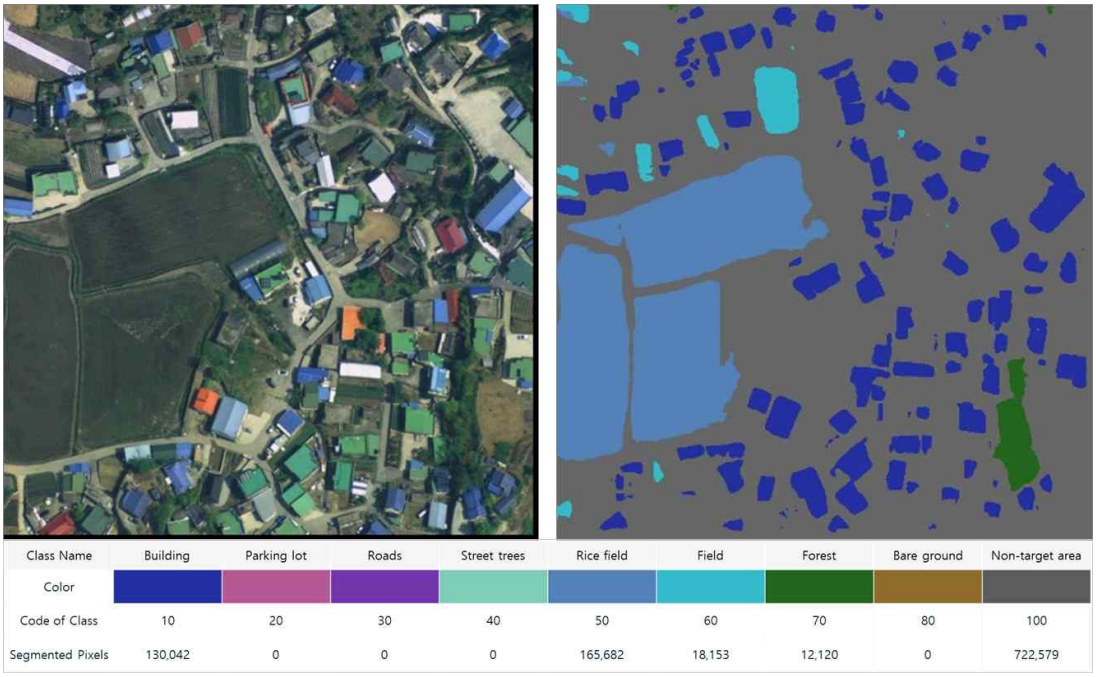
국토 모니터링은 환경 변화, 도시 발전, 자연 재해와 같은 요소들을 지속적으로 관찰하고 분석하는 활동이다. DeepLab V3는 이런 모니터링 활동에 유용하게 사용될 수 있으며, 도시 발전 추적, 농업 영역 모니터링, 자연재해 탐지 등 다양한 적용 사례를 가지고 있다. 위성영상이나 드론으로 촬영한 영상을 입력으로 사용하여 국토의 변화를 실시간으로 모니터링하는 것이 가능하다.

이 모델은 특히 다양한 스케일의 객체나 패턴을 정확하게 포착하는 능력을 지녔다. 그렇지만, 충분한 학습 데이터 필요성, 실시간 처리의 한계 등 여러 단점도 있다. 이러한 한계를 극복하기 위한 추가적인 최적화가 필요하다.

<Figure 3>에서는 DeepLab V3 기반의 토지피복 분류 모델을 활용하여 추론한 결과를 보여준다. 왼쪽 영상은 원본 항공영상이며, 오른쪽 영상은 해당 사진의 토지피복 분류 결과를 나타낸다. 토지피복 분류에서는 건물, 주차장, 도로, 가로수, 논, 밭, 산림, 나지로 8개의 클래스로 추론하였다.

