

論文99-36S-1-10

이동창을 이용한 차선 인식 및 장애물 감지

(Lane Recognition and Obstacle Detection Using Moving Windows)

崔勝郁*, 李章明*

(Sung Yug Choi and Jang Myung Lee)

요 약

본 논문은 주행중의 자동차에 장착된 카메라의 입력 영상으로부터 이동창을 활용하여 차선을 인식하고 장애물을 감지하는 기법을 기술한다. 영상 정보로부터 장애물을 감지하기 위해서 차선의 위치를 빠른 시간에 추출하는 것이 매우 중요하다. 이를 위하여 한 프레임의 영상에서 차선의 입력이 예상되는 일부분만을 선정하는 기법이 일반적이다. 본 논문에서는 주행 차량의 영상 정보로부터 장애물을 감지하기 위하여, 도로의 곡률에 따라 차선의 입력 예상 위치를 추정하여 크기가 조절된 이동하는 창을 설정하여 정확한 차선의 위치를 추출하고, 나아가 주행 차선 내에 위치하는 장애물을 검출할 수 있는 기법을 제안한다. 이를 고정창을 이용하는 경우와 레이저 혹은 레이더 센서를 사용하는 경우와 비교하여, 정확도가 높음을 보였다.

Abstract

To detect obstacles and lane-markers for driving vehicles, a new moving window scheme where moving windows are assigned to an image frame captured by a camera is addressed. For the detection of obstacles, it is important to estimate lane-markers precisely and rapidly. For this purpose, selecting some parts of an image frame at the expected lane locations, i.e., selecting windows are generally adopted for extracting lane-markers efficiently. In this paper, a new scheme that extracts lane-markers precisely by assigning variable size windows at the expected locations of lane-markers considering the road curvature and finally detects obstacles within a driving lane is proposed. The accuracy improvement using this moving window scheme is showed by comparing to the conventional fixed window method and to using radar or laser sensors.

I. 서론

각국에서는 교통사고에 의한 손실을 최소화하기 위하여 자동차의 안전성을 강조한 차세대 지능형 고안전 자동차의 개념을 도입하고 있다. 이때, 주행 차량의 안전 확보를 위해 CCD 카메라를 포함한 여러 가지 센서를 이용함으로써 운전자의 주행 안전성 및 조작성을 향상시켜 자동차의 안전성을 극대화하는 노력을 경주

하고 있다^{[1],[2]}. 이러한 지능형 자동차를 기반으로 한 자율 주행 시스템으로 발표되어진 개발 체계의 사례로는 미국에서의 IVHS(intelligent vehicle highway system)중에 AVCS(advanced vehicle control system)와 일본의 ASV(advanced safety vehicle) 등이 있다. 국내에서도 이에 준하여 여러 시스템이 발표되고 있다^{[3],[4],[10]}. 이에 따라 주행 중 운전자의 부담을 경감시킬 목적으로 추돌 경고 시스템에 대한 연구가 행해지고 있으며, 레이더(radar), 레이저(laser)센서 등을 활용하고 있다. 그러나, 레이더 및 레이저 센서는 직진성이 강하여 복잡한 운전 환경과

* 正會員, 釜山大學校 電子工學科
(Dept. of Elec. Eng., Pusan National University)
接受日字:1998年3月5日, 수정완료일:1998年10月7日

굴곡이 많은 우리 나라의 도로 환경에서는 오경보를 발생할 확률이 높기 때문에, 이를 감소시키기 위해서는 CCD 카메라를 사용한 차선 정보 추출에 의한 장애물 검출 알고리즘의 연구가 필요하다. 그러나, CCD 카메라를 이용하는데 있어 정보처리량(640×480 [pixel])의 증가로 인한 연산 속도의 저하와, 날씨 등에 의한 조명의 영향을 크게 받는 문제점이 있다.

본 논문에서는 도로의 차선(lane mark)에 대한 정보를 구하는데 있어, 전처리 과정에서 LUT(look-up table)를 사용한 B.P.T(band-pass threshold) 기법을 새롭게 제안하였다. 또한, 차선 추출 과정에서 차선을 따라 가면서 설정된 sub-block, 즉, moving window를 사용하는 기법을 개발하여 종래의 고정된 sub-block을 사용하는 경우에 비교하여, 차선 인식의 정확도를 높이고 정보 처리량을 줄이도록 하였다.

1. 영상 처리 기법

본 절은 전처리 및 장애물 검출에 관련된 기본적인 영상처리 기술들에 대하여 기술한다. 영상 처리의 과정에서 카메라에서 취득은 원근법에 의한 투사로 가정한다. 실제 물체의 정확한 위치나 크기의 정보를 영상으로부터 얻고자 하는 경우에는 카메라의 focal length 및 scale factor 등의 내부 파라미터의 캘리브레이션이 필요하다^[5] 본 연구에서는 취득된 차선 정보로부터 차선의 진행 방향 추출만을 요구하므로 카메라 캘리브레이션은 필요하지 않다. 또한, 입력되는 도로 영상은 차선과 도로의 대비가 뚜렷한 맑은 날씨의 고속 도로 주행상황을 전제로 하고 있다.

