
 KETI 한국전자기술연구원 <small>Korea Electronics Technology Institute</small>	속도 추정 알고리즘 성능분석 보고서				
	문서분류	문서관리자	버전	최초작성일	최종수정일
	보고서	장수현	1.0	2021년 2월 15일	2021년 3월 8일

속도 추정 알고리즘 성능분석 보고서

작성자	한국전자기술연구원	한서우
	한국전자기술연구원	장수현

 KE·TI 한국전자기술연구원 <small>Korea Electronics Technology Institute</small>	속도 추정 알고리즘 성능분석 보고서				
	문서분류	문서관리자	버전	최초작성일	최종수정일
	보고서	장수현	1.0	2021년 2월 15일	2021년 3월 8일

차 례

1. 차량 속도 추정 방법	3
1.1 Virtual detection line(VDL)을 이용한 속도 추정	3
1.2 글로벌 좌표를 이용한 속도 추정	4
1.3 Optical flow를 이용한 속도 추정	4
2. VDL을 이용한 속도 추정 성능 향상 방법	7
2.1 Bounding box point 위치 설정	7
2.2 VDL 선정 방법	8
2.3 거리, 시간 정확도	9

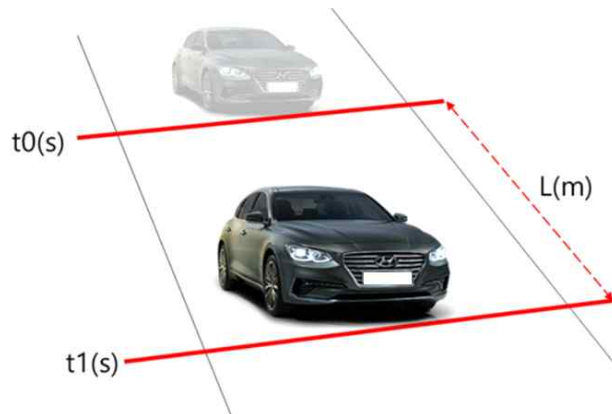
KE-TI 한국전자기술연구원 Korea Electronics Technology Institute	속도 추정 알고리즘 성능분석 보고서				
	문서분류	문서관리자	버전	최초작성일	최종수정일
	보고서	장수현	1.0	2021년 2월 15일	2021년 3월 8일

1. 차량 속도 추정 방법

1.1. Virtual detection line(VDL)을 이용한 속도 추정

▶ VDL 구간 1개 이용

- 차로를 가로지르는 VDL 2개를 그려 차량이 각 line을 지나는 시간을 계산하여 차량의 순간 속도를 계산함
- 간단하며 빠르게 계산할 수 있다는 장점이 있으나 영상의 fps, bounding box(Bbox) 좌표, 카메라 calibration 유무에 따라 거리 오차가 발생할 수 있으므로 속도 정확도가 낮아질 수 있음
- 도로의 정체 상황 발생 시 정확한 속도를 계산하기 어려움

