

선행차량 인식을 위한 DLI 기반의 영상처리 알고리즘

황희정<sup>1</sup>, 백광렬<sup>1</sup>, 이운근<sup>2</sup>  
 부산대학교 전자공학과<sup>1</sup>, 부산경상대학 디지털애니메이션과<sup>2</sup>

Image processing algorithm for preceding vehicle detection based on DLI

H. J. Hwang<sup>1</sup>, K. R. Baek<sup>1</sup>, U. K. Yi<sup>2</sup>  
 Pusan National University<sup>1</sup>, Busan Gyeongsang College<sup>2</sup>

**Abstract** - 본 논문은 차량 내에 설치된 두 대의 CCD 카메라를 이용하여 도로 영상으로부터 주행차선내에 있는 장애물을 인식하는 새로운 알고리즘을 제시한다. 제안된 알고리즘은 주행하는 차선과 관련이 있는 차선 정보만을 이용하여, 스테레오 영상에서 변이도를 추출할 수 있는 변이도 함수인 DLI(Disparity of lane-related information)를 정의하였다. DLI는 선행 차량과 같은 장애물은 주위보다 상대적으로 큰 에지값을 가진다는 특성을 이용하여, 주행차선 내에 있는 장애물의 유무를 검출하고 위치를 유추한다. 제안된 방법은 특징점의 탐색공간을 현저히 줄여 실시간 처리문제를 해결할 수 있는 장점을 가지고 있다. 본 논문에서는 DLI를 이용한 선행차량 인식기법의 성능을 검증하기 위하여 다양한 환경의 도로영상에 알고리즘을 적용하여 제안한 방법의 우수함을 확인하였다.

선행차량을 포함한 장애물 검출에 기반이 되는 DLI를 구축한다. DLI는 좌우 차선 중 한 차선을 중심으로 얻은 영상의 변이도 맵이다. DLI를 구축한 후, 이를 이용하여 장애물 인식을 하게 된다. 장애물 인식은 탐색 윈도우 구축, 블록 설정, 텍스처 추출의 순으로 진행된다.

그림 1은 본 논문에서 제안하는 전체 알고리즘의 구성을 나타낸다.

1. 서론

차세대 교통 시스템인 ITS (Intelligent Transport System)의 개발과 함께 자동차의 안전성과 편의성을 높이기 위한 새로운 기술 개발이 활발히 되고 있다. 이러한 기술은 운전자의 판단과 행동 없이 차량 스스로가 능동과 프로세싱 능력을 가지고 차량을 능동적으로 제어하여 예기치 못한 사고를 방지하는 등의 편의성을 높여주는 기술이다. 본 논문에서는 이러한 기술들을 이용하여 주행 중인 차차량 전방의 차선 및 차량을 신뢰성 있게 인식할 수 있는 스테레오 비전 기반의 영상처리 알고리즘에 대해 기술한다.

본 논문에서 제안하는 선행차량 인식 기법은 스테레오 비전에 기반을 하고 있다. 그동안 학계에서 스테레오 화상 방식을 이용한 3차원 복원에 관련된 연구가 활발히 진행되어 왔지만 자동차의 능동형 안전기술에 적용한 사례는 드물다. 본 논문에서 제안하는 DLI를 이용한 방법은 특징점의 탐색공간을 현저히 줄여 실시간 처리문제를 해결할 수 있는 장점을 가지고 있다.

본 논문에서는 DLI를 이용한 선행차량 인식기법의 성능을 검증하기 위하여 실제 도로에서 발생할 수 있는 다양한 환경의 도로영상에 알고리즘을 적용하였다. 그리고 실험 결과를 통하여 DLI 함수와 선행차량 인식 사이에 높은 상관성이 있음을 밝히고, 알고리즘의 유효성을 입증하였다.

2. 알고리즘의 개요

본 논문은 차량 내에 설치된 두 대의 CCD 카메라를 이용하여 도로 영상으로부터 차선과 장애물을 인식하는 알고리즘을 제시한다. 제안된 알고리즘에서 인식하려는 장애물은 주행 차선의 전방에 있는 물체를 뜻한다. 본 논문에서는 장애물 인식을 위해, 장애물의 방향과 위치는 급격히 변하지 않고 연속성을 갖는다는 가정을 한다.