또한 <Figure 4>는 DeepLab V3 모델을 기반으로 2018년과 2020년의 특정 지역의 항공영상에 대한 변화탐지를 수행한 결과를 보여준다. 왼쪽 위의 영상은 변화 전 영상이며 오른쪽 위의 영상은 변화 후 영상이다. 오른쪽 아래 영상은 2018년과 2020년의 건물에 대한 변화탐지 결과를 나타내어 건물이 건축된 부분의 변화를 파악할 수 있다. 오른쪽 아래 영상은 도로를 대상으로 변화탐지를 수행하여 신규 도로 건설이나 기존 도로의 변경된 부분을 확인할 수 있다.

<Figure 5>에서는 DeepLab V3 기반으로 2018년과 2020년의 기존 건물과 논, 밭이 함께 혼재한 지역의 항공영상을 분석하였다. 결과적으로, 논과 밭에서 건물로 변화하는 부분은 잘 탐지되었다. 그러나 기존 건물이 위치한 영역에서는 노이즈 픽셀들이 관찰되었다. 이는 항공영상의 촬영 각도에 따른 정사영상의 건물 정합오차의 영향으로 보이며, 따라서 이러한 오차를 수정하거나 최소화하기 위한 추가적인 처리 로직의 개발이 필요하다고 판단된다.



Class Name	Building	Parking lot	Roads	Street trees	Rice field	Field	Forest	Bare ground	Non-target area
Color									
Code of Class	10	20	30	40	50	60	70	80	100
Segmented Pixels	130,042	0	0	0	165,682	18,153	12,120	0	722,579

<Figure 3> Land Cover Classification Using DeepLab V3. Left: (Original image. Right: Inferred Segmentation Image)

<Figure 4> Analysis of Land Cover Changes Using DeepLab V3. (Top left: Image of 2018. Top right: Image of 2020. Bottom left Detected changes in buildings. Bottom right: Detected changes in roads)

<Figure 5> Analysis of land cover changes using DeepLab V3 Based on Mixed Trrain Images. (Top left: Image of 2018. Top right: Image of 2020. Bottom center: Detected changes in buildings)

3.4 학습자료 및 모델 성능 평가

<Table 1>에서는 모델에 대한 학습자료에 대한 정보를 보여주며 각 모델은 LX공사 내부 사업과 AI Hub에서 제공하는 자료와 모델을 사용하였다. 모델 학습의 평가는 객체탐지 및 분할에서 모델의 성능을 평가하기 위한 지표로 mAP(mean Average Precision)과 Accuracy를 사용했다. AP(Average Precision)은 객체탐지에서 모델의 성능을 평가하기 위한 지표로 사용된다. AP는 모델의 정밀도(Precision)과 재현율(Recall) 사이의 관계를 평균화하여 제공한다. 정밀도와 재현율은 식 (1)과 식 (2)를 통해서 구할 수 있으며 <Table 2>와 같은 오차행렬(Confusion matrix)

<Table 1> Comparison of Deep Learning Model Configurations for Image Analysis

	Rotated Mask R-CNN	YOLO v3	DeepLab V3
Result Type	Instance Segmentation	Object Detection	Semantic Segmentation
Tile Size	410 × 410	1024 × 1024	512 × 512
Number of Classes	2[1)]	4[2)]	8[3)]
Number of Objects	16,438	3,400,000	-
Training(Images)	3,027	154,549	13,520
Validation(Images)	1,911	121,777	1,690
Testing(Images)	-	118,847	-
Evaluation Metrics	mAP(0.50)	mAP(0.75)	Accuracy(0.91)
Data Format	COCO	YOLO	Segmentation Mask

Note: 1) Building, Greenhouse
2) Building (roof type), Building (rooftop type), Greenhouse (white), Greenhouse (black)
3) Building, Parking lot, Road, Street tree, Rice field, Field, Forest, Bare ground.

