

교통사고 과실비율 자동측정 시스템

*최진우, 이유노, 박인규
인하대학교 정보통신공학과

e-mail: chlwsdn29@naver.com, ocard9611@naver.com, pik@inha.ac.kr

Automatic Measurement System for Accident Rate of Accidental Accidents

* Jin-woo Choi, Yu-noh Lee, In-Kyu Park
Department of Information and Communication Engineering
Inha University

Abstract

The study is about a system that automatically measures the rate of negligence in traffic accidents. In CCTV footage, the ratio of negligence of the two vehicles is calculated using the surrounding conditions and vehicle path. Models YOLO and DeepSort were used to track vehicle routes and Mask R CNN to detect traffic accidents. We judged that it would be a dangerous moment if the vehicle's Binding Box overlapped, and decided to make an accident decision on Mask R CNN. The accident decision was judged to be a traffic accident if the masks of the vehicles overlap. The 3D-CNN model determines whether a situation is a traffic accident. The 3D CNN model was created using Dataset built in the GTA5 environment. If a traffic accident is determined in a certain situation, a random forest model is used to impose a rate of negligence on the two vehicles in which the accident occurred. The Random Forest model is made of Dataset, which has features of the vehicle's key x coordinates, the key y coordinates, the change in focus x coordinates,

and the key y coordinate changes. Each traffic accident has a random forest model, and each accident is subject to a rate of negligence using the features of the vehicles in the accident.

I. 서론

현재 크고 작은 교통사고가 주위에서 발생하고 있다. 그리고 교통사고에는 과실 비율이 적용되고 잘못의 정도가 정해지게 된다. 그런데 사고를 처리할 때 필요한 교통사고 과실 비율 판단이 전적으로 주관적이다. 그 주관적 결론을 내리는 것은 교통사고 당사자들이 아닌 보험사 직원들이다. 일반인들도 쉽게 객관적인 교통사고 과실 비율을 알 수 있도록 연구를 진행하였다.

기존연구 중에서 블랙박스 영상을 이용해 교통 사고 과실비율을 측정하는 연구가 있다. 본 연구는 공정하게 볼 수 있는 CCTV 사고 영상을 이용해 과실 비율을 자동으로 판단해주는 시스템을 구축하려 한다. 본 논문에서의 교통사고 상황, 과실 비율의 정보는 참고문헌 [7]대한민국 자동차사고 과실비율 분쟁 심의위원회의 정보를 근거로 두고 있다.

II. 본론

2.1. 시스템 개요

그림 1은 시스템의 전체 프로세스를 간략하게 나타낸 것이다. 본 시스템은 간단하게 다섯 가지로 요약할 수 있다. CCTV 사고 영상을 입력하면 자동 차를 감지하고 tracking하여 각 차량마다 이동경로를 저장한다. 두 차량이 부딪히는지 여부를 판단한다. 사고가 발생했다고 판단하였으면 그 사고가 어떤 case의 사고인지 분류한다. 분류된 케이스에 해당하는 random Forest모델로 사고가 발생한 두 차량의 Feature들을 넘겨준다. 그 Feature들을 사용하여 두 차량의 과실비율을 계산한다. 이 때 사용자는 GUI를 이용해 사고영상 입력을 할 수 있고 영상이 입력되면 사고가 발생하여 과실비율이 표시될 때까지 영상이 재생된다.

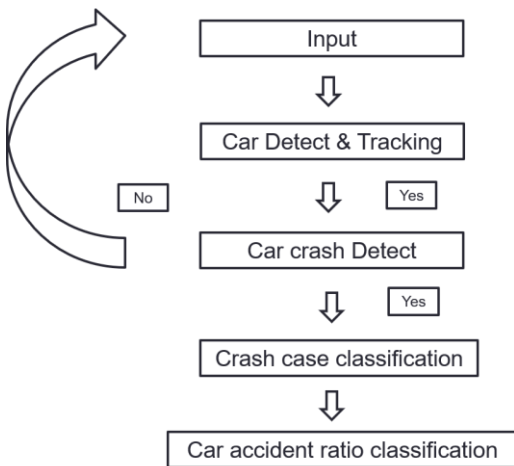


그림 1. 시스템 개요

2.2 상황 Dataset

본 연구를 진행하기 위해선 많은 양의 CCTV 영상 데이터가 필요했다. 한국에서 흔하게 발생한다고 판단되고 다양한 과실비율을 보여줄 수 있는 상황 10가지를 선정하였다. 그러나 해당 상황에 알맞은 CCTV 영상데이터를 구하기는 힘들었고 실제와 비슷한 GTA5 환경에서 Data를 구축하였다.