본 논문에서 차선 인식과 장애물 인식을 위하여 사용하는 것은 영상의 에지 정보이며, 에지 정보는 영상 밝기의 미분값으로 크기와 방향 성분을 가지고 있다. 본 논문에서는 차선 인식을 위하여 에지의 방향 성분을 이용하고, 변이도 검출과 장애물 인식을 위하여 에지의 크기(Edge를 이용한다. 차선 정보는 참고문헌 [1]에 제시된 EDF(Edge distribution function)를 이용하여 얻는다. EDF는 에지의 방향각에 대한 히스토그램으로 차선방향의 주변에서 피크 값을 가진다. 차선 인식 완료 후, 본 논문에서는 주행차선 내에 존재하는

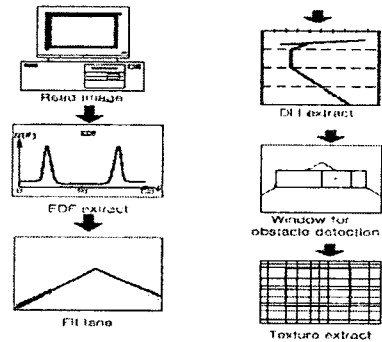


그림 1. DLI를 이용한 장애물 인식 알고리즘의 구성  
 Fig. 1. Organization of the algorithm.

3. DLI 검출 알고리즘

스테레오 영상의 변이도는 좌우 영상 전체의 정보를 이용하여 구하는 것이 일반적인 방법이다. 본 논문에서는 영상 전체의 정보를 사용하지 않고, 차선 인식에서 얻은 차선 관련 정보 가운데 한 차선의 정보만을 이용하여 변이도를 구한다. 본 논문에서는 영상의 차선 관련 정보의 일부분을 이용해서 얻은 변이도를 DLI (Disparity of lane-related information)라고 정의한다. DLI는 인식된 차선에서 한 차선의 정보만을 이용하여 얻기 때문에 변이도 검출을 위해 많은 양의 정보를 필요로 하지 않고, 영상 전체의 극히 일부분에 대한 정보만이 요구되어진다. DLI는 선행 차량과 같은 장애물이 주위보다 상대적으로 큰 에지값을 가진다는 특성을 이용해서 주행차선 내에 있는 장애물의 유무 검출뿐만 아니라, 장애물의 위치도 유추할 수 있다.

일반적으로 스테레오 정합을 수행하면 폐색 영역이 나타나게 된다. 스테레오 정합을 할 때, 좌우 영상 중 어떤 영상을 기준으로 하는지에 따라 각기 다른 폐색 영역을 가진다. 제시한 알고리즘에서는 폐색 영역이 일어나지 않는 부분을 중심으로 변이도를 구하기 위하여, 스테레오 영상의 좌측 영상을 중심으로 변이도를 구할 때에는 우측 차선을 중심 차선으로 정하고, 스테레오 영상의 우측 영상을 중심으로 변이도를 구할 때에는 좌측 차선을 중심 차선으로 정한다.

기준 영상을 정하고 기준 차선이 정해지면, DLI를 얻기 위하여 윈도우를 설정한다. 기준 차선을 중심으로 크기가 30x4 픽셀인 윈도우를 그림 2와 같이 설정한다. 그림 2에서 직선은 스테레오 영상의 우측 영상에 있는 차선이고, 점선은 스테레오 영상의 좌측 영상에 있는 차선이다. 스테레오 영상의 우측 영상을 기준 영상으로 정합을 할 때, 그림 2와 같이 실선과 일치하는 점선 영역을 찾기 위해서는 우측

영상에서 설정한 윈도우를 좌측 영상에서 일정 간격만큼 우측 방향으로 이동하여 적합을 이루는 좌측 영상에서의 위치를 구한다.