을 이용하여 구해진다. 정밀도는 참으로 예측한 결과에 대해서 예측값과 실제값이 참으로 일치하는 비율을 뜻하며 재현율은 실제 참값 중에서 예측 모델이 참으로 예측한 비율을 뜻한다. AP는 정밀도-재현율 곡선(Precision-Recall Curve) 그래프를 그렸을 때 커브 아래의 면적을 나타내며, 식(3)과 같이 표현될 수 있다. 여기서 $p(r)$은 재현율 r에서의 정밀도를 의미한다. 실제 계산에서는 정밀도-재현율 곡선 아래의 면적을 이산적으로 근사하여 계산하게 된다[11]. 정확도(Accuracy)는 전체 픽셀 중 올바르게 예측된 픽셀의 비율을 나타내고 식 (4)와 같이 구할 수 있다.

<Table 2> Confusion Matrix

		Predicted	
		Positive	Negative
Actual	Positive	TP (True Positive)	FN (False Negative)
	Negative	FP (False Positive)	TN (True Negative)

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$AP = \int_0^1 p(r)dr \quad (3)$$

$$Accuracy = \frac{TP+TN}{FP+FN+TP+TN} \quad (4)$$

Rotated Mask R-CNN은 인스턴스 세그멘테이션 방식으로 항공영상을 410×410 크기의 영상 패치로 분할하여 학습을 수행하였으며 대상 객체는 건물과 비닐하우스를 대상으로 하였다. 3,027장의 학습영상과 1,911장의 검증 영상을 통해서 학습 및 검증을 수행한 결과 mAP 0.50로 평가되었다[8]. 해당 모델에서 사용했던 학습자료 양은 YOLO v3보다 적은 양이었다. 따라서 학습자료를 구축하여 학습할 경우 모델의 성능 향상 있을 것으로 예상된다. YOLO는 Object detection 방식으로 1024×1024 크기의 영상 패치로 분할하여 학습을 수행하였으며 건물은 지붕형과 옥상형, 비닐하우스는 흑과 백을 구분하여 대상 객체를 구분하였고 154,549장의 학습영상과 121,777장의 검증 영상, 118,847장의 테스트 영상을 통해서 학습 및 검증을 수행한 결과 mAP는 0.75로 평가되었다[4]. 현재 YOLO v3 이후로도 다양한 버전의 YOLO 모델이 출시가 되었기 때문에, 이후 버전의 YOLO 모델을 적용했을 때 성능이 어떻게 변화하는지에 대한 연구가 필요하다. DeepLab V3는 Semantic Segmentation 방식으로 512×512 크기의 영상패치로 분할하여 학습을 수행하였고 건물, 주차장, 도로, 가로수 등 8종을 대상으로 학습을 수행하였으며 13,520장의 학습영상과 1,691장의 검증 영상을 통해서 학습 및 검증을 수행한 결과 정확도는 0.91로 평가되었다[1]. 최근에는 DeepLab V3를 포함한 여러 의미론적 분할 모델들이 발표되었다. Transformer 구조를 기반으로 한 SegFormer와 같은 모델들은 전역적인 정보의 활용하여 추론 성능을 높이려는 연구가 진행되고 있으므로 DeepLab V3와 SegFormer를 비롯한 최신 모델들 사이의 성능 비교 연구가 필요하다.

YOLO의 학습자료의 경우 154,549장, Rotated Mask R-CNN은 3,027장으로 차이를 보이며, 이러한 이유로 YOLO는 mAP 0.75, Rotated Mask R-CNN는 0.50로 정확도 차이를 보였을 것으로 판단된다. 따라서 YOLO로 구축된 모델이 다양한 조건과 배경에서 객체를 학습하고 일반화 성능이 높을 것으로 보이며 Rotated Mask R-CNN의 경우 추가 학습자료를 구축하여 다양한 패턴과 상황을 학습한다면 정확도가 향상될 것으로 예측된다.