1) LUT를 사용한 B.P.T 기법에 의한 전처리

디지털 영상 정보로부터 이진화된 정보를 얻기 위하여 영상처리 보드에서 제공되는 LUT를 활용하여 B.P.T 기법을 실현하였다. 기존의 binary threshold 기법은 단순히 특정 밝기 이상의 정보를 논리 1로 처리한다. 따라서, 차선보다 gray 레벨이 높은 모든 신호는 논리 1로 처리되기 때문에 차선 탐색에 어려움을 남긴다. 이러한 문제를 해결하기 위하여 차선 정보에 해당되는 밝기값을 기준으로 TH_L 및 TH_H의 두 개의 임계값을 설정하여 이 임계값 범위안에 들어오는 gray 레벨을 갖는 신호만을 논리 1로 처리하도록 하였다. 두 개의 임계값에 의해 구분되어진 이진화값 TH(논리: 1), TL(논리: 0)은 LUT의 출력 G_{out}로 식 (1)과 같이 표시된다.

$$G_{out} = \begin{cases} TL, & 0 \leq G_{in} < THD_L, \quad THD_H < G_{in} \leq 255 \\ TH, & THD_L \leq G_{in} \leq THD_H \end{cases} \quad (1)$$

식 (1)에서 THD_L, THD_H, TH, TL은 0 보다 크고 255 보다 작은 범위에 임의의 정수이며, 0 < TL < TH < 255 인 관계에 있다. 디지털 영상 입력의 gray 레벨이 기억소자의 어드레스로 사용되어 기억소자에 저장된 데이터가 출력되는, B.P.T 기법을 LUT로 구현하는 것에 대하여 그림 1에 나타내었다.

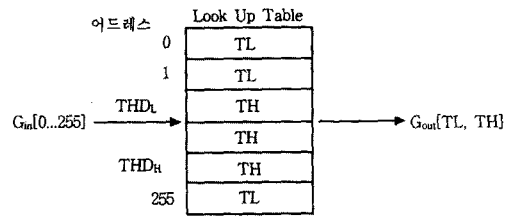


그림 1. B.P.T 방법의 LUT 실현

Fig. 1. Implementation of B.P.T technique using LUT.

도로 영상에 산재해 있는 차선보다 밝은 값의 잡음을 전처리 과정에서 고속으로 제거하기 위하여 B.P.T 기법이 효율적인 것은 실험에서 보여준다. 그러나, 임계값 THD_L 및 THD_H 설정하는 과정은 도로 정보의 종류에 따라 다양하게 선택되어야 하므로 본 연구에서는 초기 설정은 임의로 하고 후속된 영상에 대해서는 이전 프레임에서 차선으로 인식된 부위의 gray 레벨을 기준으로 일정한 범위의 임계값들을 설정하여 사용할 수 있었다. 만일 B.P.T 기법을 LUT가 아닌 CPU에서 연산한다면 카메라로부터 입력된 영상을 영상 처리 보드의 FM(frame memory)에 저장한 후, 이를 다시 읽어들이어 임계값에 따라 이진화하는 과정을 거치게 된다. 반면에 LUT를 사용하면 카메라로 입력되는 영상이 A/D를 거쳐 FM에 저장되는 중간과정에서 Dual Port RAM을 이용하여 B.P.T 기법을 실행함으로써 이진화된다.

2. 세선화 기법

LUT를 사용하여 이진화 되어진 영상 정보는 잡음 신호를 포함하고 있으므로 직접 차선 정보 추출을 위하여 활용되어질 수가 없다. 따라서 이진화된 영상 정보는 세선화 과정에서^[6] 고주파 잡음 성분들을 제거

하여 차선 정보를 추출하기 적합한 영상 정보로 변환한다.

먼저, 확대 과정에서 3×3 마스크를 사용하여 주변의 8개 픽셀 중에 1개라도 픽셀의 값이 TH의 값을 가지면 마스크의 가운데 위치한 픽셀을 TH로 결정한다. 이는 마스크내부의 픽셀을 $B = \{b_0, b_1, \dots, b_8\}$ 이라 할 때, 해당하는 위치에서의 영상 정보는 $A = \{A_{b_0}, A_{b_1}, \dots, A_{b_8}\}$ 라 하면 식 (2)와 같이 나타낼 수 있다.

$$A \oplus B = \bigcup_{b_i \in B} A_{b_i} \quad (2)$$

확대 루틴에서는 TH(차선에 해당됨) 정보에 섞여 있는 TL 값을 갖는 잡음을 제거하여 깨끗한 차선을 얻는 역할을 수행한다.

이후, 축소 과정에서는 반대로 TL 정보에 섞여 있는 TH 정보를 제거하는 기능을 수행하는 것으로, 3×3 마스크를 사용하여 주변 8개의 픽셀 값이 모두 TH인 경우에만 마스크 중앙의 픽셀 값(p)을 TH로 결정한다. 이는 식 (3)으로 나타낼 수 있다.

$$A \ominus B = \{p \mid B_p \subseteq A\} \quad (3)$$

마지막으로 확대와 수축루틴을 거친 후, 차선으로 인식되는 TH 정보의 횡방향 굽기가 한 픽셀이 될 때까지 세선화 과정을 반복한다. 세선화 과정은 일반적으로 4~5회 정도의 반복 작업에 의하여 완료된다.

지금까지 설명된 세선화 기법은 LUT를 사용한 B.P.T 기법과 달리 CPU의 연산에 의해 실행된다. 따라서, 다음에 설명되어질 moving window에 지정된 부분에만 세선화 기법을 적용한다.