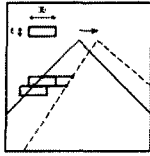


그림 2. DLI 알고리즘  
Fig. 2. DLI algorithm.

식 (1)은 좌우 영상의 에지 크기를 비교하는데 사용되는 절대 거리 (Absolute distance)이다.

$$C_n = \sum |L_{n,i} - R_i| \quad (1)$$

여기서  $C_n$ 은  $n$ 번째 윈도우에서의 절대 거리를 나타낸다. 우측 영상이 기준 영상일 경우,  $L_{n,i}$ 는 좌측 영상의  $n$ 번째 윈도우의  $i$ 번째에 있는 화소의 에지 크기이고,  $R_i$ 는 우측 영상에 있는 기준 차선에 썬어진 윈도우의  $i$ 번째에 있는 화소의 에지 크기이다.  $m$ 개의 윈도우 비교 값 중에서 가장 작은 값을 가지는  $C_n$ 을 찾아서 그때의  $n$ 값을 저장한다. 이  $n$ 값이 바로 그 좌표에서의 변이도이고, 식 (2)에 있는 DLI 값이 된다.

$$DLI = \text{Min}(C_n), \quad (1 \leq n \leq 80) \quad (2)$$

그림 3은 DLI 그래프의 영역 구분도로, DLI는 크게 네 영역으로 구분됨을 알 수 있다. DLI 그래프에서 수평축은 변이도를 나타내고, 수직축은 입력 영상의 수직축 좌표와 일치한다. DLI 그래프의 변이도축 위에는 변이도 간격이 20 픽셀이 되는 곳마다 눈금으로 표시가 되어 있다. 변이도는 식 (2)에서 구한 DLI 값이고, 수직축 좌표는 DLI에 사용된 입력 영상의 수직축 좌표이다. DLI는 일정한 기울기를 가지는 영역, 일정한 값을 유지하는 영역, 폐색영역, 소실점 영역으로 나누어진다. 변이도는 가까운 거리에 있는 물체일수록 큰 값을 가지고, 먼 거리에 있는 물체일수록 작은 값을 가진다.

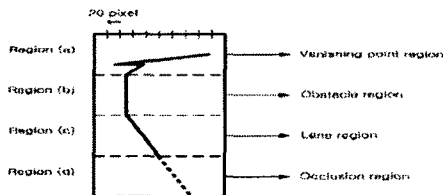


그림 3. DLI의 영역 구분도  
Fig. 3. Division of the DLI graph.

그림 3의 영역 (c)는 DLI 그래프에서 일정한 비율로 변이도가 커지는 영역으로 주행 차선이 있는 영역이다. 주행 차량과 가까운 거리에 있는 차선의 변이도는 크고, 주행 차량과 먼 거리에 있는 차선은 변이도가 작다. 주행하는 도로에 연속적으로 그려져 있는 차선은 거리에 따라 변이도가 일정하게 변한다.

그림 3의 영역 (b)는 DLI 그래프에서 변이도가 일정한 값을 유지하는 영역으로 주행 차선 내에 선행 차량과 같은 장애물이 있는 것을 뜻한다. 주행하는 차선에 선행 차량이 있는 경우, 입력 영상에서 이 선행 차량은 차의 뒷부분을 중심으로 영상에 나타나게 된다. 주행 차량에서 선행 차량의 뒷부분 내에 있는 영역까지의 거리는 같다. 그리고 선행 차량이 있는 곳에서는 에지 크기가 상대적으로 크게 나타나기 때문에 이 영역을 중심으로 DLI가 구해진다. 선행 차량의 뒷부분은 주행 차량과 일정한 거리에 있기 때문에 그 영역이 나타난 곳의 변이도는 같은 값을 가지게 되고, 그 결과 DLI는 수직축과 평행한 직선의 형태로 나타난다.

그림 3의 영역 (d)는 스테레오 영상의 폐색영역이 나타나

는 곳으로 정확한 변이도를 알 수 없는 영역이다. 차선이 있는 영역에 나타나는 1차 직선 형태의 변이도를 이용하여 그림 3의 영역 (d)에 있는 점선과 같이 폐색영역에서의 변이도를 유추할 수 있다.