Augmentation을 활용하여 기존데이터에 8배를 증가시켜 Dataset을 구축하였다.

상황	Data 수	Augmentation
차선 변경	201	1608
역주행	201	1608
주차 또는 정차 중 사고	201	1608

주행 중 추돌	201	1608
주차장 사고	201	1608
차도가 아닌 곳에서 진입	201	1608
T자 교차로에서 직진 대 좌, 우회전 사고	201	1608
정차 후 출발 사고	201	1608
사거리에서 좌회전 대 좌회전 사고	201	1608
이면 도로	201	1608

표 1. 각 case별 dataset 수

2.3 Car Detect & Tracking

차량을 감지하고 각 차량마다 이동경로를 얻기 위해 YOLO, DeepSort 모델을 사용하였다. YOLO는 이미지를 SxS의 셀로 나누고 각 셀마다 하나의 객체를 예측하여 클래스 맵을 만든다. 미리 설정된 개수의 boundary box를 통해 객체의 위치와 크기를 파악하여 객체를 감지한다.

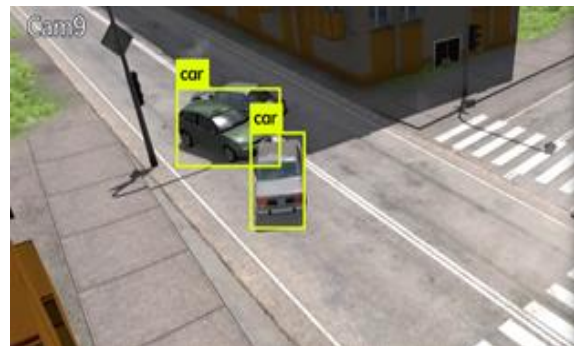


그림 2. YOLO를 이용한 car detect

DeepSort는 칼만필터와 헝가리안 알고리즘으로 구현된 sort와 딥러닝의 합성어이다. 이미지의 feature가 비슷하거나 기존의 속도를 기반으로 다음 위치를 예측하여 다음 이미지에서 같은 아이디를 부여한다. 영상을 입력하면 영상에서 차량을 감지하고 차량의 ID를 부여하여 tracking 하였다. 차량 각각의 위치를 모든 프레임마다 저장하여 차량의 이동경로를 얻었다. 이때 YOLO 모델을 통해 얻은 각 차량의 bounding box의 중점 x, y좌표, 각 좌표의 변화량 총 4개의 feature를 저장한다. 사고 발생여부를 판단하기 위해 차량 두 대가 가까워지는 경우 사고 발생 가능성이 높은 것으로 판단하였다. 이를 두 차량의 bounding box가 겹치면 두 bounding box를 포함하는 영역에 대해 mask R CNN을 실행해 사고 발생여부를 판단한다.



그림 3. DeepSort를 이용한 car tracking

2.3 Crash Detect

R-CNN의 일종으로 객체가 있을 법한 영역에 대해 object detection을 하여 객체를 감지한다. 그리고 각 영역에서 픽셀이 객체인지 아닌지를 masking하여 semantic segmentation을 하는 모델이다. YOLO를 통해 얻은 영역에서 mask R-CNN을 실행하면 두 자동차의 mask를 얻을 수 있다. 그림 4처럼 mask가 있는 곳은 흰색, 그렇지 않은 곳은 검은색으로 표시하여 백 그라운드와 포 그라운드를 분리하였다. 사고가 나면 두 차량은 맞닿아 있게 되므로 두 흰색 영역이 겹칠 때 사고로 판단하도록 하였다. 이 때 차량이 있는 부분만으로 사고를 판단하면 miss detection이 발생하는 경우가 있었다. 때문에 Opencv의 ellipse함수를 이용해 영역에 맞는 타원을 추가하여 해결하였다.

