

철도 궤도의 이상상황 예방을 위한 영상처리와 딥러닝을 융합한 지능형 철도 레일 탐지 알고리즘[☆]

Intelligent Railway Detection Algorithm Fusing Image Processing and Deep Learning for the Prevent of Unusual Events

정 주 호¹ 김 다 현¹ 김 철 수² 오 염 덕¹ 안 준 호^{1*}
Ju-ho Jung Da-hyeon Kim Chul-su Kim Ryum-duck Oh Jun-ho Ahn

요 약

고속철도의 출현과 함께 철도는 국내외에서 자주 사용하는 교통수단 중 하나이다. 또한, 환경적인 측면에서도 다른 교통수단에 비해 이산화탄소 배출량도 적은 편이며 에너지 효율성은 높다. 철도에 관한 관심이 높아질 수록 철도의 안전과 관련된 문제는 중요한 관심사 중 하나이다. 그 중 시각적 이상현상은 철도 앞에 동물이나 사람 등 다양한 장애물이 갑자기 나타나 사고가 발생한다. 이러한 사고들을 예방하기 위해 철도 레일을 탐지하는 것은 기본적으로 탐지해야 하는 영역 중 하나이다. 철도에 설치된 카메라를 통해 영상을 수집할 수 있으며 철도 레일 탐지 방법은 전통적인 방식과 딥러닝 알고리즘을 이용한 방식이 있다. 전통적인 방식은 레일 주변의 다양한 노이즈로 인해 정확한 탐지가 어려우며 딥러닝 알고리즘을 이용하면 정확도 높게 탐지할 수 있으며 두 알고리즘을 융합하여 정확한 철도 레일을 탐지한다. 제안하는 알고리즘은 수집한 데이터를 기반으로 철도 레일 탐지에 대한 정확도를 판단한다.

☞ 주제어 : 고속철도, 철도 레일, 전통적인 방식, 딥러닝, 융합

ABSTRACT

With the advent of high-speed railways, railways are one of the most frequently used means of transportation at home and abroad. In addition, in terms of environment, carbon dioxide emissions are lower and energy efficiency is higher than other transportation. As the interest in railways increases, the issue related to railway safety is one of the important concerns. Among them, visual abnormalities occur when various obstacles such as animals and people suddenly appear in front of the railroad. To prevent these accidents, detecting rail tracks is one of the areas that must basically be detected. Images can be collected through cameras installed on railways, and the method of detecting railway rails has a traditional method and a method using deep learning algorithm. The traditional method is difficult to detect accurately due to the various noise around the rail, and using the deep learning algorithm, it can detect accurately, and it combines the two algorithms to detect the exact rail. The proposed algorithm determines the accuracy of railway rail detection based on the data collected.

☞ keyword : Rapid-transit railway, railway, deep learning, fusion

1. 서 론

고속철도의 출현과 함께 철도는 국내외에서 자주 사용하는 교통수단 중 하나이다. 그 이유는 다른 교통수단과 비교했을 때 안전성과 정시성이 우수하기 때문이다. 일반적인 여객열차는 한 량에 좌석이 70석 규모이며 6~8량 또는 10~20량 단위로 수송되기 때문에 대량 수송이 가능하여 단위 수송 비용이 낮다는 장점[1]이 있다. 또한, 환경적인 측면에서 교통수단에 따른 이산화탄소 배출량이 주요 관심사이다. 철도[2]는 승용차의 12%, 화물차의 8% 정도의 이산화탄소를 배출하지만, 에너지 효율성은 승용차의 8배, 화물차의 14배나 높기 때문에 철도에 대한

1 Dept. of Software, Korea National University of Transportation, Chungju, 27468, Korea

2 Dept. of Railway Vehicle System Engineering, Korea National University of Transportation, Uiwang, 16106, Korea

* Corresponding author (jhahn@ut.ac.kr)

[Received 30 June 2020, Reviewed 13 July 2020, Accepted 4 August 2020]

[☆] This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded(20QPWO-C158106-01) by the Ministry of Land, Infrastructure and Transport of Korean government.