그림 3의 영역 (a)는 소실점 부근의 영역으로 정확한 변이도 값을 얻을 수 없고, 신뢰성 있는 변이도를 구하기 힘든 부분으로 불안정한 변이도 값을 가진다. 소실점 이상의 영역은 주행차량과의 거리차가 큰 영역으로 본 논문이 다루는 차선과 장애물의 관심 범위를 벗어난다. 이 영역은 본 논문에서 제안한 알고리즘에도 영향을 미치지 않는다.

#### 4. 장애물 인식 알고리즘

##### 4.1 탐색 윈도우 설정

본 논문에서는 DLI 정보를 이용해서 주행 차선 내에 있는 장애물을 인식한다. DLI는 차선 내에 장애물이 있을 가능성이 큰 영역을 설정하는데 중요한 역할을 한다. DLI 그래프를 보면 변이도가 일정한 값을 유지하는 영역이 있는데, 본 논문에서는 이 영역을 물체가 있을 가능성이 큰 영역으로 인식한다. 그리고 변이도가 일정한 값을 유지하는 영역 내에 있는 차선과 이 영역이 만나는 곳을 중심으로 탐색 윈도우를 설정한다.

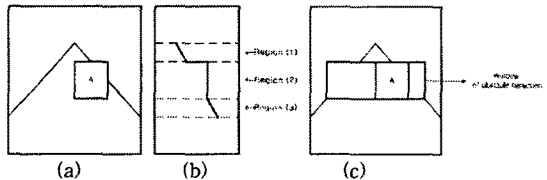


그림 4. 탐색 윈도우 설정

Fig. 4. Selecting of the window for obstacle detection.

그림 4 (a)에 있는 A는 주행 차선 전방에 있는 장애물이고, 그림 4 (b)는 이 영상의 DLI 그래프이다. 그림 4 (b)의 DLI 그래프를 보면, A가 있는 곳인 영역 (2)에서는 변이도 값이 일정하게 유지되고, 차선이 있는 영역 (1)과 영역 (3)에서의 변이도는 기울기가 일정한 일차 직선으로 나타나는 것을 알 수 있다. 탐색 윈도우는 장애물이 있을 확률이 높다고 판단되는 영역 (2)와 주행 차선이 만나는 곳을 중심으로 그림 4 (c)와 같이 설정 된다.

##### 4.2 블록 설정

본 논문에서는 장애물을 인식하기 위해서 탐색 윈도우를 중심으로 블록을 설정 한다. 블록을 설정하기 위해서는 수직 에지와 수평 에지가 필요한데, 본 논문에서는 소벨 연산자를 이용하였다. 식 (3)은 소벨 마스크를 이용하여 얻은 에지값으로,  $G_x$ 는 수직 마스크를 이용해서 얻은 에지값이고,  $G_y$ 는 수평 마스크를 이용해서 얻은 에지값이다.

$$G_x = |(a_{13} + 2a_{23} + a_{33}) - (a_{11} + 2a_{21} + a_{31})| \quad (3)$$

$$G_y = |(a_{11} + 2a_{12} + a_{13}) - (a_{31} + 2a_{32} + a_{33})|$$

그림 5에 있는 X, Y는 탐색 윈도우의 수평 길이와 수직 길이이고, N은 X, Y를 각각 등간격으로 나눌때의 기준 간격이 된다. 블록 설정을 위해서  $G_x$ 와  $G_y$ 를 이용하여 각 축에 대한 히스토그램을 구하고, N 간격에서 가장 큰 히스토그램을 가지는 곳의 좌표를 검색 한다.

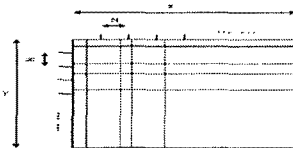


그림 5. 블록 설정

Fig. 5. Selecting of the block window for obstacle detection.

