

자율주행 트랙터 환경에서 쓰러진 사람에 대한 데이터 증강

백화평¹, 안한세², 채희성³, 정용화⁴

¹고려대학교 컴퓨터융합소프트웨어학과 석사과정

²고려대학교 컴퓨터융합소프트웨어학과 박사과정

³(주)파인에스엔에스 CEO 대표이사

⁴고려대학교 컴퓨터융합소프트웨어학과 교수

qorrns156@korea.ac.kr, hansahn@korea.ac.kr, ceo@pinesns.co.kr,
ychungy@korea.ac.kr

Dataset Augmentation on Fallen Person Objects in a Autonomous Driving Tractor Environment

Hwapyeong Baek*, Hanse Ahn*, Heesung Chae*, Yongwha Chung*

*Dept. of Computer Convergence Software, Korea University

*Pinesns Co., Ltd.

요약

데이터 증강은 데이터 불균형 문제를 해결하기 위해 일반화 성능을 향상시킨다. 이는 과적합 문제를 해결하고 정확도를 높이는 데 도움을 준다. 과적합을 해결하기 위해서 본 논문에서는 분할 마스크 라벨링을 자동화하여 효율성을 높이고, RoI를 활용한 분할 Copy-Paste 데이터 증강 기법을 제안한다. 본 논문의 제안 방법을 적용한 결과 YOLOv8 모델에서 기존의 분할, 박스 Copy-Paste 데이터 증강 기법과 비교해서 쓰러진 사람 객체에 대한 정확도가 10.2% 증가함으로써 제안한 방법이 일반화 성능을 높이는 데 효과가 있음을 확인하였다.

1. 서론

최근 딥러닝의 성장과 함께 CNN이 발전되고 있다. 이로 인해 이미지 분류, 객체 탐지 및 이미지 분할과 같은 컴퓨터 비전 분야에서 새로운 기법들이 등장하고 있다. 현재 정확도와 속도 차원에서의 성능을 개선하기 위해 모델 프루닝, 앙상블, 어텐션, 네트워크 구조 변경 등 다양한 기법들이 활용된다. 이러한 기법들에서 해결하기 어려운 문제 중 하나가 일반화 성능을 향상시키는 것이다. 일반화 성능은 이전에 본 데이터와 이전에 본 적이 없는 데이터를 평가할 때의 성능 차이를 말한다. 일반화 성능이 낮으면 과적합 문제가 생긴다[1]. 과적합이란 특정 데이터셋에만 필요 이상으로 학습하는 것을 말한다. 이는 본 적 없는 테스트 데이터셋에서 정확도를 낮추게 하는 원인이 된다. 과적합 문제를 해결하기 위한 방법 중 하나는 데이터 증강이다. 데이터 증강이란 학습할 데이터를 변형하여 데이터를 늘리는 기법이다. 다른 기법과 달리 학습할 때 사용되기 때문에 추론 속도에 영향을 주지 않는다는 장점이 있다. 우리는 데이터 증강을 이용하여 자율주행 트랙터 환경에서 쓰러진 사람 객체의 탐지 정확도를 높이는

것에 초점을 맞춘다.

자율주행 트랙터 환경에서 가장 위험한 요소는 장애물보다 피할 수 없는 사람과 부딪히는 것이다. 이러한 문제를 해결하기 위해 NREC[2] 데이터셋이 제안되었다. 그러나 해당 데이터셋은 쓰러진 사람 객체에 대한 비율이 14.2%로 비교적 적다는 문제가 있으며, 이것은 데이터 불균형 문제와 일반화 성능을 낮아지게 하는 문제로 이어진다[3]. 이를 해결하기 위해 본 논문에서는 RoI(Region of Interest)를 이용한 분할 Copy-Paste 데이터 증강 기법을 제안한다. 본 논문에서는 쓰러진 사람의 공공 데이터셋[4]를 사용하여, 해당 이미지 안에 있는 객체와 배경 이미지를 결합한 합성 데이터를 생성하는 기법을 제안한다.