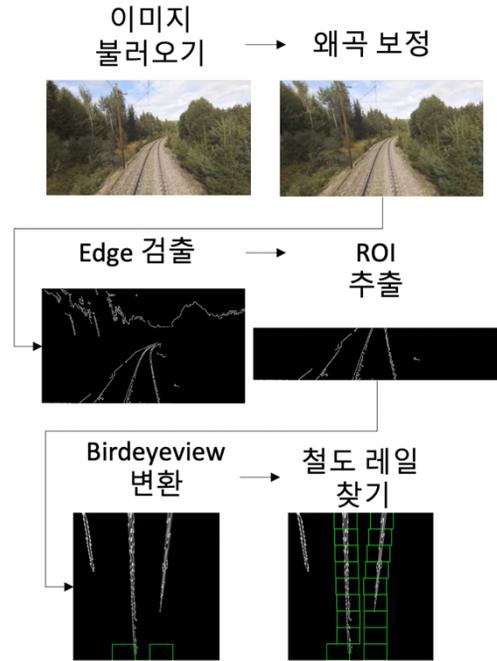
관심이 지속되고 있다. 철도에 관한 관심이 높아질수록 철도의 안전과 관련된 문제는 교통 분야에서 우려되는 문제 중 하나이다. 현재도 철도와 관련된 사고[3]는 빈번하게 발생하며 철도의 결함, 철도의 시각적 이상, 운전자의 잘못된 판단 등 다양한 원인에 의해 발생된다. 철도의 시각적 이상 현상은 철도 앞에 동물이나 사람 등 장애물이 갑자기 나타나 발생하는 사고 등이 있다. 이러한 사고들을 예방하기 위한 운전 보조 시스템을 위해 철도 교통 시스템의 안전에 관한 관심은 높아지고 있으며 시각적 이상 현상을 탐지하기 위해 기반되는 기술은 철도 레일을 인식하고 탐지하는 기술이다.

Vision 알고리즘들을 기반으로 철도 레일을 탐지하는 다양한 연구들이 진행되고 있으며 전통적인 방식으로는 원본 이미지에 다양한 필터를 적용하여 철도 레일을 탐지하는 알고리즘들이 있다. 원본 이미지를 Gray Color Space로 변환한 뒤 Canny edge를 통해 철도 객체를 탐지하고 Closing process를 통해 철도의 결함 등을 예측하는 연구[4]도 있으며 Sobel edge를 통해 처리한 값을 Canny edge를 이용하여 처리한 뒤 Laplacian filtering 연산을 하는 연구[5]도 있다. 필터를 적용한 이미지를 분류기를 통해 철도 레일을 탐지하는 다양한 연구들이 진행되고 있으며 Edge 검출 알고리즘을 통해 노이즈를 제거하고 이미지의 형태적 특징을 추출하여 허프 변환 및 계층적 Support Vector Machine(SVM) 분류기를 사용해 레일을 검출하는 연구[6]도 있다. Superpixel를 사용해 SVM분류기의 Term Frequency - Inverse Document Frequency (TF-IDF) 가 적용되는 기능을 기반으로 이미지를 학습하여 철도 레일을 분류하는 연구[7]도 있다. 또한, Resize시킨 이미지의 색상 대비를 줄이기 위한 필터를 적용하고 이미지의 고르지 않은 조명을 보정하기 위해 이미지에서 이미지 오프닝을 빼준 값을 Self-Organizing Map(SOM) 네트워크를 사용하여 복수 개의 클러스터로 구분한 뒤 그룹화하여 철도 레일을 분류하는 연구[8]도 있다. 동적 Region of Interest(ROI) 를 생성하고 이전 프레임의 결과와 이후 프레임에서 다항식 계수의 변화량의 제한한 값을 Multi-Layer Perceptron(MLP) 네트워크를 사용하여 회귀에 따른 다항식 최소 제곱 근사치를 통해 철도 레일을 예측하는 연구[9]도 있으며 철도 레일을 Residual net을 이용하여 탐지하는 연구[10]도 있다. 철도 레일의 연속성을 이용하여 레일의 각도를 통해 철도 레일을 탐지하는 연구[11]도 있으며 각도 정렬 측정 방법을 사용했으며 Edge 추출 과정 없이 색상과 및 변환에 강한 추출 방법이다.