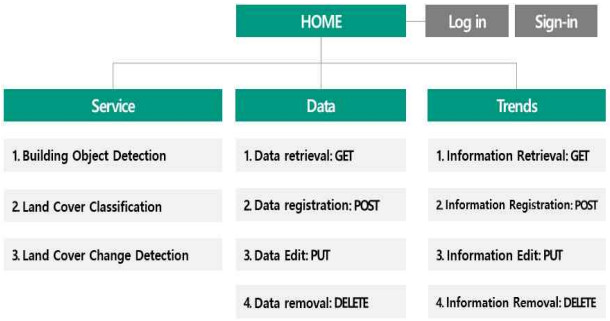
YOLO, Rotated Mask R-CNN 그리고 DeepLab V3를 각각 다른 방식의 학습모델을 사용하여 영상 내 정보를 추출한다. YOLO의 경우 One-Stage 방식을 통해 객체의 위치와 분류를 동시에 수행하기 때문에 실시간 처리가 필요하거나 대규모 데이터를 처리하는 경우 사용할 수 있으며, Rotated Mask R-CNN은 Two-Stage 방식을 통해 영상 내의 객체 위치를 추정하고 정확한 객체 분류 및 바운딩 박스를 결정하게 된다. 이 경우 복잡하게 배열된 건물이나 인프라 같은 배경에서의 객체의 형태와 위치를 파악할 때 사용할 수 있으며 국토모니터링에서 필요한 정확도, 처리속도, 복잡도 등의 요소를 고려하여 적절한 방식을 선택해야 할 필요성이 있다. DeepLab V3는 영상의 각 픽셀을 특정 범주로 할당하는 방식으로 국토의 산림, 주차장, 논, 밭 등을 분류하고 지도화 하는데 사용할 수 있으며 이러한 특성을 기반으로 변화탐지를 수행하였다.

4. 시스템 구성

4.1 시스템 아키텍처 설계

웹 서비스 설계 및 구현은 Spring boot와 Flask를 활용하였으며 AI 모델 관련 서비스, 데이터, 트랜드 3가지 주요 부분으로 <Figure 6>과 같이 구성하였다. 서비스 범주에서는 복잡한 딥러닝 모델과 하드웨어 환경을 간소화하여 국토모니터링을 위한 객체 탐지 및 분할 기술을 쉽게 제공한다. 또한 데이터 범주에서는 국토모니터링에 필요한 항공 및 드론 영상 기반의 학습데이터를 효과적으로 관리하고 공유할 수 있도록 설계하였다. 또한 데이터의 추가, 수정, 조회, 삭제가 자유롭게 가능하도록 하였다.. 트랜드 범주

는 AI의 최신 기술을 공유할 수 있으며, 현재의 트렌드를 파악하고 필요한 썸네일 영상 등을 쉽게 다운로드할 수 있도록 지원한다. 이 부분 역시 CRUD(Create, Read, Update and Delete) 기능을 구현했다.



<Figure 6> User-Centric Service Overview

서버의 구성과 주요 기능은 <Table 3>과 같이 구성된다. AI 서버는 Flask와 Miniforge 프레임워크를 사용하여 파이썬 기반의 AI 서버를 구축하였으며 텐서플로우와 파이토치 기반의 딥러닝 추론 기능을 제공하며 Spring boot 기반의 웹 서비스 서버에서 요청한 영상에 대한 객체탐지 및 토지피복 분류, 변화 탐지 추론을 수행하고 반환하도록 구성했다. 또한 DB 서버는 PostgreSQL을 사용하여 데이터를 저장 및 관리한다.

<Table 4>에서는 웹 서비스 주요 기능의 설명과 API 엔드포인트에 대한 상세정보를 나타내며 서비스의 확장성과 유연성을 위해 RESTful 아키텍처를 기반으로 구성했다. RESTful 아키텍처는 자원 지향적 구조로서 각 자원에 대한 URL 엔드포인트를 중심으로 구성되어 있다. 엔드포인트는 자료에 대한 고유한 접근포인트를 제공해주고 표준 HTTP (HyperText Transfer Protocol) Method를 통해 자원에 접근한다.