2. 관련 연구

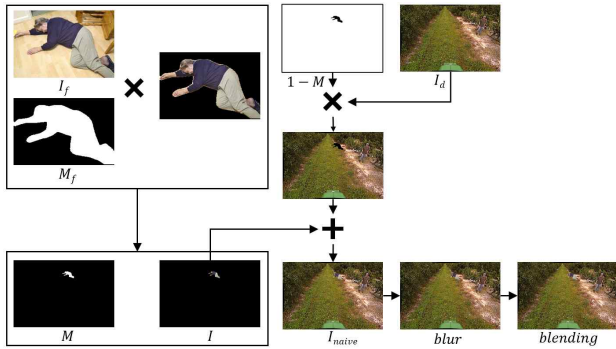
데이터 증강은 컴퓨터 비전 분야에서 부족한 데이터를 증가시켜 모델을 학습하는데 사용된다. 데이터 증강에는 Flipping, Rotating, Cropping, Color Jittering, Edge Enhancement, Fancy PCA, Copy-Paste[1,5] 등등의 다양한 데이터 증강 기법이 존재한다. 이러한 기법 중 Copy and Paste 데이터

증강 기법은 이미지 내에 객체를 임의의 크기와 임의의 좌표로 설정하여 다른 배경 이미지와 합성하는 기법이다. 이는 분할, 박스 등의 객체로 추출하여 사용할 수 있다. 최근에도 이미지 객체 분할 분야에서 Copy-Paste가 널리 사용되고 있다[5-6].

3. 제안 방법

본 논문에서는 트랙터 환경에서 쓰러진 사람 객체의 탐지 정확도를 높이기 위한 새로운 데이터 증강 기법을 제안한다. 기존의 Copy-Paste 기법을 기반으로 단점을 보완하여 발전시켰다. 기존의 방법에 대한 문제점은 RoI가 아닌 크기와 위치에 붙여넣게 되어 잘못된 위치와 잘못된 크기를 학습하게 된다. 이는 정확도를 낮추는 결과를 초래한다. 또한, 붙여넣을 분할 객체에 마스크 라벨링을 수작업으로 해야 한다는 번거로움이 존재한다. 이는 많은 데이터가 필요할 때, 모든 객체를 수작업으로 라벨링하기 어렵다는 문제점이 발생한다. 위 문제점들을 해결하기 위한 본 논문에 기여는 다음과 같다.

- YOLOv8[7]과 같은 분할 모델을 이용하여 자동으로 공공 데이터셋의 마스크 라벨링을 추출하여 합성 이미지를 생성하는 방법을 제안한다.
- 이전과 현재 이미지와의 픽셀값들의 차를 이용해 구간을 나누어 RoI(Region of Interest)의 크기와 좌표를 구하는 방법을 제안한다.



(그림 1) 쓰러진 사람의 이미지를 생성하기 위해 제안된 프레임워크.

3.1 객체 붙여넣기

그림 1과 같이 공공 데이터셋 I_f 에서 COCO로 사전 학습된 YOLOv8x-seg 모델을 이용하여 마스크 M_f 를 구한다. 마스크의 쓰러진 사람을 포함하는 영역은 1이고, 그 외의 영역은 0으로 설정한다. 본 논문 이미지인 I_d 에 다음과 같은 합성 식 (1)을 적용하여 합성 이미지인 I_{naive} 를 생성한다[5].

$$I_{naive} = M \times I + (1 - M) \times I_d \quad (1)$$

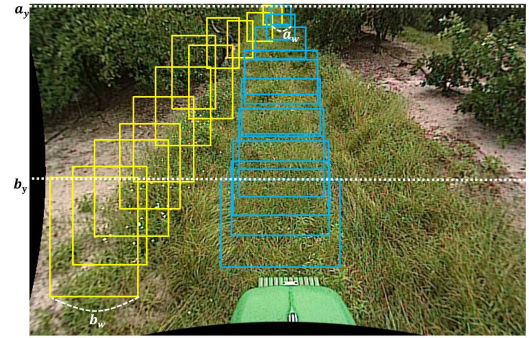
이미지 생성 도중 다른 객체와 겹치는 부분이 있으면 이미지 생성을 하지 않는다.