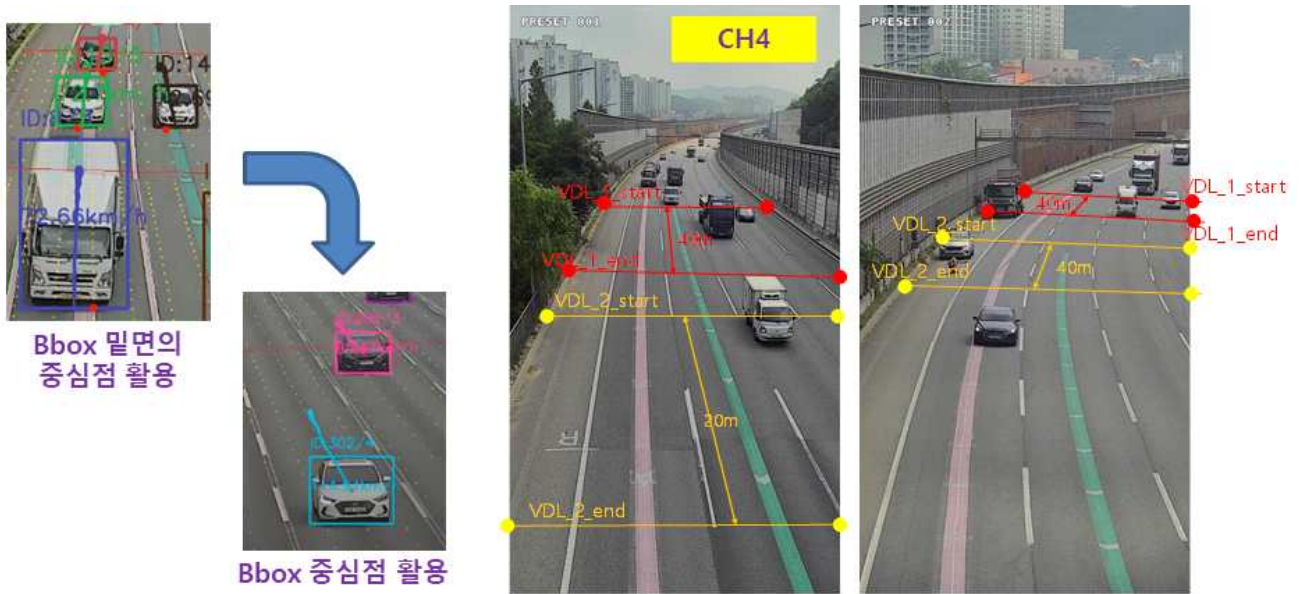


[그림 1] virtual detection line을 이용한 속도 추정의 예시

▶ VDL 구간 2개 이용

- 차로를 가로지르는 VDL 구간을 2개로 설정하고 차량이 각 line을 지나는 시간을 계산하여 차량의 순간 속도를 계산함
 - (목표) 활용 ground truth(GT) 부재로 정확한 속도 추정보다는, 각 VDL 구간에서의 속도 추정 오차범위를 줄이는 것을 목표로 하였음
 - (제안1) Bbox 아래 중심점이 아닌, Bbox 중심점을 기준으로 트래킹 결과를 속도 추정에 활용
각 구간에서 추정오차 줄였으나 효과 크지 않음
 - 또한, VDL 2개 활용하여 속도 업데이트 구간 추가
 - (결과) 개선 효과 크지 않음. 각 VDL 구간에서 추정된 속도 오차 범위가 여전히 큼.

KE-TI 한국전자기술연구원 Korea Electronics Technology Institute	속도 추정 알고리즘 성능분석 보고서				
	문서분류	문서관리자	버전	최초작성일	최종수정일
	보고서	장수현	1.0	2021년 2월 15일	2021년 3월 8일



[그림 2] 2개의 VDL 구간 활용하여 속도 추정

1.2. 글로벌 좌표를 이용한 속도 추정

- 영상에 일정한 간격으로 점을 찍어 차량이 이동한 거리를 점 간격으로 계산함
- 정체 상황과 같이 차량의 속도가 일정하지 않은 순간에도 속도의 변화를 계산할 수 있음
- 영상의 차선이 vanishing point에 가까워질수록 일정한 간격으로 점을 찍기 힘들고 점 간의 간격을 정확히 계산하기 어려움



[그림 3] 일정한 간격으로 생성된 글로벌 좌표의 예시

1.3. Optical flow를 이용한 속도 추정

- Optical flow란 연속된 frame 사이의 픽셀 변화량을 계산하는 방법으로 고속도로 CCTV 환경에서 배경은 정지되어 있지만, 차량은 매 frame마다 움직이기 때문에 optical flow를 속도 추정에 활용함
- 글로벌 좌표를 이용하여 계산한 속도 정보를 optical flow의 변화량으로 보정하여 차량 속도 추정에 대한