본 연구에서는 Canny edge[4, 5]를 사용한 전통적인 방식을 통해 철도 레일을 탐지한다. 하지만, 철도 레일은 다양한 환경에 설치되어 있으며 Edge는 주변 노이즈에 취약하기 때문에 정확한 철도 레일의 위치를 탐지하기 어렵다. 이를 극복하고 정확한 철도 레일을 탐지하기 위해 딥러닝 알고리즘[12, 13, 14, 15] 사용하여 철도 레일을 탐지한다. 하지만, 본 연구에서 사용하는 딥러닝 알고리즘은 Bounding box를 통해 물체를 탐지하기 때문에 철도 레일의 유무만 판단할 수 있다. 그래서, 두 알고리즘을 융합한 융합 알고리즘을 이용하여 기존의 전통적인 방식보다 높은 정확도로 철도 레일을 탐지한다. 철도 레일을 탐지하기 위한 전통적인 방식, 딥러닝 알고리즘과 융합 알고리즘에 대해 설명하고 각 알고리즘의 철도 레일 탐지 결과 및 평가를 통해 성능을 확인하고자 한다.

2. 알고리즘

2.1 전통적인 방식



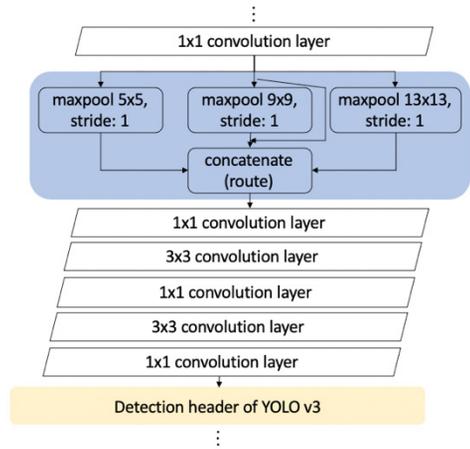
(그림 1) 전통적인 방식의 순서도
(Figure 1) Flowchart for the traditional method

철도 레일을 탐지하기 위한 전통적인 방식의 Image processing 방법은 이미지에서 Features를 추출하고 추출

된 Features를 기반으로 철도 레일을 탐지한다. 전통적인 방식의 전체적인 과정은 그림 1과 같다. 원본 이미지를 입력받고 이미지의 왜곡을 보정하며 이미지의 Features를 추출하기 위해 Canny edge detector를 사용한다. 이미지에서 Edge를 추출하기 위한 다양한 알고리즘이 존재하며 1차 미분 방식과 2차 미분 방식의 기법들이 사용된다. 대표적으로, 1차 미분 방식 중 Sobel이 많이 사용되며 모든 방향의 Edge를 추출할 수 있다는 장점이 있다. 2차 미분 연산 방식은 Canny가 많이 사용되며 노이즈로 인해 잘못된 Edge를 계산하는 것을 방지하기 위해 개발되었고 Canny edge detector가 이미지의 Edge를 잘 탐지하기 때문에 이 기법을 사용했다. 그 다음, 배경 등 다른 노이즈를 줄이기 위하여 ROI 영역을 지정했고 이미지의 y축을 기준으로 영역을 지정하여 레일 윗부분의 배경 및 물체들을 고려하지 않는다. ROI 영역이 지정된 이미지는 이미지의 원근이 존재하기 때문에 철도 레일을 정상적으로 탐지하기 어렵다. 이러한 이미지의 원근을 왜곡시키기 위해 Opencv의 Perspective transform 함수를 이용하여 Brid eye view 형태인 이미지를 위에서 바라보는 듯한 형태로 변환한다. 변환된 이미지를 기반으로 철도 레일을 탐지하기 위해 Sliding window search 기법을 이용하며 Window 안에 탐지되는 Edge 픽셀의 갯수가 지정한 Threshold 이상일 경우 철도 레일이 존재하는 영역이라고 판단한다. 탐지된 철도 레일의 영역들은 레일 Fitting 및 Smoothing 과정을 거쳐 각각 왼쪽과 오른쪽 레일을 추출하고 정상적으로 탐지했는지 확인하기 위해 원본 이미지에 탐지한 철도 레일에 색을 칠하여 시각적으로 확인하고자 했다.