식 (4)에서  $His_{G_x}$ 는 수직축에 대한 히스토그램이고,  $His_{G_y}$ 는 수평축에 대한 히스토그램이다.  $C_x$ 는 수직축을 중심으로  $N$  간격에서 가장 큰 히스토그램의 좌표를 저장한 함수고,  $C_y$ 는 수평축을 중심으로  $N$  간격에서 가장 큰 히스토그램의 좌표를 저장한 함수이다.

$$C_x = MAX \left( \sum_{j=1}^N His_{G_x}(j) \right) \quad (4)$$

$$C_y = MAX \left( \sum_{i=1}^N His_{G_y}(i) \right)$$

위 식을 이용하면 일정 간격 내에서 가장 큰 히스토그램 값을 가지는 곳의 수평 좌표와 수직 좌표를 구할 수 있다. 블록은  $C_x$ 와  $C_y$ 에 저장된 좌표값을 이용해서 그림 5와 같이 설정된다.

#### 4.3 텍스처 추출

블록이 설정되면 탐색 윈도우 내에 물체가 있는지를 구분해야 한다. 본 알고리즘에서는, 탐색 윈도우 안에 물체가 있는지를 구분하기 위해서 블록 설정에서 구한 수평 예지  $G_y$ 를 이용한다.

$$T_w = E [ G_y ( Window ( X , Y ) ) ] \quad (5)$$

$$T_b = E [ G_y ( Block ( N , N ) ) ]$$

$$T_b > T_w \times Weight \quad (6)$$

식 (5)의  $T_w$ 는 탐색 윈도우 안의 수평 예지  $G_y$ 의 평균이고,  $T_b$ 는 탐색 윈도우 내에 있는 블록에 대한 수평 예지  $G_y$ 의 평균이다. 본 논문에서는 식 (6)을 만족하는 블록을 물체가 있을 가능성이 큰 영역으로 판단한다. 탐색 윈도우 내에 있는 각 블록마다 식 (6)이 만족되는지를 검색하고, 식 (6)을 만족하는 블록이 있으면 그 블록의 위치를 저장한다. 텍스처는 식 (6)을 만족하는 블록을 중심으로 추출한다. 그림 6에 있는 검게 채워져 있는 블록은 탐색 윈도우 내에서 물체가 있을 가능성이 큰 영역을 표시한 것이다.

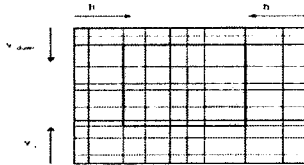


그림 6. 텍스처 추출

Fig. 6. Extraction of texture for obstacle detection.

탐색 윈도우에서 텍스처의 추출을 위해서 각 축마다 검게 채워진 블록의 수를 검사 한다. 수평축의 좌측 끝에서부터  $h_1$  방향으로 각 축마다 블록의 수를 구한다. 2개 이상의 블록이 있는 축이 검색되면 그 축의 좌표를 저장하고 검색을 마친다. 수평축의 우측 끝에서부터  $h_2$  방향으로 각 축마다 블록의 수를 구한다. 2개 이상의 블록이 있는 축이 검색되면 그 축의 좌표를 저장하고 검색을 마친다. 수직축은  $v_{up}$ 과  $v_{down}$  방향으로 수평축과 같은 방법으로 각각의 좌표를 구한다. 이렇게 얻은 4개의 좌표를 이용해서 사각 블록이 만들어지면 탐색 윈도우 내에 물체가 있는 것으로 판단하고, 그 블록을 중심으로 텍스처를 추출한다. 만약, 사각 블록이 형성되지 않으면 탐색 윈도우 내에 물체가 없는 것으로 판단하고 새로운 영상을 입력받는다.

#### 5. 실험 및 결과고찰

본 논문에서는 DLI를 이용한 선행차량 인식기법의 성능을 검증하기 위하여 실제 도로에서 발생할 수 있는 다양한 환경의 도로영상에 알고리즘을 적용하였다.

