

**Vision - based Vehicle Detection Using HOG and OS Fuzzy-ELM**

윤창용\* · 이희진\*\*†  
Changyong Yoon\* and Heejin Lee\*\*†

\*수원과학대학교 전기과, \*\*한경대학교 전기전자제어공학과,  
\*Department of Electrical Engineering, Suwon Science College  
\*\*Department of Electrical, Electronic and Control Engineering, Hankyong National University

**요 약**

본 논문에서는 컴퓨터 비전 및 영상처리 기술을 접목하여 지능형 차량에 적용할 수 있는 실시간 차량 검출 알고리즘을 제안한다. 도로 환경의 빠르게 변화하는 배경과 차량의 다양성 때문에 차량의 실시간 검출은 부정확성 및 계산량 증가의 어려움을 가지고 있다. 본 논문은 기존 방법들의 이러한 문제점들을 해결하기 위하여 먼저, 복잡한 배경이 포함되어 있는 실시간 입력 영상으로부터 수직 에지 정보와 차량의 그림자 색정보를 사용하여 후보군을 검출한다. 다음으로, 검출된 후보군 영역들로부터 HOG 특징점을 추출한 후, 마지막으로 추출된 특징점들을 단일층 전방향 신경망 구조를 기반으로 하는 OS 퍼지-ELM을 사용하여 분류한다. 본 논문에서 제안된 방법을 사용하여 실험을 수행한 결과로써 기존의 ELM 및 OS-ELM 방법보다 계산량 및 정확성면에서 향상되었음을 보인다.

**키워드** : 퍼지 OS-ELM, 차량 검출, 단일층 전방향 신경망, HOG, 컴퓨터 비전

**Abstract**

This paper describes an algorithm for detecting vehicles detection in real time. The proposed algorithm has the technique based on computer vision and image processing. In real, complex environment such as one with road traffic, many algorithms have great difficulty such as low detection rate and increasing computational time due to complex backgrounds and rapid changes. To overcome this problem in this paper, the proposed algorithm consists of the following methods. First, to effectively separate the candidate regions, we use vertical and horizontal edge information, and shadow values from input image sequences. Second, we extract features by using HOG from the selected candidate regions. Finally, this paper uses the OS fuzzy-ELM based on SLFN to classify the extracted features. The experimental results show that the proposed method perform well for detecting vehicles and improves the accuracy and the computational time of detecting.

**Key Words** : ELM, Vehicle Detection, SLFN, HOG, Computer Vision

Received: Aug. 24, 2015  
Revised : Nov. 11, 2015  
Accepted: Nov. 24, 2015  
†Corresponding author  
lhjin@hknu.ac.kr

**1. 서 론**

본 연구는 한경대학교 2014년도 학술연구 조성비의 지원에 의한 것임(This work was supported by a research grant from Hankyong National University in the year of 2014)  
This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

최근 지능형 교통 시스템(ITS : Intelligent Transport System)에서 관심이 높아지면서 지능형 자동차에 대한 연구가 활발히 진행되고 있다. 과거의 자동차가 차량의 고유의 성능인 주행 속도와 연비 효율 향상 등에 치중되었던 반면에, 지능형 자동차는 보다 능동적이고 적극적인 안전성을 추구한다. 특히, 운전자의 안전성과 편의성을 향상시키기 위해 사전 경보 시스템(AWS : Advance Warning System) 및 적응 주행 제어 시스템(ACC : Adaptive Cruise Control) 등과 같은 분야들의 연구는 컴퓨터 비전 기술과 접목되어 진행되고 있다. 이는 인간의 여러 가지 감각기관 중에서 시각이 가장 많은 비중을 차지

하듯이, 지능형 자동차의 여러 가지 안전 시스템은 차량 주변 환경의 인식을 위하여 컴퓨터 비전기술을 핵심기술로써 사용하기 때문이다.

도로 위의 차량 검출은 다음과 같은 단계로 구성된다. 먼저, 차량에 설치된 카메라로부터 획득한 도로 영상에서 차량이 존재할 것으로 예상되는 후보군 영역을 검출한다. 이 단계는 영상 전체 화면에서 차량을 검출을 위한 검색을 수행하지 않고 후보군 영역에서만 차량을 검출함으로써 계산량을 단축할 수 있다는 장점을 가지고 있다[1].