3.2 구간 나누기

본 논문에서 RoI를 구하기 위해 이미지들을 여러 구간으로 나누고 해당 구간 내에 박스 어노테이션을 이용하여 RoI에 해당하는 크기와 좌표를 구하는데 사용한다. 영상처리 기법을 사용하여 픽셀 값의 분포도가 이전 프레임과의 차이가 일정 이상 나면 다른 구간이라고 판단한다. 그 기준은 눈으로 보고 판단한다. 해당 구간 내에서 크기와 좌표를 구하기 위한 식은 (2)와 같다.

$$f_{y,w}(x) = \frac{b_{y,w} - a_{y,w}}{n}x + a_{y,w} \quad (2)$$

여기서 a 는 YOLO 박스 어노테이션 정보의 최솟값을 의미하고 b 는 최댓값을 의미한다. n 은 구간 내의 이미지의 수이다. $f_{y,w}(x)$ 는 구간 내의 x 번째 이미지에 대응하는 붙여넣는 객체의 y 좌표와 너비이다. 그림 2와 같이 구간 내에서 가장 큰 박스와 가장 작은 박스에 대응하는 크기와 좌표를 구한다. 또한, 트랙터가 멈출 때와 움직일 때를 이전 프레임과의 픽셀 차를 이용하여 쓰러진 사람의 좌표와 크기를 조정한다. 그리고 트랙터가 후진하지 않는다는 특징을 고려해 크기와 좌표가 작아지지 않게 설정한다.

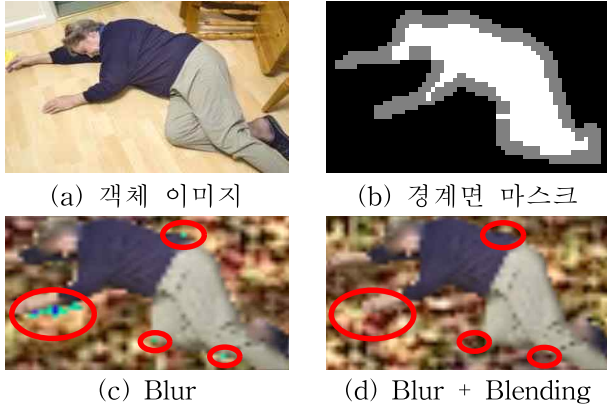


(그림 2) 구간 내에 박스 어노테이션의 최솟값 a , 최댓값 b .

3.3 영상 처리

YOLOv8 분할 모델을 이용하기 때문에 붙여넣을 객체의 가장자리가 해당 배경을 포함한 상태로 가져오는 경우가 발생한다. 이것은 노이즈의 원인이 된다. 학습할 때, 이를 완화하기 위해 마스크의 경계면에 Gaussian blur를 사용한다.

그러나 blur를 적용했을 때, M_f 이 1보다 작은 값으로 나오게 된다. 이것은 그림 3에 (c)와 같이 오히려 노이즈를 더 발생시킨다. 이는 학습할 때, 노이즈의 특징까지 학습하게 되므로 정확도를 더 낮추는 원인이 된다. 이것을 없애기 위해 우리는 alpha



(그림 3) 영상처리 기법 비교.

blending 기법을 사용한다. 여기서 alpha blending이란 M_f 와 M_f 에 대응하는 배경 이미지 B 에 대해서 M_f 가 1보다 작은 값이 존재할 경우, 그 비율만큼 객체에 가중치를 주어 합성하는 기법이다. 식은 (3)과 같다.