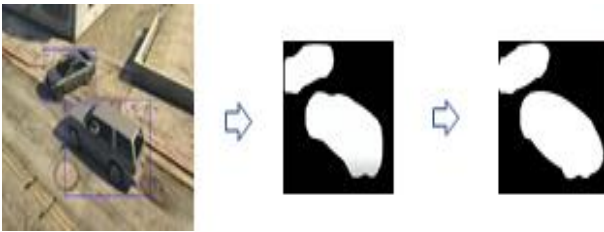
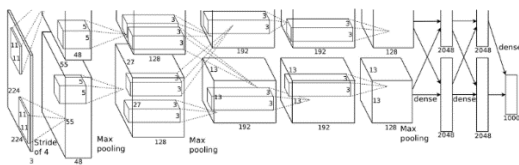


그림 4. mask R CNN으로 얻은 mask와 타원을 추가한 형태

2.4 Case Classification



출처: Krizhevsky 외 [2012]

그림 5. 3D-CNN 구조

사고의 case를 분류하기 위해 3D-CNN을 사용하였다. 3차원 CNN은 2차원 CNN을 시간 축으로 한 차원 확장시킨 형태의 인공 신경망이다. 2차원 CNN은 일반적으로 이미지를 입력 받아서 그 공간적인 특성을 찾아내어 이미지를 분류하거나, 그와 관련된 응용 분야에 뛰어난 성능을 보이지만, 시간 정보를 다룰 수 없기 때문에 동영상 데이터는 처리할 수 없는 한계가 있다. 반면 3차원 CNN은 컨볼루션 연산과 풀링 연산을 시간 성분까지 함께 계산하도록 함으로써, 영상 데이터의 특징을 뽑아 낼 수 있다는 장점이 있다.

$$v_{ij}^{xyz} = \tanh \left(b_{ij} + \sum_m \sum_{p=0}^{R_i-1} \sum_{q=0}^{Q_i-1} \sum_{r=0}^{L_i-1} w_{ijm}^{pqr} v_{(i-1)m}^{(x+p)(y+q)(z+r)} \right) \quad (1)$$

(1)은 3D 컨볼루션의 식이다. 여기서 (x,y,z)는 feature map/volume의 좌표이고, (p,q,r)에서 앞의 두 index는 kernel의 spatial dimension index이다. r은 temporal dimension index이다. j/m은 feature map/volume을 의미하고 l는 convolution layer를 의미한다.

video clip의 size는 $c \times l \times h \times w$ 를 가지며 여기서 c,l,h,w는 각각 channel의 수, frame의 길이, frame의 높이 및 폭을 나타낸다.

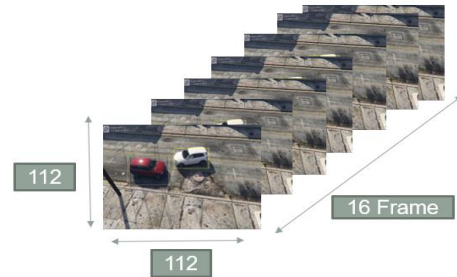


그림 6. 16프레임으로 이루어진 한 Clip

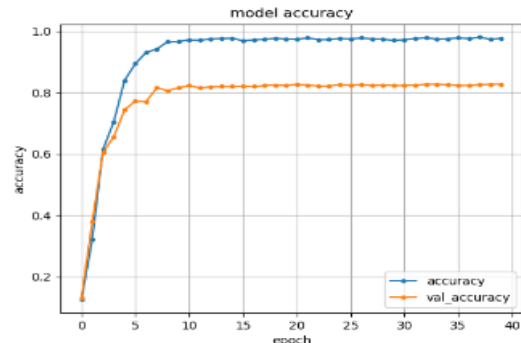


그림 7. 3D-CNN accuracy 그래프

본 연구는 가로 112px 세로 112px 크기의 16프레임씩을 한 clip으로 보고 학습을 진행하였다. 모델은 5개

의 Layer로 이루어져 있으며 kernel size는 $3 \times 3 \times 3$ 이다. 컨볼루션 layer의 filter는 64,128,128,256,256으로 학습하였다. 학습한 모델을 가지고 10가지 상황을 분류하는데 사용하였다. 정확도는 그림 7과 같이 약 81 ~ 82%의 정확도를 갖는다. Epoch은 40, Batch size는 16으로 학습하였다.