그림 7(a), (b)는 입력영상을 나타내며, 그림 7(c)는 DLI 그래프를 나타낸다. 그림 7의 (a), (b)에는 장애물 인식을 위한 탐색 윈도우 설정, 블록 설정, 텍스처 추출의 과정이

모두 나타나 있다. 그림 7 (c), (d), (e), (f)는 DLI를 이용한 장애물 인식 알고리즘의 수행 결과로, 주행 차선에 있는 선행 차량이 텍스처로 인식된 것을 확인할 수 있다.

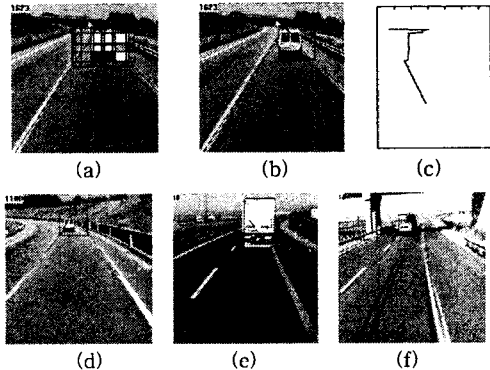


그림 7. 선행차량 인식

Fig. 7. Extraction of preceding vehicle.

#### 6. 결 론

지금까지 스테레오 영상을 이용한 많은 연구가 진행되고 있지만, 대응점 탐색 등에는 아직까지 계산 속도를 획기적으로 향상시킬 수 있는 알고리즘이 없고, 실시간 처리에도 치명적인 단점을 가지고 있다. 본 논문에서 제안한 알고리즘은 스테레오 영상에서 변이도를 구할때 DLI를 이용하여 특징점의 탐색 공간을 현저히 줄였다. 그 결과 DLI를 이용한 알고리즘은 메모리, 계산량 등을 향상시키고 실시간 처리 문제를 해결 할 수 있다. 게다가 제시한 알고리즘은 변이도를 구할 때 영상의 예지 크기 값을 비교하기 때문에 좌우 영상의 밝기 값을 비교하는 것보다 잡음이 강하고 더 정확한 절대 거리를 얻을 수 있다.

본 논문에서 도출된 결과는 논문에서 본질적으로 목표를 두고 있는 응용기술로서 ITS 분야의 차선이탈경보 및 방지 시스템 그리고, AGV (autonomous guidance vehicle) 등의 이동로봇 분야에 응용될 수 있으며, 향후 본 논문의 결과를 적용하기 위한 정보활용 방안에 대한 연구가 진행되어야 한다. 그리고, 제안된 DLI 기반의 장애물 인식 알고리즘은 도로환경의 영상처리에 한정된 것이 아니라 각종 패턴분류, 인식시스템 등에 직접적인 응용이 가능하며, 신뢰성을 요구하는 진단시스템 등 타 분야로의 확장 적용이 가능한 것이다.

#### [참 고 문 헌]

- [1] J. W. Lee, U. K. Yi, and K. R. Baek, "A Cumulative Distribution Function of Edge Direction for Road-Lane Detection," *IEICE Trans. Information and Systems*, vol. E84-D, no. 9, pp. 1206-1216, 2001.
- [2] C. J. Taylor, J. Malik, J. Weber, "A real-time approach to stereopsis and lane-finding," *Intelligent Vehicles Symposium, Proceedings of the 1996 IEEE*, pp. 207-212, 1996.
- [3] K. Saneyoshi, "Drive assist system using stereo image recognition," *Intelligent Vehicles Symposium, Proceedings of the 1996 IEEE*, pp. 230-235, 1996.
- [4] K. Saneyoshi, "3-D Image Recognition System by Means of Stereoscope Combined with Ordinary Image Processing," *Intelligent Vehicles '94 Symposium*, pp. 13-18, 1994.
- [5] K. Saneyoshi, K. Hanawa, K. Kise, Y. Sogawa, "3-D Image Recognition System For Drive Assist," *Intelligent Vehicles '93 Symposium*, pp. 60-65, 1993.
- [6] K. Hanawa and Y. Sogawa, "Development of Stereo Image Recognition System for ADA," *IVS2001*, pp. 177-182, 2001.