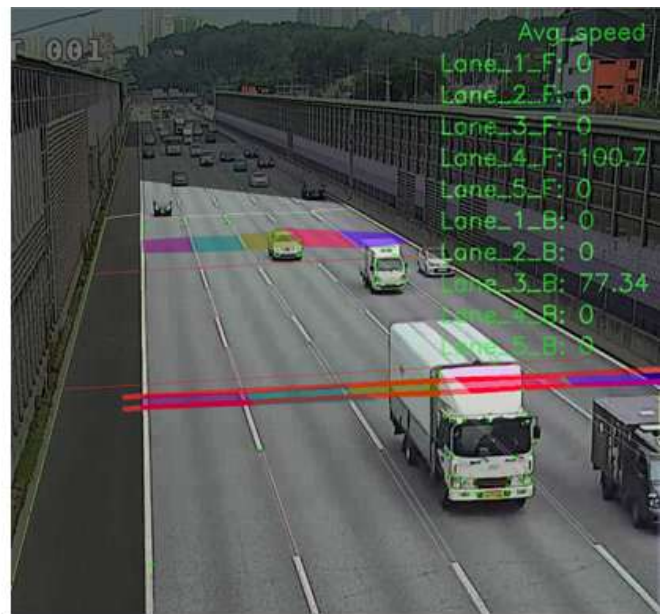
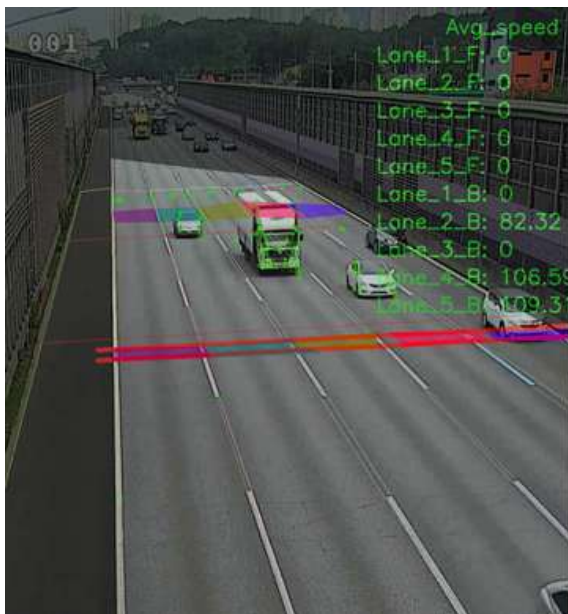
KE-TI 한국전자기술연구원 Korea Electronics Technology Institute	속도 추정 알고리즘 성능분석 보고서				
	문서분류	문서관리자	버전	최초작성일	최종수정일
	보고서	장수현	1.0	2021년 2월 15일	2021년 3월 8일

정확도를 보완하기 위해 검토



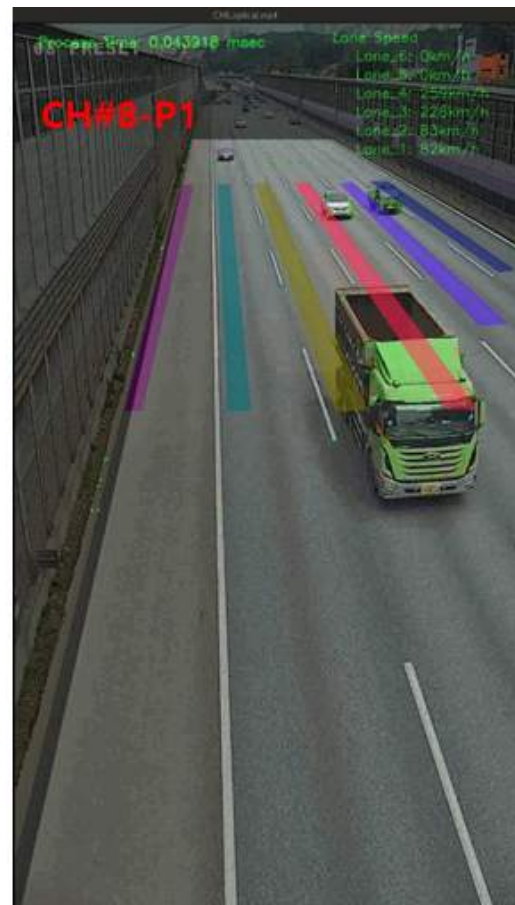
[그림 4] optical flow를 이용한 픽셀의 변화량

- 카메라 각도로 인해 차량이 자기 차선뿐만 아니라 옆 차로 침범. 따라서 차량 ID별 속도 추정 불가하며, 차로별 평균속도 추정에 활용 검토
- 구현 복잡도를 고려하여, sparse optical flow 방식을 활용하여, 차로별 속도 추정에 활용하였으나, CCTV가 설치가 도로를 직각으로 보고 있지 않아, 차로별 속도 추정에 활용될 pixel 정보를 정확히 추출하기 힘들어 정확한 차로별 평균속도 추정이 힘들어, 본 과제에서 적용하지 않기로 하였음.