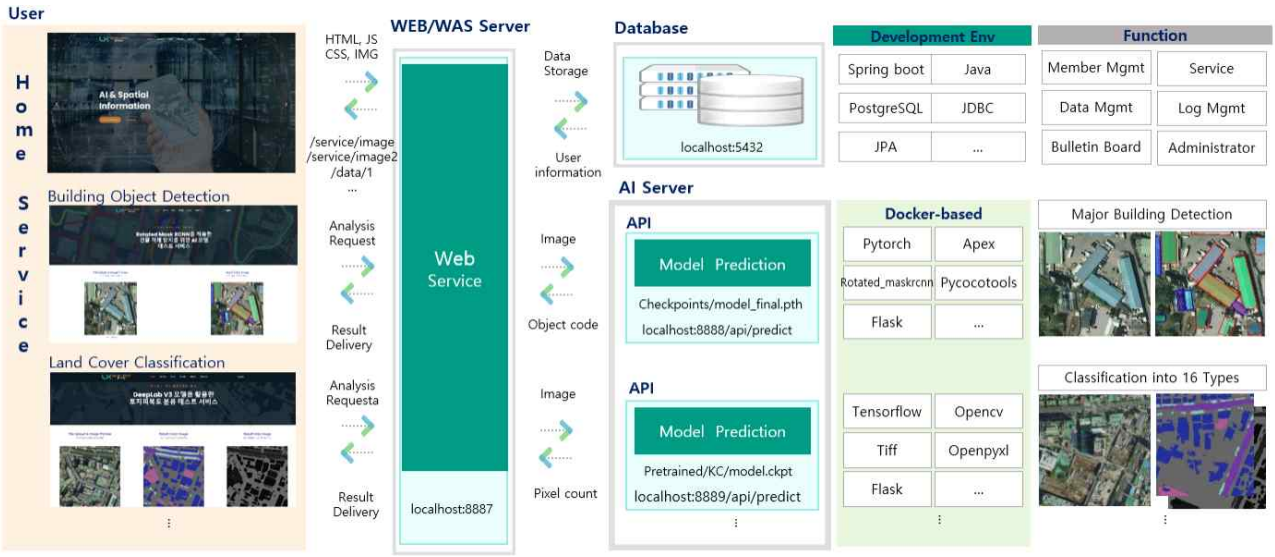
<Table 3> Summary of Server Configurations and Key Features

Server Type	Framework	Programming Language	Hardware Specifications	Key Features	Libraries
AI	Flask, Miniforge	Python	PowerEdge R750 (32Core, 512GB, A100 80GB)	Deep Learning Inference	Tensorflow, PyTorch
WEB/WAS	Spring boot	Java	Virtualized PowerFlex R640 (6Core, 16GB)	User Interface, Business Logic	Maven, JPA, JSP, Vue.js, Tomcat
DB	PostgreSQL	SQL		Data Storage	PostgreSQL

<Table 4> Main API Endpoints of AI Web Service

Service Category	Method	Endpoint	Description	Required Parameters
Service	POST	/building-detection	Object Detection based on Rotated Mask R-CNN, YOLO	image: analysis target
		/change-detection	Land Cover Classification based on DeepLab V3	image1: before change image2: after change
		/land-cover-classification	Land Cover Change Detection based on DeepLab V3	image: analysis target
Data	GET	/datas	Retrieving a list of available data	-
		/datas/{seq}	Viewing detailed content of specific data	seq
		/datas/{dataseq}/files/{seq}	Downloading data files	seq dataseq
	POST	/datas	Adding new data entries	-
	PUT	/datas/{seq}	Updating existing data entries	seq
	DELETE	/datas/{seq}	Removing data entries	seq
		/datas/{dataseq}/files/{seq}	Removing data files	seq dataseq
Trend	GET	/trends	Retrieving a list of current trends	
		/trends/{seq}	Viewing detailed content of specific trends	seq
		/trends/{trendseq}/files/{seq}	Downloading thumbnail images for trends	seq trendseq:
	POST	/trends	Adding new trend entries	-
	PUT	/trends/{seq}	Updating existing trend entries	seq
	DELETE	/trends/{seq}	Removing trend entries	seq
		/trends/{trendseq}/files/{seq}	Removing thumbnail images for trends	seq trendseq

Note: In the table below, ‘seq’ denotes sequence, ‘dataseq’ denotes data sequence, and ‘trendseq’ denotes trend sequence.



