



**R e p o r t**

**Deep-Cycle**

**작성일 2019-08-27**

**팀 고려대-4조**

**이름 심재준**

박정운

박채림

박현종

황준식

정효희

|  |  |
| --- | --- |
| 1. 서론 | 우리나라 재활용률은 약 60%로 OECD국가 중 2위를 차지하고 있습니다. 하지만 이 통계에는 숨겨진 진실이 있습니다.  쓰레기 처리 과정은 수집-선별-처리 3단계를 거칩니다. 상식적으로 ‘재활용률’이라 함은 3단계를 모두 거치는 재활용 쓰레기의 비율이라고 생각됩니다. 그러나 우리나라는 선별장까지만 가는 비율을 재활용률로 정의합니다. 선별장에서 실제 처리 단계까지 이뤄지는 재활용 쓰레기의 비율은 공공기관의 통계상 60%입니다. 민간업체가 주를 이루는 우리나라 쓰레기 처리 산업의 특성상 공개되지 않은 데이터를 고려한다면 실제로는 60% 미만일 것입니다.  따라서 쓰레기 처리 3단계를 모두 거치는 재활용 쓰레기의 비율은 60%가 아니라 36%미만인 것입니다.  재활용 쓰레기가 재활용되지 못함으로써 발생하는 경제적 손실, 환경 문제를 딥러닝 기술로 해결해보고자 합니다. |

|  |  |
| --- | --- |
| 1. 본론 | 버려지는 쓰레기 중 재활용 가치가 높은 캔, 페트, 병을 이미지 처리 기술만으로 구분하고자 합니다. |

|  |  |
| --- | --- |
| 1. 데이터 | 1. 데이터셋 구성  * 국내 주요 포털사이트(구글, 네이버, 다음)에서 총 60개의 키워드로 캔, 페트, 병 이미지 총 35,000장 수집 * Annotation이 불가능한 html, gif파일 제외 * 검색어와 유관한 이미지 직접 필터링 * 사용 가능한 데이터 3,500장 수집  1. **데이터 전처리**  * jpg, jpeg, png파일의 shape을 3으로 통일 * RGBA => RGB로 통일 * Train : valid = 8 : 2  1. **Annotation**  * VGG Annotation 웹사이트를 활용해 Segmentation 직접 지정  1. **Augmentation**  * Flip, Rotation, Gray-Scale 활용  1. **최종 데이터셋**  * 이미지 개수: train 11,024개, valid 2,720개 => 총 13,744개 * Segmentation 개수: train 17,760개, valid 5,336개 => 총 23,096개 * 데이터 크기: 980MB |
| 1. 모델 | 1. Classification: Custom CNN, VGG, ResNet  * Pre-Trained: ImageNet * 단일 물체만 있는 정제된 이미지 데이터가 부족해 만족스러운 성능이 나오지 않음 * 최종 Acc: 65%  1. Object Detection: YOLO v3, Inception v3  * Pre-Trained: MS COCO * 물체가 겹쳐있는 경우 사용자가 Box만으로 물체를 구분하기 어려움  1. Segmentation: MaskRCNN  * Pre-Trained: MS COCO / Epoch: 30 / Loss: 0.5794 * 물체에 직접 영역이 씌워져 대량의 물체가 겹쳐져 있어도 구별하기 쉬움 |

|  |  |
| --- | --- |
| 1. 결론 | 1. **기대효과 및 발전방향**  * 재활용 쓰레기를 확실히 분류하고, 명확한 분리배출 가이드를 제공할 수 있음 * 사용자 식별 기술과 하드웨어를 결합해 리워드를 제공하면 분리배출을 장려해 재활용률을 높일 수 있음 * 사용량이 많아질수록 추가 데이터를 확보해 성능이 향상됨 * 다양한 데이터가 확보된다면 3종 이상의 쓰레기를 분류해 세계적인 쓰레기 문제를 해결할 수 있음 |

|  |  |
| --- | --- |
| 1. 인용 | - https://tensorflow.blog/2017/06/05/from-r-cnn-to-mask-r-cnn/  - http://www.robots.ox.ac.uk/~vgg/software/via/via-1.0.6.html  - https://github.com/aleju/imgaug  - https://github.com/matterport/Mask\_RCNN  - https://www.youtube.com/watch?v=uUYR6IEm5VM  - https://github.com/CrookedNoob/Mask\_RCNN-Multi-Class-Detection  - https://github.com/priya-dwivedi/Deep-Learning/tree/master/mask\_rcnn\_damage\_detection  - https://www.nocutnews.co.kr/news/5159798  - https://brunch.co.kr/@hangganread/121  - http://benefit.is/17832  - http://www.superbin.co.kr/new/contents/product.php  - https://youtu.be/0XQtHSY4cCE?t=213 |