[그림 5] sparse optical flow 활용 속도 추정 (차로별 속도 추정에 활용할 pixel 영역 정의) - 가로형

KE-TI 한국전자기술연구원 Korea Electronics Technology Institute	속도 추정 알고리즘 성능분석 보고서				
	문서분류	문서관리자	버전	최초작성일	최종수정일
	보고서	장수현	1.0	2021년 2월 15일	2021년 3월 8일



[그림 6] sparse optical flow 활용 속도 추정 (차로별 속도 추정에 활용할 pixel 영역 정의) - 세로형

KE-TI 한국전자기술연구원 Korea Electronics Technology Institute	속도 추정 알고리즘 성능분석 보고서				
	문서분류	문서관리자	버전	최초작성일	최종수정일
	보고서	장수현	1.0	2021년 2월 15일	2021년 3월 8일

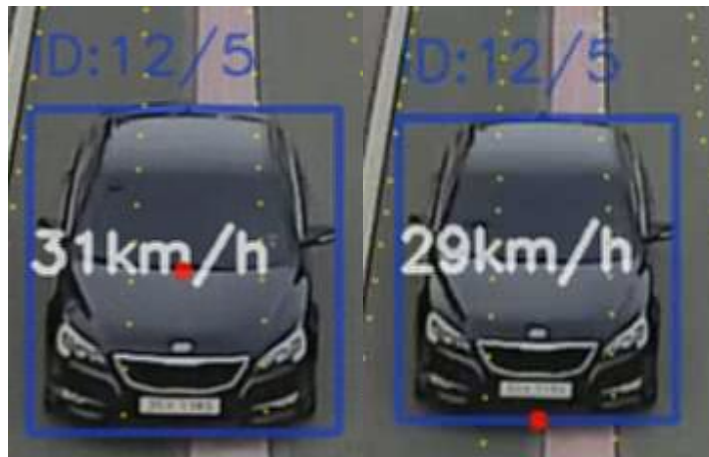
2. VDL을 이용한 속도 추정 성능 향상 방법

- ▶ VDL 구간 2개 이용VDL을 이용하여 속도를 추정하는 경우, 선행적으로 카메라 calibration이 진행되어야 하며 detection, tracking 알고리즘이 인퍼런스 영상 환경에 맞춰 학습이 진행되어야 함
- ▶ VDL을 이용하여 속도 추정 공식
그림 1의 상황일 때, VDL을 이용하여 속도를 추정할 시 아래와 같은 수식으로 속도를 추정할 수 있음

$$speed(km/h) = \frac{L(m)}{t1(s) - t0(s)} \times 3.6$$

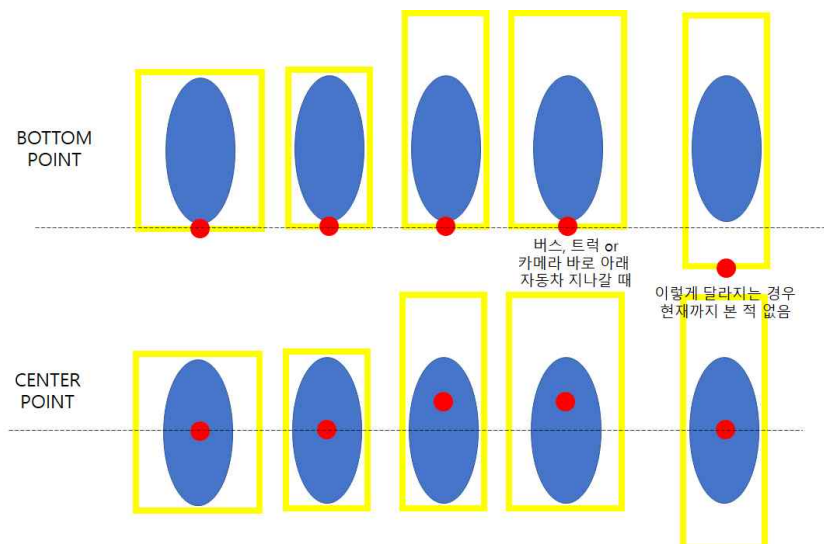
2.1. Bounding box point 위치 설정

- Bounding box(Bbox) 위치 예시



[그림 7] Bbox 위치 예시(왼-center point, 오-bottom point)

- Bounding box(Bbox)의 중심점에 따른 오차 분석
그림 8과 같이 Bbox를 center point로 사용할 때보다 bottom point로 사용할 때, detection, tracking의 오차에 크게 구애받지 않음