<Figure 7> AI-based Land Monitoring Service Implementation Architecture

Service 카테고리에서는 딥러닝 추론 기능의 복잡성을 고려하여 Rotated Mask R-CNN과 YOLO 기반의 객체탐지, DeepLab V3 기반의 토지피복분류 및 변화탐지 기능은 개별 엔드포인트로 분리했다.

4.2 AI 기반 서비스 구현 방법

AI 기반 서비스는 <Figure 7>과 같이 구성된다. 객체 탐지는 사용자가 제공하는 단일 영상을 분석하여 특정 객체의 위치와 수를 파악하는 과정으로 시스템은 영상 내 객체의 바운딩 박스와 함께 탐지된 객체의 수를 사용자에게 반환하고 시각화하여 제공한다. 또한 토지피복 분석은 지정된 영역의 토지 사용 형태를 파악하는 분석하는 것으로 단일 영상에 대한 세그멘테이션된 영상과 각 클래스의 픽셀 수를 반환한다. 토지피복 변화탐지는 시간에 따른 토지 사용의 변화를 감지하는 과정으로 변화 전후 영상을 담은 요청을 보내고 딥러닝 모델을 이용하여 각 영상의 토지피복 세그멘테이션을 수행하고 각 픽셀에 대해 클래스가 변화한 부분을 찾아 제공한다. 각 모델은 독립적인 환경에서 특정 라이브러리와 버전 요구사항을 충족한 개발환경으로 구성되어 있다.

5. 결 론

본 연구에서 개발된 웹서비스는 LX공사의 업무망 내에서 운영된다. 이 웹서비스는 파이토치, 텐서플로우, 케라스 등의 오픈소스 머신러닝, 딥러닝, 신경망 라이브러리를 활용한다. 그러므로 정보보안을 위한 방화벽 등을 고려한 내외부망 연계가 필요하다.

웹서비스는 업무망 내에서 직원들이 자유롭게 접속할 수 있도록 상시 개방되며, 딥러닝 서버는 외부의 오픈소스 라이브러리 제공 웹사이트와 연계될 수 있도록 중간지대에 배치된다. 다만 사전에 인증된 사용자만이 이를 이용할 수 있다. 웹서비스에서 딥러닝 프로세스를 호출할 경우, 결과는 웹서비스 서버를 통해 제공된다.

웹서비스에서 제공하는 AI 모델은 주로 건물, 비닐하우스 등의 개발제한구역 내 불법가설물 탐지와 도로, 하천, 농경지 등 토지피복분류에 사용된다. AI 모델을 활용하면 대량의 데이터를 신속하게 분석하고 처리할 수 있지만, 학습의 정도에 따라 성능 차이가 발생한다. 따라서 지속적인 학습과 데이터의 종류나 형식 변화에 따른 모델 수정이 필요하다. 웹서비스는 사용자의 요구에 따라 분석 기능을 다양화하고, 학습자료를 추가하여 AI 모델을 고도화해야 한다.

본 연구에서는 LX공사의 연구자와 공간정보 업무 담당자를 대상으로 국토모니터링에 활용할 수 있는 딥러닝 모델을 웹서비스로 개발하였다. 현재는 항공사진을 활용하여 객체 추출과 토지피복분류가 가능한 두 가지 모델을 구현하였다. 앞으로는 고해상도 드론영상 활용 모델과 객체 추출 대상의 확대를 계획하고 있다. LX공사의 Land-XI 시스템과 연계하여 국토모니터링의 AI 활용 능력을 강화하는 것이 목표이다.