그 다음, 차량을 다른 객체와 효과적으로 분류하기 위해 사용할 수 있는 차량의 특성을 대표하는 특징점을 추출한다. 영상에서 객체의 특성을 잘 표현하는 특징점을 추출하는 방법으로 HOG(Histogram Of Gradient), PCA(Principal Component Analysis), LOC(Local Orientation Coding), SIFT(Scale Invariant Feature Transform), 가버(Gabor)필터, Haar-like 변환, 웨이블릿 변환 등을 사용할 수 있다[2]. HOG는 기울기(gradient)의 방향성을 이용한 블록별 히스토그램을 구현한다. 이를 위해 물체의 형상 특징을 벡터형태로 저장하며 국소 영역에 대한 밝기의 분포 방향을 히스토그램으로 나타내어 계산하는 것이다[3]. 이는 근접한 픽셀의 분포를 조명의 영향을 적게 받으면서 국소영역에 기하학적인 시각 변화에 대하여 강인한 특성을 가지고 있다.

마지막으로, 차량의 특징점들을 분류하기 위한 단계로써 신경망, 베이스 분류기, SVM(Support Vector Machine), EM(Expectation Maximization)등과 같은 방법이 사용된다. 이와 같은 방법들은 학습과 분류에 기반을 둔 관심 객체의 인식 및 검증의 목적으로 활용된다.

기존의 전향 신경망 방식은 반복적으로 파라미터를 조정하면서 오차를 줄여나가기 위하여 기울기 하강법(gradient descent)을 기반으로 하는 학습을 수행하므로 학습 속도가 느리다. Huang [4]등에 의해 제안되어 최근에 많이 연구되는 ELM(Extreme Learning Machine)은 인식이 우수하면서도 기존 신경망의 단점인 느린 학습 속도를 개선하였다는 장점을 가지고 있다. 이들이 제안한 ELM은 방사 기저 함수(RBF: Radial Basis Function)를 가지는 단일층 전향 신경망을 기반으로 하며, 방사 기저 함수의 중심점과 영향 넓이를 사용하여 은닉층 출력 행렬을 만들고, 반복적인 학습 대신에 모어-페로스의 일반화된 역행렬(Moore-Penrose generalized inverse)에 의해 분석적으로 만들어진 출력 기중치를 사용하여 계산한다[5].

ELM과 같은 배치 학습(Batch Learning) 방법이 발표된 이후, 실제 환경에서는 학습을 위한 데이터 집합의 크기가 거대한 경우가 많았으며 이 경우 은닉층 출력 행렬의 크기 또한 커지므로 계산 시간 및 노력이 많이 소요된다. 따라서, 이를 해결하기 위해 학습을 위한 데이터를 덩어리 단위(chunk-by-chunk) 혹은 하나씩(one-by-one) 처리하고, 새로운 데이터가 입력될 때 마다 다시 학습할 필요가 없는 순차적인 학습(Sequential Learning) 알고리즘이 제안되었다[5]. 본 논문에서는 대표적인 순차적인 학습 방법인 온라인 ELM(OS-ELM: Online Sequential Extreme Learning Machine)과 온라인 퍼지-ELM(Online Fuzzy-Extreme Learning Machine)에 대하여 소개하고, 실시간 차량 검출을 위한 특징점 분류기에 적용한다.

본 논문은 다음과 같이 구성된다. 먼저, 2장에서는 본

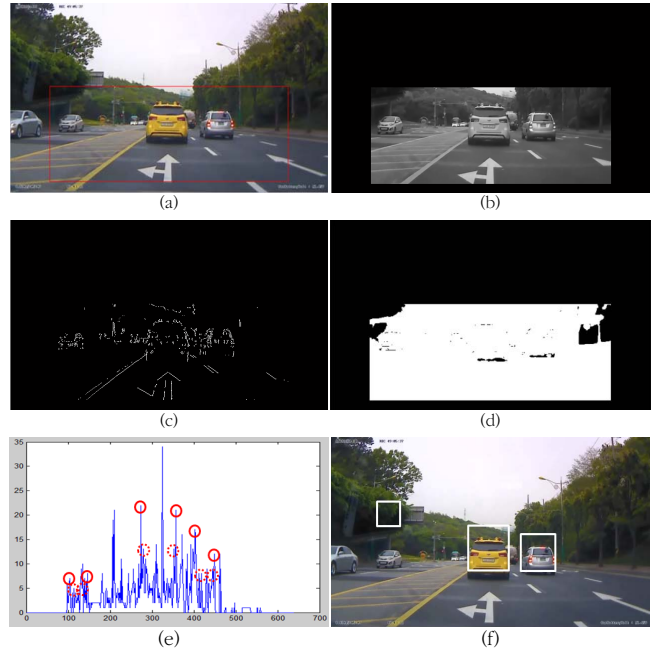


그림 1. 후보군 검출 단계  
(a) 입력 영상 및 관심 영역 설정 (b) 그레이 영상으로 변환 (c) 수직 에지 정보 검출 (d) 차량의 그림자 영역 추출 (e) 수직 에지 정보의 히스토그램 표현 (f) 차량 후보군 검출 결과