2.2 Deep Learning 알고리즘

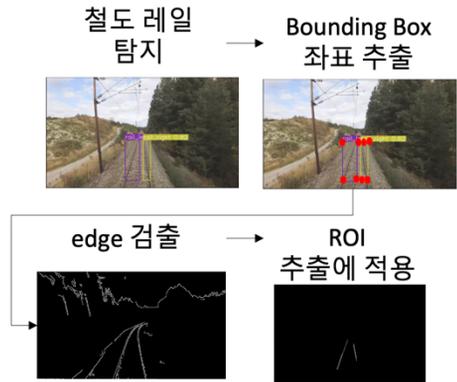
철도 레일을 탐지하기 위한 전통적인 방식은 철도 레일 주변의 다양한 노이즈 때문에 탐지에 많은 영향을 받는다. 본 연구에서는 다양한 환경에서 높은 정확도로 철도 레일을 탐지하기 위해 딥러닝 알고리즘을 사용하며 연구에서 사용한 네트워크 구조는 그림 2와 같다. 1 Stage object detection 알고리즘 중 YOLO v3 SPP 알고리즘을 사용하여 빠른 수행 속도와 준수한 정확도를 유지할 수 있다. 수행 속도는 느리지만 높은 정확도로 탐지할 수 있는 2 Stage object detection 알고리즘들도 존재하지만, 본 연구에서는 추후에 제한된 컴퓨팅 자원에서도 활용될 수 있도록 1 Stage object detection 알고리즘을 사용했다. 또한, YOLO v3 알고리즘은 기본적인 네트워크 구



(그림 2) YOLO v3 SPP의 구조
(Figure 2) Architecture of YOLO v3 SPP

조뿐만 아니라 Tiny, Slim 등 다양한 모듈을 추가하여 성능 및 실행 속도를 개선시킨 구조[15]들이 많이 존재한다. 본 연구에서는 정확도를 조금 더 높인 SPP 모듈을 추가한 YOLO v3 SPP 알고리즘을 사용했다.

2.3 융합 알고리즘



(그림 3) 융합 알고리즘의 구조
(Figure 3) Architecture of Fusion Algorithm

본 연구에서는 철도 레일을 탐지하기 위해 전통적인 방식과 딥러닝 알고리즘을 융합한 융합 알고리즘을 제안하며 융합 알고리즘의 구조는 그림 3과 같다. 전통적인 방식을 이용하여 Edge를 탐지한 후, ROI 영역을 추출할 때 이미지의 Height를 기준으로 철도 레일 위의 배경을

제거했다. 하지만, 융합 알고리즘에서는 철도 레일 주변의 노이즈를 제거하기 위하여 YOLO v3 SPP 알고리즘을 통해 탐지한 왼쪽과 오른쪽 철도 레일의 영역을 제외한 나머지 edge를 모두 제거하여 철도 레일의 탐지 정확도를 높이고자 했다. 우선, Transfer learning 기법[16]으로 학습된 YOLO v3 SPP 알고리즘 모델을 이용하여 왼쪽과 오른쪽 철도 레일을 탐지한다. 정상적으로 탐지 되면 Bounding box의 Xmin, Ymin, Xmax, Ymax 의 좌표를 추출할 수 있다. 4개의 좌표를 이용하여 Edge를 적용한 이미지에 왼쪽과 오른쪽 철도 레일의 영역을 제외한 나머지 Edge를 모두 제거한다. 이를 적용한 이미지를 이용하여 Brid eye view 형태로 이미지를 변환 후 Sliding window search 기법을 사용하여 철도 레일을 탐지하게 된다. 이는 전통적인 방식보다 철도 레일 주변의 노이즈를 제거하기 때문에 효율적으로 탐지할 수 있다.