마지막으로, 현재 구현된 웹서비스는 관리자 측면에서 AI 모델을 학습하고 추론하는 기능을 제공하는 것에 주안점을 두고 있다. 그러나 향후 기술 발전과 사용자의 요구

사항 변화에 따라 더욱 진화된 서비스 플랫폼이 요구된다. 이러한 맥락에서 No-Code 기반의 AI 학습, 추론, 활용 기능을 제공하는 사용자 맞춤형 서비스 플랫폼의 구축이 필요하며, 이는 사용자들이 별도의 코딩 없이도 AI를 손쉽게 활용할 수 있게 만드는 방향성을 제시하는 추가 연구가 필요할 것이다. 본 연구는 그에 대한 시작점으로 국토모니터링의 정밀화와 AI 기술 발전을 고려할 때, 지속적인 연구와 기술개발이 이어지는 연구가 필요하며 이를 통해 국토의 변화를 신속하고 정확하게 모니터링 할 수 있는 웹서비스가 완성될 것으로 기대한다.

References

[1] AI hub, https://www.aihub.or.kr/.

[2] Chen, L.C., Papandreou, G., Schroff, F., and Adam, H., Rethinking Atrous Convolution for Semantic Image Segmentation, arXiv Pre-print, 2017, arXiv:1706.05587

[3] Ha, J., Park, K., and Kim, M., Development of Road Crack Detection System Using Deep Learning-based Segmentation and Object Detection, *Journal of Society for e-Business Studies*, 2021, Vol. 26, No. 1, pp. 93-106.

[4] Jeon, J.B., Yoo, E.J., Jeong, G.H., and Kim, B.E., A Study on the Implementation Plan of National Land using Survey for LX Future Business Model Discovery, LX Spatial Information Research Institute, 2020

[5] Jo, W. and Park, K. H., Deep learning based Land Cover Change Detection Using U-Net, *Journal of the Korean Geographical Society*, 2022, Vol. 57, No. 3, pp. 297-306

[6] Jo, W. and Park, K.H., Deep learning based Land Cover Change Detection Using U-Net, *Journal of the Korean Geographical Society*, 2022, Vol. 57, No. 3, pp. 297-306.

[7] Kornienko, D. V., Mishina, S. V., Shcherbatykh, S. V., and Melnikov, M. O., Principles of Securing RESTful API Web Services Developed with Python Frameworks, *Journal of Physics: Conference Series*, Vol. 2094, No. 3, IOP Publishing, 2021, p. 032016.

[8] Lee, J.H., Ji, Y.S., and Kang, J.H., A Study on Monitoring for Restricted Development Zone, LX Spatial Information Research Institute, 2020

[9] Mo, J.S., Seong, S.K., and Choi, J.W., Comparative evaluation of deep learning-based building extraction techniques using aerial images, *Journal of the Korean Society of Surveying*, 2021, Vol. 39, No. 3, pp. 157-165.

[10] Pack, H.J., A Study on Implementation of Web-based Machine Learning Platform and Applications[dissertation], [Deajeon, Korea]: Pai Chai University, 2020.

[11] Park, S., Park, Y., Jang, S., and Kim, T. H., A Study on Evaluating the Possibility of Monitoring Ships of CAS500-1 Images Based on YOLO Algorithm: A Case Study of a Busan New Port and an Oakland Port in California, *Korean Journal of Remote Sensing*, 2022, Vol. 38, No. 6-1, pp. 1463~1478

[12] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A., You Only Look Once: Unified, Real-Time Object Detection, arXiv Pre-print, 2015, arXiv:1506.02640

[13] Wang, P., Hu, W., Zhang, J., Wen, Y., Xu, C., and Qian, D., Enhanced Rotated Mask R-CNN for Chromosome Segmentation, *43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2021, Mexico, pp. 2769-2772

[14] Yoo, J.S. and Lee, M.H., Design of Enterprise System Platform based on Spring Data JPA in Spring Framework Environment, *Journal of Convergence for Information Technology. Convergence Society for SMB*, 2019, Vol. 9, No. 12, pp. 39-46.

[15] Zhang, F., Sun, G., Zheng, B., and Dong, L., Design and Implementation of Energy Management System Based on Spring Boot Framework, *Information*, 2021, Vol. 12, No. 11, p. 457.

ORCID

In-Hak Kong | http://orcid.org/0009-0002-3877-5541
Dong-Hoon Jeong | http://orcid.org/0009-0009-6960-1692
Gu-Ha Jeong | http://orcid.org/0009-0000-2368-3092