[그림 8] Bbox 위치에 따른 오차 발생 상황

<div> <div>KE-TI</div> <div>한국전자기술연구원</div> <div>Korea Electronics Technology Institute</div> </div>	속도 추정 알고리즘 성능분석 보고서				
	문서분류	문서관리자	버전	최초작성일	최종수정일
	보고서	장수현	1.0	2021년 2월 15일	2021년 3월 8일


2.2. VDL 선정 방법

- VDL을 어느 위치에 갖냐에 따라 속도 추정 성능이 달라짐
- 곡선 도로, 상행선 하행선이 분리된 도로가 아닌 이상, 상행선과 하행선의 VDL이 평행하게 그어져야 거리 오차가 줄어듦
- 차량 검출 및 트래킹이 잘 되는 위치에 VDL을 그어야, Bbox bottom point가 오차 없이 얻을 수 있음

[표 1] VDL 위치별 평가

VDL 예시 이미지	사용 가능 여부	이유
	○	<ul style="list-style-type: none"> • Detection, tracking이 잘되는 곳에 VDL이 그어져 있음 • 단, 해당 각도에서 detection, tracking 학습이 잘 되어있어야 함
	○	<ul style="list-style-type: none"> • Detection, tracking이 잘되는 곳에 VDL이 그어져 있음 • 단, 해당 각도에서 detection, tracking 학습이 잘 되어있어야 함
	△	<ul style="list-style-type: none"> • 하행선의 경우, detection, tracking이 잘되는 곳에 VDL이 그어져 있음 • 그에 비해, 상행선의 경우 하행선과 다른 차량의 feature를 보여주어 해당 각도에서 detection, tracking 학습이 잘 되어있어야 함

KE-TI 한국전자기술연구원 Korea Electronics Technology Institute	속도 추정 알고리즘 성능분석 보고서				
	문서분류	문서관리자	버전	최초작성일	최종수정일
	보고서	장수현	1.0	2021년 2월 15일	2021년 3월 8일

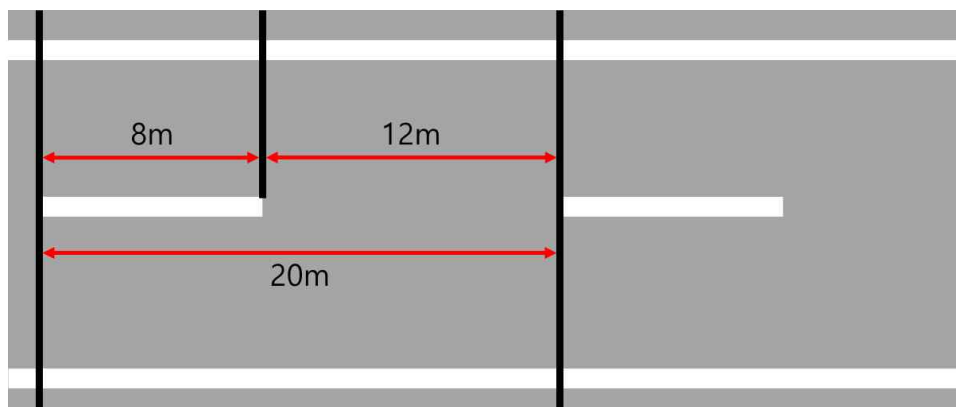
	X	<ul style="list-style-type: none"> • VDL이 위치한 지점의 차량 크기가 매우 작음 • Detection, tracking이 잘 안 되는 곳에 VDL이 그어져 있음
	X	<ul style="list-style-type: none"> • 상행선, 하행선의 VDL이 평행하지 않음 • 상행선의 경우, 1차로와 2차로의 20m 거리가 상이함

2.3. 거리, 시간 정확도

- VDL로 속도를 추정하는 경우, 이동 거리를 이동하는데 걸린 시간으로 나누어 속도를 추정하게 됨
- 따라서, 이동 거리와 이동 시간이 정확해야 정확한 속도값을 얻을 수 있음

▶ 이동 거리

- 대한민국 고속도로 차선길이 규격은 그림 9와 같음
- 고속도로 차선길이 규격을 기반으로 거리를 정의



[그림 9] 고속도로 차선길이 규격

▶ 이동 시간

- 실시간 속도 추정 시, VDL을 지나는 실제 시간을 time stamp를 이용하여 기록(e.g. 파이썬 코드 예시: time.time())
- 실시간이 아닌 오프라인 영상으로 속도 추정 시, 입력 영상의 frame number를 이용