$$M_f = \alpha M_f + (1 - \alpha)B \quad (3)$$

알고리즘 영상처리를 이용한 객체 붙여넣기

입력: boundary_pixel = 경계면 픽셀 두께

h, w = 붙여넣을 객체의 높이와 너비

M_f = 붙여넣을 객체의 마스크

출력: I_{naive} = 객체를 붙여넣은 결과 이미지

check = 0, 현재 위치 상태를 확인하는 변수

blurred_img = Gaussian blur를 적용한 이미지

$B = M_f$ 에 대응하는 배경 이미지

for 0 to h, w **do**

if $M_f == 1$ **then**

if 이전의 $M_f == 0$ and check == 0 **then**

 boundary_pixel만큼 $M_f = 0.5$

 check=1 **end if**

if 다음의 $M_f == 0$ and check == 1 **then**

 boundary_pixel만큼 $M_f = 0.5$

 check = 0 **end if**

if $M_f == 0.5$ **then**

$M_f = \text{blurred_img}$ **end if**

end for

end for

$B = M_f \times \text{배경 이미지}$

$M_f = \text{Alpha_Blending}(M_f, B)$

M_f = 각 25% 확률로 90, 180, 270도 회전

$I_{naive} = M \times I + (1 - M) \times I_d$

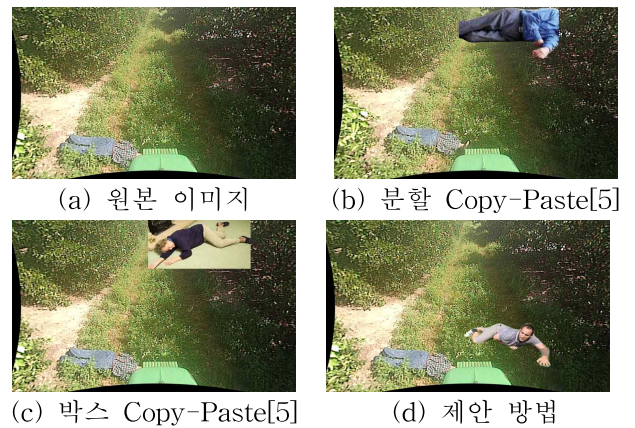
4. 실험

본 논문에서 표 1과 같이 NREC[2] 데이터셋을 사용하였으며, 학습, 검증, 테스트 세트를 6:2:2 비율로 총 49,404장의 데이터셋을 사용하였다. 해상도는 720x480이며, 학습과 검증 데이터셋은 각각 29,643장, 9,880장을 사용하였다. 테스트 데이터셋으로는 쓰러진 사람 데이터인 2,256장에서 평가하였다.

<표 1> NREC[2] 데이터셋 분류.

	학습	검증	테스트
본 적 없는 데이터	0	8,821	8,821
전체 데이터	29,643	9,880	9,881
쓰러진 사람 데이터	4,234	1,412	2,256

Fall Detection 공공 데이터셋[3]에서 트랙터 환경에서 쓰러진 사람 객체에 적합한 이미지를 눈으로 보고 판단하여 37장의 이미지를 선정하였다. 해당 데이터셋은 265x190부터 1300x989까지 다양한 크기의 해상도가 포함된다. 그림 4는 분할, 박스 Copy-Paste와 제안 방법에 대한 데이터 증강 결과 비교 사진이다.



(그림 4) Copy-Paste[5] 방법과 제안 방법 결과 비교 사진.

4.1 평가 지표

본 논문에서 AP(Average Precision) 평가 지표를 사용하였다. 예측된 박스와 실제 박스와 일치하는 비율을 IoU(Intersection over Union)이라고 한다. 예측된 박스와 실제 박스의 IoU가 50% 이상이면, TP(True Positive)라고 하고, 50% 미만이면 FP(False Positive)라고 한다. 그리고 실제 박스인

곳에 예측하지 않는 경우를 FN(False Negative)이라고 한다. Precision은 정밀도를 의미하며 식은 (4)와 같다.