3. Moving window 기법

연속 영상의 한 프레임(640×480 픽셀)으로부터 차선 정보를 추출하는 과정에서 메모리 용량의 한계와 이에 따른 속도 저하를 피하기 위해, 필요한 정보만을 sub-block으로 설정하여 연산 시간을 단축시키고자 하는 노력이 진행되고 있다^{[7], [8], [9]}. 그러나, 주행 차량의 경우에는 도로의 곡률(횡단 선형: 수평을 기준으로 좌우 변화)이 지속적으로 변화하기 때문에 한 영상 프레임 내에 차선이 어디에 위치할 것인가에 대한 예측은 매우 어렵다.

본 논문에서는 기존의 특정 크기의 sub-block을

특정 위치에 설정하여 영상 정보의 처리량을 줄이는 방법을 개선하여, 이 sub-block의 위치를 차선을 따라 동적으로 이동시키는 기법을 제안한다. 이때 sub-block의 가로 및 세로 방향 크기의 선정은 자기 다음의 선정 기준에 따른다.

- 가로 기준: 한 sub-block 내에 존재하는 차선이 다음 프레임의 영상에서도 이 sub-block 내에 존재하여야 한다. 또한, 후속 프레임의 영상에서는 전 프레임에서 설정된 sub-block의 check point로부터 새로운 check point를 설정하는 조건은 만족되어야 한다. 이를 위하여, 고려해야 할 요소들은 여러 가지가 있으나, 이 값들 중 CCD 카메라 부착각도는 광축이 도로면에 평행하게 고정하고, 도로의 곡률과 주행 차량의 속도는 변하므로 최대값을 설정하여 이 최대값 하에서 가로 조건을 만족하도록 가로 방향의 sub-block의 크기를 정한다. 참고로 표 1에 한국 도로 공사에서 사용하는 속도에 따른 도로의 곡률 데이터를 나타내었다.

표 1. 도로의 곡률 기준
Table 1. Standard road curvature.

도로의 설계 속도 (km/h)	120	100	80	70	60
최소 곡률 반경 (m)	710	460	280	200	140
각 도 (°)	2.8	3.7	5.0	6.2	8.0

- 세로 기준: 세로 방향의 길이는 차선을 추출할 때, 진행 방향의 정확도를 결정한다. 따라서, 카메라 각도가 일정하게 유지된 상황에서 도로의 곡률(종단 선형: 수직을 기준으로 상하 변화)이 없는 직선 도로를 등간격으로 측정하여 정확한 세로 방향 크기를 결정한다. 결정되어진 sub-block의 세로 크기는 곡선 도로의 경우 영상 프레임에서 차선이 차지하는 부분이 감소하므로, 이 감소 비율에 따라 sub-block의 세로 크기를 줄여 주어야만 정확한 차선을 추출할 수 있다. 본 연구에서는 평지의 직선 도로에서 결정된 sub-block의 세로 크기를 기준으로 하며, 입력 영상이 곡률을 가질 때는 일정 범위의 세로 크기를 만들어 놓고 선택적으로 사용하였다. 이에 따라 sub-block 내부의 y축 좌표는 일정하다.

지금까지 설명된 기준에 따라, 그림 2는 실험에 사

용된 단위 sub-block 내에서 직교 좌표에 대한 기준 위치 정보 및 sub-block의 크기를 나타내고 있으며, 그림 2의 sub-block은 주행 차량에서 가장 먼 위치에 있는 것이다.

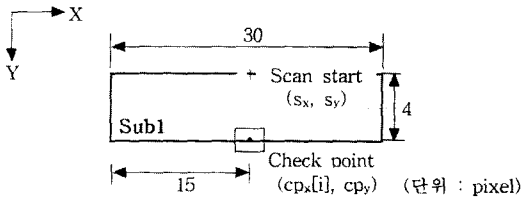


그림 2. Sub-block 내부 정보
Fig. 2. Internal information of sub-block.

2. Moving window에 의한 차선 인식

직교 좌표 상에 불규칙적으로 분포하는 밝기 정보 (도로 정보)를 사용하여 차선을 찾아내는 방법으로 기존 논문에서는 고정된 sub-block에서 직선 추출을 응용하고 있으나^[9], 연속 영상 처리를 위하여 본 논문에서는 moving window를 구성하는 핵심 요소인 check point를 사용한 차선 정보 추출을 제안하고 있다.

1) Moving window에 의한 차선 추출

입력 영상에서 check point를 결정하고, 차선을 추출하는 상황을 순서대로 나타내면 그림 3과 같다. 주행 차량의 좌우에 차선이 존재한다는 가정하에 주행 차량을 기준으로 동일한 거리에 있는 단위 sub-block을 왼쪽, 오른쪽으로 번갈아 조사하는 방법을 사용한다.

Check point 위치 선정은 그림 3과 같이 sub-block 경계의 scan start 위치(s_x, s_y)로부터 시작한다. s_y 및 cp_y 의 값은 sub-block의 세로 기준에 의하여 일정한 좌표값을 가짐을 앞서 설명하였다.

Check point의 결정은 moving window 기법의 핵심이 되는 것으로 scan start 위치로부터 3x3의 마스크를 사용하여 차선을 탐색하는 것으로부터 시작된다. 그림 4는 왼쪽 차선을 위한 하나의 sub-block에 3x3의 마스크를 설정한 후, 차선 탐색을 위한 방향의 우선 순위를 나타내고 있다.