3. 실험 및 결과

본 연구에서는 철도 레일을 탐지하기 위하여 유튜브를 통해 영상 데이터를 수집하여 분석했다. 모든 실험에 사용된 데이터는 좋은 날씨 환경에서 기차에 카메라를 설치하여 수집된 영상에서 추출했고 검증 데이터 325장의 이미지를 기반으로 각 알고리즘의 성능을 비교했다. 전통적인 방식을 통해 철도 레일을 탐지한 실험의 결과는 표 1과 같다.

(표 1) 전통적인 방식의 결과

(Table 1) A result of traditional method

Algorithm	F-measure	Accuracy
Traditional method	0.90	0.82

전통적인 방식의 정확도는 0.82이며 철도 레일을 정상적으로 탐지한 이미지는 268장이며 철도 레일 영역을 벗어나 탐지한 이미지는 57장이다. 전통적인 방식을 통해 Canny edge detector를 이용하여 Edge detection한 결과는 그림 4와 같으며 Canny edge를 적용하기 전에 Gaussian blur를 적용했다. Gaussian blur를 적용한 이유는 blur 기법을 통해 1차적으로 노이즈를 제거하기 위함이다. Edge가 적용된 이미지에 ROI 영역을 추출하고 Bird eye view 형태로 이미지를 변환한 후에 Sliding window search 기법을 이용하여 철도 레일을 탐지하여 레일 탐지 영역을 원본 이미지에 적용한 결과는 그림 5와 같다. 전통적인 방식은 Edge를 기반으로 철도 레일을 탐지하기



(그림 4) canny edge detector의 결과

(Figure 4) A detection result of canny edge detector



(그림 5) 전통적인 방식을 이용한 철도 레일 탐지 결과

(Figure 5) A result of railway detection using traditional method

때문에 철도 레일 주변의 다양한 노이즈로 인해 철도 레일을 정상적으로 탐지하지 못하는 경우가 발생된다.

딥러닝 알고리즘인 YOLO v3 SPP 알고리즘을 이용하여 철도 레일을 탐지한 실험의 결과는 표 2와 같다.

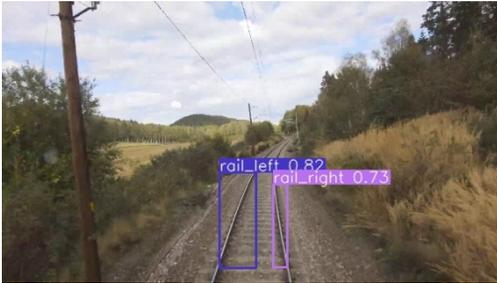
(표 2) YOLO v3 SPP의 결과

(Table 2) A result of YOLO v3 SPP

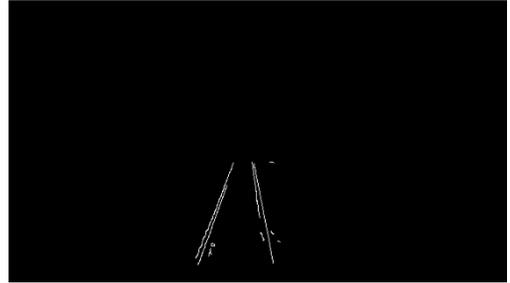
Algorithm	F-measure	Accuracy
Yolo V3 SPP	1	0.99