2.5 과실 비율 Dataset구축 툴 제작 및 Dataset 구축

사고 난 두 차량의 추적 정보들을 이용하여 두 차량에 과실비율을 부과하였다. 과실 비율을 부과할 모델은 Random Forest모델을 사용하였다. 때문에 Random Forest모델에 학습시킬 Dataset을 구축하였다. 선택한 차량을 추적하는 기능과 필요한 특징들을 자동 추출하여 csv파일로 자동 저장하는 툴을 개발하였다.

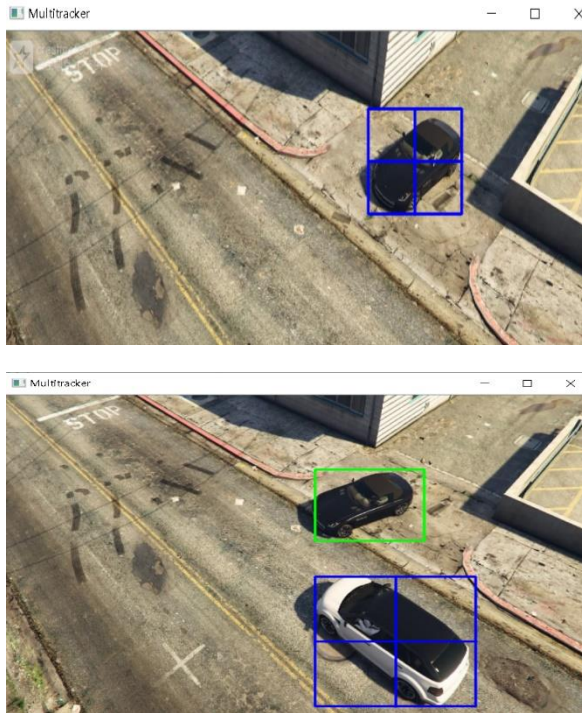


그림 8. 각 라벨에 맞는 차량에 ROI 그리기



그림 9. 사고 발생 전까지 차량 추적

xposition	yposition	x_change	y_change	label
301.5	184.5	0	0	0
296.5	183	5	1.5	0
290.5	180.5	6	2.5	0
284.5	178	6	2.5	0
279.5	175.5	5	2.5	0
274.5	173	5	2.5	0
268.5	170.5	6	2.5	0

xposition	yposition	xspeed	yspeed	label
158	45.5	0	0	1
156	48	2	2.5	1
153	49.5	3	1.5	1
151	53	2	3.5	1
148	55.5	3	2.5	1
146	58	2	2.5	1
144	61.5	2	3.5	1
142	65	2	3.5	1

그림 10. csv파일에 자동 저장

그림 9에서는 차량라벨에 맞게 ROI를 각각 그려준다. ROI안에 있는 차량은 KCF tracker에 의해 tracking하게 된다. 추적하면서 Feature인 bounding box의 중점 x좌표, 중점 y좌표, 중점 x좌표의 변화량, 중점 y좌표 변화량을 저장한다. 사용자가 원하는 순간까지 Feature들을 저장할 수 있으며 다음 동영상으로 넘기게 되면 라벨에 해당되는 csv파일에 Feature들이 자동으로 저장된다.

상황	Label 0	Label 1
차선 변경	4382	4460
역주행	2293	2117
주차 또는 정차 중 사고	4245	4235
주행 중 추돌	1912	1928

주차장 사고	3085	3094
차도가 아닌 곳에서 진입	1704	1713
T자 교차로에서 직진 대 좌, 우회전 사고	3122	3115
정차 후 출발 사고	1333	1316
사거리에서 좌회전사고	377	356

표 2. 각 case별 과실비율 dataset 수

2.6 Random Forest

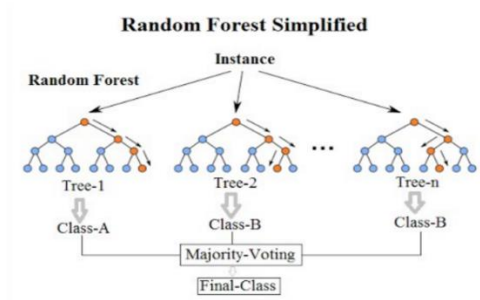


그림 11. Random Forest 모델 구조

Random Forest 모델은 성능과 변동의 폭이 크다는 단점을 갖고 있는 의사 결정 트리의 단점을 극복하기 위해 등장하게 되었다.