차선 탐색 과정을 거쳐 check point의 위치가 결정되면, 이를 기준으로 새로운 sub-block을 생성한 후 후속 영상을 입력한다.

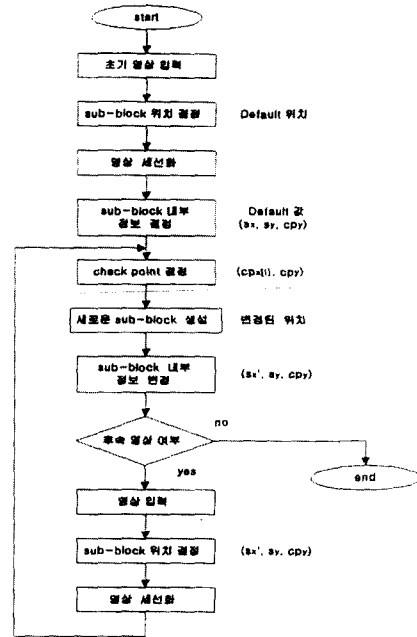


그림 3. Moving window 기법의 흐름도
Fig. 3. Flowchart of moving window technique.

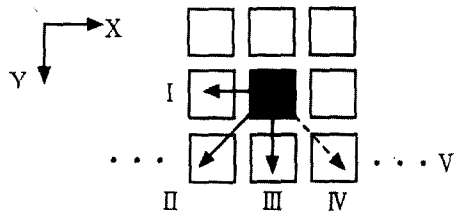


그림 4. 마스크 내부 조사 방향
Fig. 4. Scan direction in a mask.

제안되어진 moving window의 위치가 변경되어 가는 과정을 전체 sub-block 중 왼쪽 차선의 첫 번째 sub-block에 대하여 살펴보면, 그림 5와 같이 설명된다. 그림 5의 (a)는 사전에 결정되어진 기준 위치에서 Sub1을 설정한 후, 입력된 차선 정보에 대하여 세션화를 실시하는 것을 의미한다. (b)에서는 scan start와 check point의 위치를 Sub1에 표시하였다. (c)는 새로운 영상을 위해 check point를 찾아가는 방법을 보여 주며, 그 결과 check point가 결정되어지는 것을 나타낸다. 이 단계에서 Sub1에 대해 check point가 이동되어 X축 좌표($cp[1]_x$)가 새롭게 결정되어진다. 이 좌표에 의하여 그림 5의 (d)에서 새로운 sub-block인 Sub1'이 생성된다.

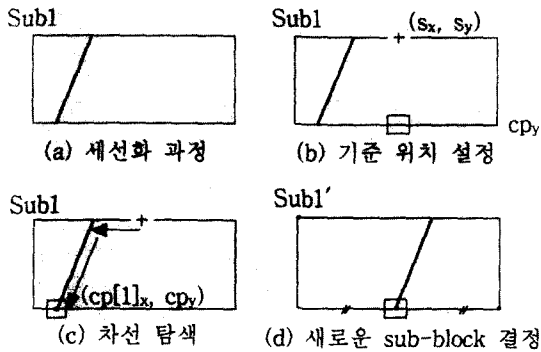


그림 5. Check point 이동 방법
Fig. 5. Determination of check point.

그러므로, 일정한 폭의 직선 도로를 도로 중앙을 따라 달리는 차량의 경우는 한번 설정된 sub-block들을 이동할 필요가 없다는 사실에 주목해야 할 것이다.

2) 차선 진행 방향 인식 알고리즘

Moving window내에 불규칙적으로 존재하는 TH 정보들로부터 구해진 check point는 정확히 차선 위에 위치하므로, 그림 6에서와 같이 check point의 연결로 sub-block내에서 차선의 방향을 추출할 수 있다. 따라서 차선의 방향을 추출하는데 별도의 연산을 필요로 하지 않는다. 구해진 차선에 해당되는 정보는 이전에 사용한 TH 값과 구분하기 위하여 \overline{TH} 로 저장한다.

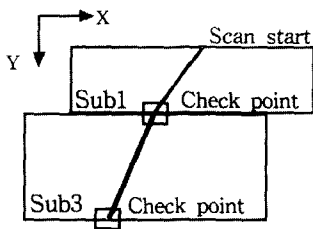


그림 6. 차선 방향 검출
Fig. 6. Detection of lane direction.

3) Fixed window에서의 차선 추출

주행 차량의 방향을 추정하기 위하여 차선이나 도로 영역을 인식할 필요가 있다. 이를 위하여 본 논문에서는 fixed window에 대하여 제안하는 방법과의 비교를 위해 LUT를 사용한 B.P.T 기법과 세선화 기법을 사용하여 check point를 찾은 후, 차선을 추출하였다. 그림 7에서 빗금 친 영역은 fixed window를 나타내며 왼쪽의 1, 2, ..., 9는 거리별 순서를 나타낸다.

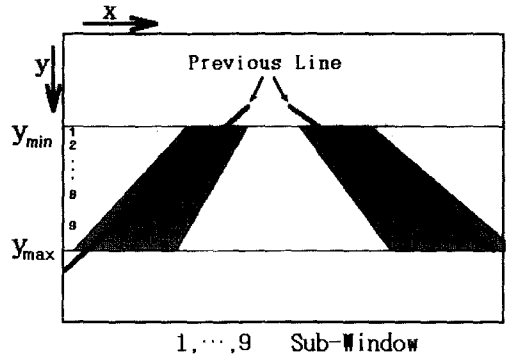


그림 7. 차선 탐색 영역
Fig. 7. Search area for lane marks.