표 2를 통해 확인한 결과 왼쪽과 오른쪽 철도 레일에 대한 탐지는 높은 정확도로 탐지해내고 있음을 확인할 수 있었다. 화창한 날씨 환경에서 YOLO v3 SPP 알고리즘을 이용하여 Transfer learning을 통해 왼쪽과 오른쪽 철도 레일을 탐지하는 모델을 학습했고 학습에 사용된 가중치는 Darknet53.conv.74 파일을 사용했고 Epoch은 1000번을 학습했으며 네트워크의 Input size는 608 x 608을 사용했다. YOLO v3 SPP 알고리즘을 이용하여 탐지한 결과는 그림 6과 같다. 대부분의 날씨가 좋은 환경에서 왼쪽과 오른쪽 철도 레일을 높은 정확도로 탐지한 것을 확인할 수 있다. 하지만, 그림 7 처럼 정상적으로 탐지하지 못하는 경우도 발생했다. 한 쪽 레일을 아예 탐지하지 못하거나 동일한 레일을 중복하여 탐지하는 경우도 발생했다.

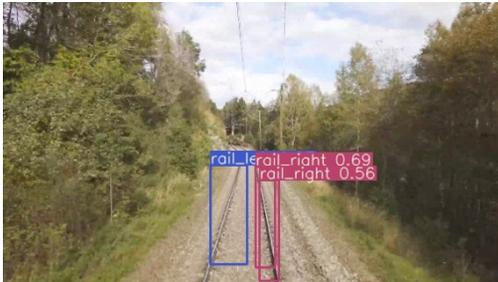
융합 알고리즘을 이용하여 철도 레일을 탐지한 결과는 표 3과 같다. 철도 레일을 정상적으로 탐지한 이미지



(그림 6) YOLO v3 SPP를 이용한 철도 레일 탐지 결과
(Figure 6) A result of railway detection using YOLO v3 SPP



(그림 8) 융합 알고리즘을 이용한 canny edge detector 결과
(Figure 8) A result of canny edge detector using fusion algorithm



(그림 7) YOLO v3 SPP를 이용한 철도 레일 오탐지 결과
(Figure 7) A false result of railway detection using YOLO v3 SPP



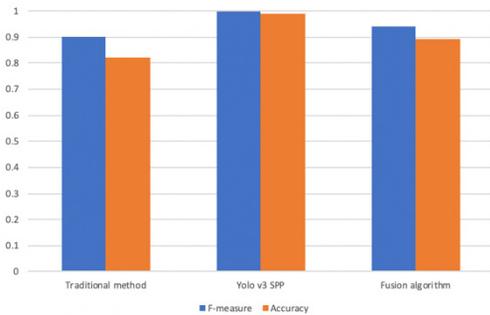
(그림 9) 융합 알고리즘을 이용한 철도 레일 탐지 결과
(Figure 9) A result of railway detection using fusion algorithm

(표 3) 융합 알고리즘의 결과
(Table 3) A result of fusion algorithm

Algorithm	F-measure	Accuracy
Fusion algorithm	0.94	0.89

는 288장, 오 탐지한 이미지는 37장이다. 전통적인 방식만 이용해서 철도 레일을 탐지하는 것보다 융합 알고리즘을 통해 철도 레일 주변의 노이즈를 제거하는 것이 정확도의 개선이 있었음을 알 수 있었다. 융합 알고리즘을 이용한 Canny edge의 결과는 그림 8과 같다. 기존에 Canny edge를 적용하고 이미지의 Height를 기준으로 ROI

영역을 추출한 그림 4에 비해, YOLO v3 SPP 알고리즘을 통해 왼쪽과 오른쪽 철도 레일 영역을 탐지하고 나머지 영역에 존재하는 Edge를 모두 제거하여 철도 레일을 탐지하는데 있어서 발생될 수 있는 노이즈를 최대한 줄이는 것을 목표로 했다. 노이즈가 제거된 Edge를 기반으로 철도 레일을 탐지한 결과는 그림 9와 같다. 철도 레일을 효과적으로 탐지하는 것은 철도의 시각적 이상 현상을 예방하기 위해 기본적으로 탐지해야 하는 영역이다. 전통적인 방식을 통해 edge를 추출 할 경우, 철도 레일 주변의 다양한 노이즈까지 탐지될 수 있기 때문에 이를 효과적으로 제거할 수 있도록, 딥러닝 알고리즘인 YOLO v3 SPP 알고리즘을 Transfer learning 기법을 통해 왼쪽과 오른쪽 차선을 탐지하는 모델을 학습했다. 학습한 모델이 왼쪽과 오른쪽 철도 레일을 탐지하게 되면 Edge를 추출한 후에 철도 레일 영역을 제외한 나머지 Edge를 모두 제거하여 철도 레일을 효과적으로 탐지할 수 있도록 하는 융합 알고리즘을 통해 전통적인 방식보다 높은 정확도로 철도 레일을 탐지할 수 있으며 그림 10을 통해 각 알고리즘의 성능을 비교할 수 있다.