Random Forest 모델은 분류, 회귀 분석 등에서 사용되는 앙상블 학습 방법의 일종으로, 훈련 과정에서 구성한 다수의 결정 트리로부터 분류 또는 회귀 분석을 출력함으로써 동작한다.

본 논문의 Random Forest 모델은 차 중점의 x좌표, 차 중점의 y좌표, 차 중점의 x좌표 변화량, 차 중점의 y좌표 변화량으로 특징을 갖는다.

본 논문은 case 마다 작은 Random Forest 모델이 있으며 3D-CNN으로 상황이 결정되면 각 차량에 과실 비율을 해당 case의 Random Forest 모델을 이용하여 부과한다.

상황	Precision	Recall	F1-score
차선 변경	77%	76%	76%
역주행	82%	83%	83%
주차 또는 정차 중 사고	99%	99%	99%
주행 중 추돌	88%	87%	87%
주차장 사고	99%	99%	99%

차도가 아닌 곳에서 진입	91%	91%	91%
T자 교차로에서 직진 대 좌, 우회전 사고	91%	91%	91%
정차 후 출발 사고	98%	98%	98%
사거리에서 좌회전 대 좌회전 사고	92%	91%	91%

표 3. 각 case의 Random Forest accuracy

III. 구현

일반인들도 쉽게 이용할 수 있도록 GUI를 구축하였다. CCTV 영상을 GUI에 넣게 되면 사고를 감지할 때까지 영상이 진행된다. 영상에서 16프레임이 지나면 예측도와 예측한 상황을 표시한다. 예측도를 각 상황마다 더한다. 가장 큰 값을 갖고 있는 상황이 해당 CCTV 영상의 사고 상황이라고 결정하였다. 사고가 발생하면 사고가 발생한 두 차량의 이동정보를 통해 과실 비율이 결정되고 각 차량에 과실 비율을 표시한다.



그림 2. 교통사고 과실 비율 자동 측정 시스템 GUI

IV. 결론 및 향후 연구 방향

이 시스템은 발전 가능성이 많다. 현재는 10가지 상황을 분류해서 과실 비율을 부과하는 시스템이다. 각 상황들은 표지판, 신호등의 영향을 받지 않는다. 때문에 표지판과 신호등까지도 고려하는 시스템으로 발전시킬 예정이다. 그리고 차대차의 사고만 고려했기 때문에 사람 대 차, 오토바이 대 차 등 다양한 상황들도 추가시킬 예정이다. 위와 같은 상황들을 추가시키면 일반인도 간단하게 사용할 수 있는 자동

과실비율 측정 시스템을 구축할 수 있을 것이다.

이 시스템은 오직 CCTV 영상으로만 사고를 감지한다. 때문에 사고가 나지 않았음에도 차량이 다른 차량의 뒤로 숨는 경우에 사고로 감지하는 한계점이 존재한다.

참고문헌

- [1] 우윤희, 최권택, 이정근, 실시간 응용을 위한 혼합형 다중 객체 추적 시스템의 구현, KITT, 2019.11.30
- [2] V. Machaca Arceda, Fast car crash detection in video, Universidad La Salle, Universidad Nacional de San Agustim, 2018.10
- [3] 윤종후, 정기현, snake 알고리즘을 이용한 실시간 이동물체 검출, 대한전자공학회, 2008.06
- [4] 허민호, 김경민, 장병탁, 딥러닝 기반 비디오 스토리 학습기술, 한국멀티미디어학회지, 2016.06
- [5] 김지섭, 남장군, 장병탁, 딥러닝 기반 비디오 분석 기술, 정보과학지, 2015.09.21
- [6] Joseph Redmon, Santosh Divvala, You Only Look Once: Unified, Real-Time Object Detection, University of Washington, 2016
- [7] <https://accident.knia.or.kr/myaccident1>
- [8] Kaiming He, Georgia Gkioxari, Piotr Dollar, Ross Girshick, Mask R-CNN, Facebook AI Research (FAIR), 2017
- [9] Nicolai Wojke, Alex Bewley, and Dietrich Paulu,. Simple online and realtime tracking with a deep association metric, In Proc. ICIP, pages 3645–3649, 2017