제안한 방법으로 차선을 추출한 후, 직선 도로 및 곡선 도로 상황에서 fixed window를 설정하는 예는 그림 8, 그림 9와 같다. 객관성을 위하여 fixed window 전체 면적을 moving window의 96%로 근접하게 설정하였다.



그림 8. Fixed windows(직선 도로)
Fig. 8. Fixed windows(straight lane).

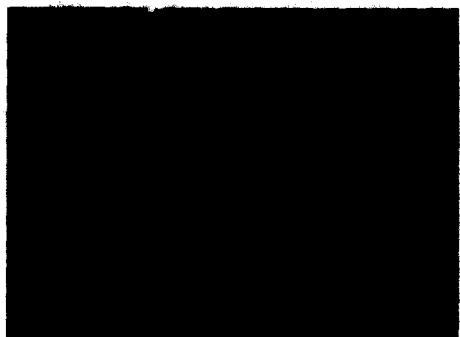


그림 9. Fixed windows (곡선 도로)
Fig. 9. Fixed windows (curved lane).

차선 정보들로부터 차선(도로)의 방향은 무한 원점

(vanishing point)을 사용하여 구할 수 있다. 이때, 무한 원점은 그림 10과 같이 양쪽 차선의 연장선이 영상 프레임 상에서 만나는 점을 의미하며, 도로의 진행 방향과 차량의 진행 방향간의 상관 관계에 따른 무한 원점의 변화는 그림 10과 같이 설명된다.

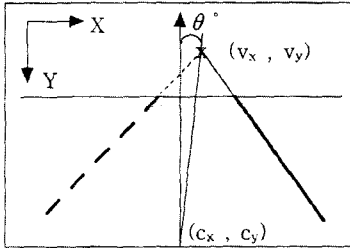


그림 10. 무한 원점의 위치
Fig. 10. Position of vanishing point.

따라서 일반적으로 차량이 주행 차선의 중앙에서 주행할 때 $\theta = 0^\circ$ 에서 정확한 직선 도로를 주행하고 있으며, 연속하는 입력 영상 중의 한 프레임에서는 그림 10과 같이 표시할 수 있다. 실제로, 곡선 도로에서 차선의 방향과 차량의 방향 사이에는 순간적으로 θ° 의 각도를 가지게 되며, 이 θ 값은 즉, 도로의 곡률을 나타내게 된다. 구해진 무한 원점(v_x, v_y)과 θ° 의 관계는 차량의 앞부분 중앙에 해당하는 위치인 (c_x, c_y)를 사용하여 식 (4)와 같이 표현된다.

$$\theta^\circ = \tan^{-1} \left(\frac{v_x - c_x}{|v_y - c_y|} \right) \quad (4)$$

4) Moving window 기법과 비교

곡률 0° 인 직선 도로와 곡률 10° 인 곡선 도로의 영상을 사용하여 fixed window 기법 및 moving window 기법에 의해 도로의 곡률(즉, 차선 방향)을 검출하여 정확도를 비교하였다.



그림 11. Moving windows(직선 도로)
Fig. 11. Moving windows(straight lane).



그림 12. Moving windows(곡선 도로)
Fig. 12. Moving windows(curved lane).

그림 11, 그림 12는 앞서 실험과 동일한 영상에 대하여 moving window를 사용하여 무한 원점을 추출한 결과를 보여 준다.

표 2는 두 기법에 의해 구해진 곡률 값을 보여주며, 직선 도로에서는 두 기법의 정확도에 차이가 문제를 초래하지 않을 정도로 미미하지만 곡선 도로에서는 상당한 차이가 있음을 알 수 있다.

표 2. 도로 영상에서 무한 원점 정보
Table 2. Vanishing point data in road image.

	직선 도로 (0°)	곡선 도로 (-10°)
Fixed-window	0.1°	-3.1°
Moving window	0.1°	-12.3°

IV. Moving window에 의한 장애물 검출

Moving window를 사용하여 차선 추출을 정확하게 빠른 시간에 추출할 수 있음을 살펴보고, 본 절에서는 moving window에 사용된 check point를 활용하여 주행 차량의 자기 차선 내에 존재하는 장애물 인식 방법에 대하여 설명한다.

1. 주행 상황에서 도로의 중앙축 추출

주행하는 차량이 도로의 차선 중앙에 위치할 때, 도로 중앙축은 좌우 차선의 평균 곡률을 갖는 성질과 moving window 의 check point가 차선의 위치를 제공하고 있다는 사실을 이용하여 도로의 중앙축을 추출한다. 중앙축은 차선을 따라 설정된 양쪽 sub-block들의 check point간의 중앙값으로 결정되어, $(c_x [i], c_y)$, $i=1, 2, \dots, n$ 으로 표시된다. $c_x [i$

]의 값은 식 (5)와 같이 구해지며, cl_y 는 cp_y 의 값과 같다.

$$cl_x[i] = \left(\frac{cp'_x[i] + cp''_x[i]}{2} \right) \quad , \quad i = 1, 2, \dots, n$$

$cp'_x[i]$: 왼쪽 sub-block에서의 check point의 x좌표
 $cp''_x[i]$: 오른쪽 sub-block에서의 check point의 x좌표