(그림 10) 전통적인 방식, 딥러닝 알고리즘, 융합 알고리즘 결과 비교

(Figure 10) Compare the results of traditional method, deep learning algorithm and fusion algorithm

4. 결 론

본 연구에서는 철도의 시각적 이상현상을 예방하기 위해 가장 기본이 되는 철도 레일을 탐지했다. 철도 레일을 탐지하기 위하여 전통적인 방식과 딥러닝 알고리즘을 사용하여 탐지했다. 전통적인 방식은 철도 레일 주변에서 발생될 수 있는 노이즈에 취약하며 정상적으로 철도 레일을 탐지하기 어려웠다. 딥러닝 알고리즘은 Object detection을 수행하기 때문에 철도 레일의 유무만 판단할 수 있다. 우리는 이를 효과적으로 융합한 융합 알고리즘을 제안했다. 융합 알고리즘은 딥러닝 알고리즘을 통해 탐지한 철도 레일의 좌표 정보를 기반으로 전통적인 방식에서 탐지한 불필요한 Edge를 모두 제거하여 철도 레일을 탐지할 수 있도록 했다. 이를 통해, 높은 정확도로 철도 레일을 탐지할 수 있었다. 추후에는 철도 레일 뿐만 아니라 다양한 시각적 이상현상들을 탐지할 수 있는 알고리즘과 다양한 센서들을 융합하여 철도의 다양한 분야에서 활용될 수 있도록 할 예정이다.

참고문헌(Reference)

[1] Statistics Korea, “Statistical Information Report for Users of Korea Railroad Statistics”, 2017.
http://kostat.go.kr/portal/korea/kor_pi/8/6/2/index.board?bmode=read&aSeq=365757

[2] Y. Lee, D. Moon, J. Yoo, “A Comparative Study on Policy of Modal Shift for Enhancing of Eco-friendly Rail Freight Transportation”, The Korean Society for Railway Spring conference, pp. 2438-2441, 2008
<https://www.koreascience.or.kr/article/CFKO200821041235034.j>

[3] Railsafety, “Five-Year Railway Accident Analysis Report”, 2019.
<https://www.railsafety.or.kr/>

[4] M. Karakose, O. Yaman, M. Baygin, K. Murat, and E. Akin, “A New Computer Vision Based Method for Rail Track Detection and Fault Diagnosis in Railways”, International Journal of Mechanical Engineering and Robotics Research, Vol. 6 No. 1, pp. 2-3, 2017.
<https://doi.org/10.18178/ijmerr.6.1.22-27>

[5] E. Páli, K. Máthé, L. Tamás, L. Buşoniú, “Railway track following with the AR.Drone using vanishing point detection”, 2014 IEEE International Conference on Automation, Quality and Testing, Robotics, pp. 4-5, 2014.
<https://doi.org/10.1109/AQTR.2014.6857870>

[6] C. Tastimur, M. Karakose, E. Akin, “A Vision Based Condition Monitoring Approach for Rail Switch and Level Crossing using Hierarchical SVM in Railways”, International Journal of Applied Mathematics, Electronics and Computers, pp. 3-6, 2016.
<https://doi.org/10.18100/ijamec.270634>

[7] Z. Teng, F. Liu, B. Zhang, “Visual railway detection by superpixel based intracellular decisions”, Multimed Tools Appl Vol. 75, pp. 2473 - 2486, 2016.
<https://doi.org/10.1007/s11042-015-2654-x>