Fig. 1. the step of candidate regions extraction  
(a) Setting an area of interest from input images (b) Conversion to gray image (c) Detection of vertical edge information (d) Detection of car shadow information (e) Histogram representation from vertical edge information (f) the result of candidate regions extraction

논문에서 사용된 에지 정보와 차량 그림자 정보를 이용한 후보군 검출, HOG를 이용한 특징점 추출, 그리고 온라인 ELM을 기반으로 하는 퍼지-ELM 방법에 대하여 소개한다. 3장에서는 2장에서 언급된 방법을 사용하여 차량 검출 알고리즘에 대하여 제안하고 기술한다. 4장에서는 검출 성공률 및 계산시간에 대한 실험 및 분석결과를 보이고 그 내용을 설명한다. 5장에서는 결론 및 향후 연구방향에 대하여 기술한다.

## 2. 후보군 검출 및 학습 분류기

### 2.1 후보군 영역 설정

본 논문의 차량 후보군 설정은 그림 1과 같은 단계로 진행된다. 그림 1-(a)와 같이 입력된 영상을 차량이 주로 분포하는 도로 영역을 관심영역(사각형으로 표시된 영역)으로 선정하여 관심영역에서만 후보군 검출을 수행한다. 그림 1-(b)는 선정된 관심영역을 그레이 영상으로 변환시킨 결과이다. 그림 1-(c)는 소벨(Sobel) 에지 검출기를 사용하여 수직 에지를 검출한 결과이며, 그림 1-(d)는 그레이 색정보를 사용하여 차량의 그림자 영역을 추출한 결과이다. 그림 1-(e)는 그림 1-(c)의 수직 에지 정보의 픽셀수를 히스토그램 결과로 표현하였다. 실선 원은 차량 뒷면의 양쪽 에지로 판단되는 부분이며, 점선 원은 실선 원 다

음에 급격하게 감소하는 부분으로써 후보군 영역의 수직 크기를 판단할 때 사용한다. 그림 1-(f)는 차량의 그림자 정보와 히스토그램 결과를 조합하여 차량 후보군 영역을 검출한 결과이다. 후보군 영역의 수직 크기는 히스토그램 정보를 사용하며, 수평 크기는 그림자 영역의 위치 정보를 사용하여 계산한다.

**2.2 특징점 검출**

본 논문에서는 HOG를 사용하여 차량의 특징을 추출한다. 차량의 영역에 대한 밝기의 분포방향을 히스토그램으로 표현하여 특징벡터로 사용한다. 후보군 영역에 대한 기울기 성분과 방향성분을 다음과 같이 계산한다기.

$$\begin{aligned}
 m(i,j) &= \sqrt{H(i,j)^2 + V(i,j)^2} \\
 \theta(i,j) &= \tan^{-1} \frac{V(i,j)}{H(i,j)} \\
 H(i,j) &= f(i+1,j) - f(i-1,j) \\
 V(i,j) &= f(i,j+1) - f(i,j-1)
 \end{aligned}
 \tag{1}$$

$$\tag{2}$$

여기서,  $m(i,j)$ 는 위치  $(i, j)$ 의 기울기의 크기,  $\theta(i,j)$ 는 방향성분이며,  $f(i,j)$ 는 밝기값이다.

$8 \times 8$  픽셀영역으로 이루어진 셀(cell) 영역의 밝기 변화에 대한 방향성 히스토그램을 구한다.  $\theta(i,j)$ 는 기울기 크기의 방향성만을 고려하여 하나의 셀에 9개의 빈(bin)이라고 불리는 방향성 히스토그램이 존재하도록 구성한다.

**2.3 온라인-ELM(OS-ELM: Online Sequential ELM)**

온라인 ELM은 기존의 배치 ELM의 은닉층 출력 행렬  $H$ 로부터 유도할 수 있으며, 이는  $H\beta = T$ 의 해로 사용된다.

$$\begin{aligned}
 H^\dagger &= (H^T H)^{-1} H^T \\
 \hat{\beta} &= (H^T H)^{-1} H^T T
 \end{aligned}
 \tag{3}$$

여기서  $H \in R^{N \times \tilde{N}}$ ,  $N$ 은 데이터의 개수이며,  $\tilde{N}$ 은 은닉 뉴런의 수이다. 식 (3)은 재귀적 최소 자승법(RLS : Recursive Least Square)을 사용하여 순차적인 온라인-ELM 알고리즘으로 유도될 수 있다[6].