(5)

2. 장애물 인식 방법

도로의 중앙축을 사용하여 장애물을 인식하는 방법은 그림 13과 같다. 도로의 중앙축 상에 존재하는 TH 정보를 이용하여 장애물로 인식되는 영역을 중앙축을 따라 앞으로 조사해 간다. 이때 도로 정보인 TL이 아닌 TH 정보가 입력되면, 그 지점(ob1)에서 장애물이 시작된다고 판단할 수 있다. 이때 중앙축을 따라 앞으로 조사하여 5 픽셀 이상의 TL 정보가 연속으로 입력되면 ob2 지점을 장애물 종료 위치로 판단할 수 있다. 장애물의 가로 길이는 장애물의 시작(ob1)과 종료(ob2) 위치로부터 오른쪽(right 1, 2), 왼쪽(left 1, 2)으로 조사하므로, 장애물이 도로 한쪽으로 치우쳤을 경우에도 장애물을 정확히 인식할 수 있다. 또한 차선 정보인 \overline{TL} 값을 사용하여 가로 폭의 경계를 결정할 수 있으므로 장애물 인식의 범위를 주행 차선 내부로 제한할 수 있게 한다.

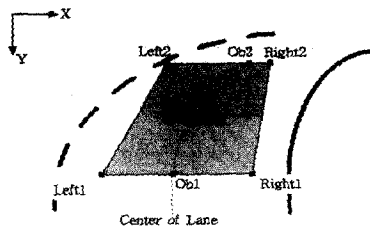


그림 13. 장애물 추출
 Fig. 13. Detection of obstacle.

3. 장애물의 거리 추정

인식되어진 장애물과 주행 차량과의 거리는 영상 취득을 위하여 사용되는 카메라의 초점거리, 장애물의 높이, 부착각도 및 높이가 고정되어 있을 때, 영상 프레임 상에서의 장애물의 위치로부터 계산해 낼 수 있다. 일례로, 초점거리 24 [mm] 인 카메라를 승용차의 백미러 부위(높이 1.1 [m])에 광축을 도로면과 평행하게 설치하여 영상을 취득하는 경우, 전방 차량의 뒷면 트렁크 높이가 0.92 [m] 이면, 같은 높이의 장애물이 주행 차량으로부터 거리별 위치를 대응되는 영상에

서의 좌표로 그림 14에 나타내었다.

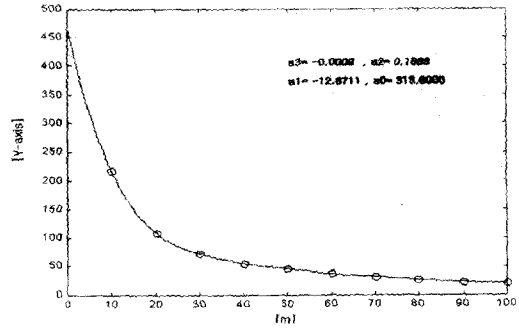


그림 14. 장애물(0.92[m] 높이)의 거리와 영상에서의 대응하는 위치

Fig. 14. Distance of obstacles(height 0.92[m]) and corresponding point in an image frame.

그림 14의 실험 결과는 식 (6)과 같이 근사화될 수 있으며, 이 식에 의해 장애물과 주행 차량과의 거리를 추정할 수 있다.

$$y = a_3x^3 + a_2x^2 + a_1x + a_0 \quad (6)$$

이때, 기존의 moving window 기법은 입력된 도로 영상에서 원근의 거리 정보를 가지는 차선을 같은 크기의 window를 설정하여 추출하고 있다^[7]. 그러나, 입력되는 도로 영상과 이에 해당하는 거리와의 관계를 간과하고 있으므로, 본 논문에서는 식 (6)을 만족하는 moving window의 세로 기준을 이용하여 장애물과의 거리를 추정할 수 있다. 그림 15와 그림 16은 위의 결과를 비교하여 나타내고 있으며, 이때 이전의 moving window 기법에서는 장애물이 몇 미터 전방에 존재하는지 알 수 없으나, 제안하는 기법에서는 장애물이 15 [m] 전방에 위치함을 알 수 있다. 비교의 객관성을 위하여 전처리와 차선 추출은 제안하는 방법을 사용하였다.



그림 15. 이전의 moving window
 Fig. 15. Previous moving window.



그림 16. 제안된 moving window
Fig. 16. Proposed moving window.

V. 실험

본 절에서는 실험 환경에 대해서 설명하고 레이더 및 레이저 센서를 사용하여 장애물 인식하는 경우와 카메라를 사용하여 장애물을 인식하는 실험 결과의 비교를 통하여, 주행 차량의 충돌 방지를 위해서는 차선 정보 추출에 의한 장애물 인식을 가능하게 하는 카메라의 사용이 필수적임을 보인다.