[8] S. Yella, R. G. Nyberg, B. Payvar, M. Dougherty, N. K. Gupta, “Machine Vision Approach for Automating Vegetation Detection on Railway Tracks”, Journal of Intelligent Systems, Vol. 22, No. 2, pp. 179-196, 2013.
<https://doi.org/10.1515/jisys-2013-0017>

[9] M. A. Selver, E. Y. Zoral, B. Belenlioglu, S. Dogan, “Predictive modeling for monocular vision based rail track extraction”, 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, pp. 1-6, 2017.
<https://doi.org/10.1109/CISP-BMEI.2017.8301928>

- [10] S. Mittal, D. Rao, "Vision Based Railway Track Monitoring Using Deep Learning", arXiv, pp. 1-6, 2017.
<https://arxiv.org/abs/1711.06423>
- [11] H. Wu, W. C. Siu, "Real time railway extraction by angle alignment measure", IEEE International Conference on Image Processing, pp. 1-5, 2015.
<https://doi.org/10.1109/ICIP.2015.7351670>
- [12] Joseph Redmon, Ali Farhadi, "YOLOv3: an Incremental Improvement", arXiv, pp. 1-6, 2018.
<https://arxiv.org/abs/1804.02767>
- [13] Zhanchao Huang, Jianlin Wang, "DC-SPP-YOLO: Dense Connection and Spatial Pyramid Pooling Based YOLO for Object Detection", arXiv, pp. 1-23, 2019.
<https://arxiv.org/abs/1903.08589>
- [14] K. He, X. Zhang, S. Ren, J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", European Conference on Computer Vision, Vol. 8691, pp. 346-361, 2014.
https://doi.org/10.1007/978-3-319-10578-9_23
- [15] P. Zhang, Y. Zhong, X. Li, "SlimeYOLOv3: Narrower, Faster and Better for Real-Time UAV", IEEE/CVF International Conference on Computer Vision Workshop, pp. 37-45, 2019.
<https://doi.org/10.1109/ICCVW.2019.00011>
- [16] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, C. Liu, "A Survey on Deep Transfer Learning. Machine Learning", International Conference on Artificial Neural Networks, pp. 270-279, 2018.
https://doi.org/10.1007/978-3-030-01424-7_27

● 저 자 소 개 ●



정 주 호(Ju-ho Jung)

2013년 03월~ 2019년 08월 한국교통대학교 소프트웨어학과(공학사)

2019년 09월~현재 한국교통대학교 일반대학원 소프트웨어학과

관심분야 : Deep learning, Vision, Pattern recognition, Model compression

E-mail : jjs1005k@ut.ac.kr



김 다 현(Da-hyeon Kim)

2018년 03월~ 현재 한국교통대학교 컴퓨터정보기술공학부 소프트웨어전공

관심분야 : 인공지능, 패턴 인식

E-mail : 1826059@ut.ac.kr

● 저 자 소개 ●



김 철 수(Chul-su Kim)

2002년 한양대학교 대학원 기계설계학과(공학박사)
2008년 3월~현재 한국철도공사, 서울교통공사 자문위원
2003년 3월~현재 : 한국교통대학교 철도공학부 교수
관심분야 : 머신러닝, 철도차량
E-mail : chalskim@ut.ac.kr



오 염 덕(Ryum-duck Oh)

1993년 홍익대학교 대학원 전자계산학과(이학박사)
1990년~현재 한국교통대학교 컴퓨터정보기술공학부 소프트웨어전공 교수
관심분야 : IoT, Database, Data Analysis
E-mail : rdoh@ut.ac.kr



안 준 호(Jun-ho Ahn)

2009년~2013년 University of Colorado, Boulder, Computer science, Ph.D.
2013년~2017년 ETRI 국가보안기술연구소
2017년~현재 한국교통대학교 컴퓨터정보기술공학부 소프트웨어전공 교수
관심분야 : 인공지능, 지능형IoT, 패턴 인식
E-mail : jhahn@ut.ac.kr