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

Recall은 재현율을 의미하며 식은 (5)와 같다.

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

AP는 x축이 recall, y축이 precision인 PR 곡선의 면적으로, recall과 precision을 포함한 성능을 정량적으로 평가하여 대부분의 컴퓨터 비전 분야에서 사용된다.

4.2 실험 결과

우리는 COCO로 사전 학습된 YOLOv8[7] 모델을 사용하였고, 학습과 검증 데이터셋 이미지 크기는 704로 설정하였고, epoch은 15으로 설정하였다. 나머지는 기본 환경 설정을 따랐다.

표 2는 데이터 증강을 적용하지 않은 baseline과 기존의 분할, 박스 Copy-Paste 데이터 증강 기법과 제안 방법을 비교한 실험 결과이다. 제안 방법은 baseline과 비교했을 때, AP가 63.5%에서 71.7%로 8.2% 증가하였다. Precision과 Recall은 각각 16.2%, 2.6% 증가하였다. 제안 방법이 기존의 분할 Copy-Paste보다 AP 10.2% 증가하였다.

<표 2> 쓰러진 사람 데이터셋에 대해서
Copy-Paste[4] 기법과 정확도 비교.

Method	Precision ↑	Recall ↑	AP ↑
baseline	73.3	58.2	63.5
baseline + 분할 Copy-Paste	69.1	55.1	61.5
baseline + 박스 Copy-Paste	76.7	53.4	59.8
Ours	89.5	60.8	71.7

표 3은 제안한 RoI 방법과 Gaussian blur, alpha blending에 대한 비교 실험이다.

<표 3> 영상처리 방법에 대한 정확도 비교.

Method	Precision ↑	Recall ↑	AP ↑
RoI	74.4	56.3	66.5
RoI+blur	78.3	60.8	68.3
RoI+blur+blending	89.5	60.8	71.7

5. 결론

데이터 증강은 데이터 불균형 등으로 인한 과적

합 문제를 해결하기 위한 해결책 중 하나이다. 본 논문에서는 RoI를 이용한 분할 Copy-Paste 데이터 증강 기법을 제안하였다. 실제 트랙터 환경과 비슷하게 쓰러진 사람을 합성하였으며, AP가 분할 Copy-Paste 데이터 증강 기법보다 10.2% 증가하였다. 이는 제안한 방법이 쓰러진 데이터셋에 대해서 더 효과적임을 보였다.

감사의 글

본 과제는 2021년도 교육부의 재원으로 한국연구재단의 지원을 받아 수행된 지자체-대학 협력기반 지역혁신 사업(2021RIS-004)의 결과임.

참고문헌

- [1] Connor Shorten, et al., "A Survey on Image Data Augmentation for Deep Learning," Journal of Big Data, 6, 1, pp. 1-48, 2019.
- [2] Zachary Pezzementi, et al., "Comparing Apples and Oranges: Off-Road Pedestrian Detection on the NREC Agricultural Person-Detection Dataset," Journal of Field Robotics, 35, 4, pp. 545-563, 2018.
- [3] Pablo Ruiz-Ponce, et al., "POSEIDON: A Data Augmentation Tool for Small Object Detection Datasets in Maritime Environments," Sensors, 23, 7, pp. 3691, 2023.
- [4] Fall Detection Dataset. Available online: <https://www.kaggle.com/datasets/uttejmarkandagatla/fall-detection-dataset> (accessed on 10 June 2023).
- [5] Golnaz Ghiasi, et al., "Simple Copy-Paste Is A Strong Data Augmentation Method for Instance Segmentation," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, 2021, pp. 2918-2928.
- [6] Suhyeon Lee, et al., "Fallen Person Detection for autonomous driving," Expert Systems with Applications, 213, pp. 119243, 2023.
- [7] Ultralytics/ultralytics Available online: <https://github.com/ultralytics/ultralytics> (accessed on 2 June 2023).