1. 실험 환경

차선 인식 및 장애물 검출을 실제 차량에서 수행하는 것과 동일한 효과를 얻기 위하여 그림 17에 보인 바와 같이 카메라, 레이더 센서, 레이저 센서, 노트북 컴퓨터를 실제 차량에 부착하여 여러 도로 환경에 대한 정보를 동시에 수집하였다. 이를 위하여 승용차에서 운전자의 눈 높이에 가까운 지면에서부터 1.1 [m] 위치에 CCD 카메라를 고정하고, 카메라의 광축을 도로면에 평행하게 설치하였다. 레이더 및 레이저 센서는 영상 디스플레이어 옆에 나란히 설치하여 주행 순간 순간에 전방 장애물로 인식된 물체와의 거리 및 주행 차량과의 상대속도 및 주행 차량의 속도를 노트북 컴퓨터에 시간과 더불어 기록하도록 하였다. 실험에서 사용된 도로 구간은 경부 고속 도로 부산-양산 간이며, 깨끗한 영상 취득을 위하여 주로 낮 시간을 활용하였다. 실험 차량은 대우 슈퍼살롱이며, 카메라는 LC-AH30D(L.G), 레이더 센서는 Falcon radar system, 레이저 센서는 Advantage laser atlanta, grabber 는 UIC512B, 노트북 컴퓨터는 삼성 S500 (pentium-100), main 컴퓨터는 삼성 SPC8320PG (pentium-133)를 사용하였다.

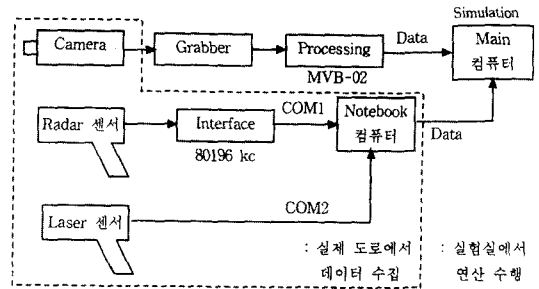


그림 17. 실험 환경 블록도
Fig. 17. Block diagram of experimental environment.

2. 실험 결과 및 검토

1) 실험 결과

우선 주행 차량의 전방 왼쪽 차선에 다른 차량이 달리고 있는 직선 및 곡선 도로 상황에 대하여 무한 원점을 구하고 이에 따른 θ° 를 그림 18 및 그림 19에 나타내었다.



그림 18. 장애물 없는 직선 도로
Fig. 18. Straight lane without obstacles.



그림 19. 장애물 없는 곡선 도로
Fig. 19. Curved lane without obstacles.

레이더 센서의 경우, 빔 폭이 12° 로 설정되어 있으

므로 측정 위치로부터 150 [m] 전방에서 7.9 [m] 폭 이내에 존재하는 물체를 장애물로 검출하게 된다. 주행 차선의 폭이 고속 도로의 경우 3.5 [m] 이므로 옆 차선의 차량까지 장애물로 인식하게 되는 것이다.



그림 20. Moving window와 중앙축(직선 도로)
Fig. 20. Moving windows and a center axis (straight lane).

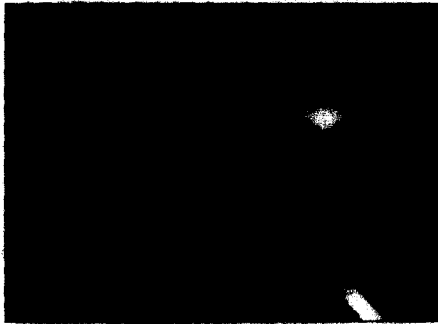


그림 21. Moving window와 중앙축(곡선 도로)
Fig. 21. Moving windows and a center axis (curved lane).



그림 22. 장애물 있는 직선 도로
Fig. 22. Straight lane with obstacles.

주행차선의 전방에 장애물이 존재하는 경우에 대하여 상황은 다음과 같다. 이때 레이저 센서의 경우 빔 폭이 0.16° (2.8mrad)로 측정 위치로부터 150 [m]

전방에서 40 [cm] 폭 이내의 물체만 감지하기 때문에, 그림 22에서처럼 직선 도로에서 장애물이 근접한 경우에는 정확하게 감지되고 있다.



그림 23. 장애물 없는 곡선 도로
Fig. 23. Curved lane with obstacles.

그러나, 도로에 곡률이 있을 때는 그림 23에서처럼 주행 차선의 전방에 장애물이 위치함에도 불구하고 감지하지 못하고 있다. 그림 24, 그림 25는 moving window에 의한 차선 추출 결과를 나타낸다.



그림 24. Moving window와 중앙축(직선 도로)
Fig. 24. Moving windows and a center axis (straight lane).



그림 25. Moving window와 중앙축(곡선 도로)
Fig. 25. Moving windows and a center axis (curved lane).

레이더 센서, 레이저 센서와 비교하여 비전 센서의 측정 각도는 20°로 가장 넓은 범위를 측정하고 있다. 그러나 비전 센서는 차선 정보를 함께 사용할 수 있기 때문에, 장애물을 정확히 인식할 수 있음을 표 III에서 알 수 있다.

표 3. 센서별 측정 대상 장애물 감지 현황
Table 3. Present condition of obstacle detection.

장애물	Radar		Laser		도로 곡률 기준치	도로 곡률 측정치
	실험 차량 속도 [km/h]	상대 차량 속도 [km/h]	추정 거리 [m]		[°]	[°]
무	87	+22	-		1	0.84
	72	+28	-		15	24.42
유	70	-3	47		3	7.61
	84	+10	-		13	22.49

2) 검토

Moving window 기법으로 영상 처리의 시간을 줄이면서 차선 정보를 정확히 추출하고자 하였다. 전술한 moving window의 크기를 최적으로 설정하는 과정을 실험하기 위하여, 적절한 영상을 실제의 도로 상황에서 취득하는데 어려움이 많았다. 이를 위해서는 가변 영상에 대한 auto threshold 방법에 대한 지속적인 연구가 요구되어진다.

향후, 실험환경에 따라 최적 크기의 moving window를 능동적으로 설정하고 카메라의 각도를 조절 가능하게 하며, 도로의 기울기, 자차 속도 등을 위한 센서가 함께 활용된다면 moving window 기법의 우수성은 더욱 돋보일 것이다.

VI. 결 론

본 논문에서는 주행 차량에서 연속적으로 입력되어지는 도로 영상을 가지고 차량의 주행 방향과 전방의 차선 내에 존재하는 장애물 여부를 검출하기 위하여, moving window 기법으로 차선을 추출하는 것을 제안하였다. 추출되어진 차선으로부터 차선간 중앙축을 구하여 이를 근거로 차량 진행 방향을 따라 주행 차선 내부를 탐색함으로써 전방의 장애물이 주행 차선 내에 있는지를 검색할 수 있음을 보였다. 또한, moving window를 이용하는 방법은 기존의 fixed window 방법과 비교하여 볼 때, 곡선 도로에서 차량의 위치로

부터 원거리의 차선을 추적할 수 있음을 확인하였다. 즉, moving window를 사용하는 경우 전체적인 차선의 방향을 결정하는 무한 원점의 위치를 더 정확히 구할 수 있음을 보였다. 또한 거리에 따라 window 크기가 조절된 제안하는 기법이 window 크기가 고정된 기존의 기법에 비하여 장애물과의 거리 측정에 유리함을 알 수 있었다. 장애물 유무를 판단하는 실험을 레이다, 레이저 센서를 사용하는 경우와 비교하여, 비전 센서를 통한 전방의 장애물 인식이 가장 신뢰도가 높다는 것을 알 수 있었다.

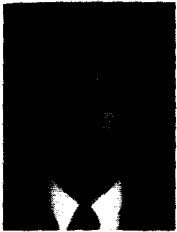
지능형 고안전 자동차에서 추돌 경보를 목적으로 할 때, 속도 및 거리의 정확성은 레이다, 레이저 센서를 사용하여 얻을 수 있으나, 차선 인식이 가능한 비전 센서의 활용이 없이는 경보 시스템이 완성될 수 없음을 보여 주었다. 이것은 미래형 자동차의 자율 주행뿐만 아니라 이동 로봇 등과 같이 CCD 영상을 사용하여 주어질 계적을 따라 가면서 장애물 회피를 할 필요성이 있는 시스템들에 적용 가능할 것이다.

참 고 문 헌

- [1] M. Hirano, "Development of Vehicle-Following Distance Warning System for Trucks and Buses," *IEEE-IEE Vehicle Navigation & Information Systems conference*, Ottawa-VNIS'93, 1993.
- [2] C.E. Smith, C.A. Richards and N.P. Papanikolopoulos, "Visual Tracking for Intelligent Vehicle-Highway Systems," *IEEE Transactions on Vehicular Technology* 45(4), pp. 744-759, 1996.
- [3] 김승남, 김채수, 박창호, 최장원, 두민수, 진화수, "Laser Radar를 이용한 Collision Warning System 개발", 한국 자동차 공학회 춘계 학술대회 논문집, pp. 167-172, 1996
- [4] 박범주, 한민홍, "무인차량의 도로주행 방법", 한국 자동제어학회 논문집, pp 773-778, 1991
- [5] 이주호, 양현승, "자율 운항 로봇의 실내 주행을 위한 복도의 투시구조 결정", 정보과학회 논문지 (B) 제22권 제12호, pp. 1681-1692, 1995
- [6] R. Jain, R. Kasturi, B.G. Schunck, *Machine Vision*, McGraw-Hill International Edition, 1995, pp. 66-69.
- [7] A.M. Waxman, J. LeMoigne, L.S. Davis, B.

- Srinivasan, T. Kushner, E. Liang and T. Siddalingaiah, "A Visual Navigation System for Autonomous Land Vehicles," *IEEE Trans. Robotics Automat.*, vol. RA-3, no. 2, pp. 124-141, 1987.
- [8] S.J. Dickinson and L.S. Davis, "A Flexible Tool for Prototyping ALV Road Following Algorithms," *IEEE Trans. Robotics & Auto.*, vol 6, no 2, pp. 232-242, 1990.
- [9] 정준익, 최성구, 노도환, "직선 Edge 추출에 의한 주행 방향 및 장애물 검출에 관한 연구", 한국 자동 제어 학술회의 논문집, Vol. 2, pp. 97-100, 1996
- [10] 한형석, 김명수, 이은경, 이석, "지능형 차량 충돌 경보 시스템의 경보 알고리즘 개발", 한국 자동 제어 학술회의 논문집, Vol. 1, pp. 656-659, 1997

저 자 소 개



崔勝郁(正會員)

1996년 2월 동아대학교 전자공학과 학사. 1998년 2월 부산대학교 대학원 전자공학과 석사. 1998년 3월 ~ 현재 부산대학교 대학원 전자공학과 박사과정. 주관심분야는 영상정보를 이용한 유효 정보 추출. 비전 시스템을

이용한 완성품 검사.



李章明(正會員)

第 34卷 S編 第 1號 參照
현 부산대학교 공과대학 전자공학과 부교수. 부산대학교 컴퓨터 및 정보통신 연구소 전자기술 부장