온라인 ELM은 조건을 먼저 선정한 후 다음의 2단계로 나누어 수행된다.

(조건) 노드 타입이 가산 (additive) 은닉형인지 혹은 RBF 은닉형인지를 결정한 후, 결정된 타입에 따라 활성화 함수  $g$  혹은 RBF 커널  $\phi$ 을 결정한다. 또한, 노드의 타입에 따라 은닉 뉴런의 수 혹은 RBF 커널의 수  $\tilde{N}$ 를 선정한다.

(단계1) 초기화 단계

학습 데이터 집합

$\mathfrak{N} = \{(x_i, t_i) | x_i \in R^n, t_i \in R^m, i = 1, \dots, \tilde{N}\}$ 으로부터, 초기 학습 데이터 집합  $\mathfrak{N}_0 = \{(x_i, t_i) | i = 1, \dots, N_0, N_0 \geq \tilde{N}\}$ 를 선정한다.

(a) 가산 은닉형인 경우는 입력 가중치  $w_i$ 와 바이어스  $b_i$

값을 할당하고, RBF 은닉형인 경우는 중심점  $\mu_i$ 과 영향 넓이  $\sigma_i$ 를 할당한다.

(b) 초기 은닉층 출력 행렬  $H_0$  ( $rank(H_0) = \tilde{N}$ )를 계산한다.

$$H_0 = \begin{Bmatrix} g(w_1 x_1 + b_1) & \dots & g(w_{\tilde{N}} x_1 + b_{\tilde{N}}) \\ \vdots & \dots & \vdots \\ g(w_1 x_{N_0} + b_1) & \dots & g(w_{\tilde{N}} x_{N_0} + b_{\tilde{N}}) \end{Bmatrix}_{N_0 \times \tilde{N}}
 \tag{4}$$

(c) 초기 출력 가중치  $\beta^{(0)} = M H_0^T T_0$ , 여기서

$$M_0 = (H_0^T H_0)^{-1}, T_0 = [t_1, \dots, t_{N_0}]^T \text{ 이 된다.}$$

(d)  $k = 0$ .

(단계2) 순차적인 학습 단계

$k+1$ 번째 새로운 관측 덩어리(chunk) 데이터가 입력되면 다음과 같이 표현한다.

$$\mathfrak{N}_{k+1} = \{(x_i, t_i)\}_{i = \left(\sum_{j=0}^k N_j\right) + 1}^{\left(\sum_{j=0}^{k+1} N_j\right)}
 \tag{5}$$

여기서  $N_{k+1}$ 은  $k+1$ 번째 관측 데이터의 수를 나타낸다.

(a)  $\mathfrak{N}_{k+1}$ 에 대한 부분 은닉층 출력 행렬  $H_{k+1}$ 를 계산한다.

$$H_{k+1} = \begin{Bmatrix} g(w_1 x_{(\sum_{j=0}^k N_j) + 1} + b_1) & \dots & g(w_{\tilde{N}} x_{(\sum_{j=0}^k N_j) + 1} + b_{\tilde{N}}) \\ \vdots & \dots & \vdots \\ g(w_1 x_{\sum_{j=0}^{k+1} N_j} + b_1) & \dots & g(w_{\tilde{N}} x_{\sum_{j=0}^{k+1} N_j} + b_{\tilde{N}}) \end{Bmatrix}_{N_k}
 \tag{6}$$

(b)  $T_{k+1} = [t_{(\sum_{j=0}^k N_j) + 1}, \dots, t_{(\sum_{j=0}^{k+1} N_j) + 1}]^T$ 이 된다.

(c) 출력 가중치를 다음과 같은 식으로 계산한다.

$$\begin{aligned}
 P_{k+1} &= P_k - P_k H_{k+1}^T (I + H_{k+1} P_k H_{k+1}^T)^{-1} H_{k+1} P_k \\
 \beta^{(k+1)} &= \beta^{(k)} + P_{k+1} H_{k+1}^T (T_{k+1} - H_{k+1} \beta^{(k)})
 \end{aligned}
 \tag{7}$$

(d)  $k = k+1$ .

**2.3 온라인 퍼지-ELM(OS Fuzzy-ELM)**

본 논문에서는 차량 검출을 위한 학습방법으로 방사기저함수(RBF)를 기반으로 하는 퍼지 신경망 구조의 OS 퍼지-ELM을 사용한다. 해당 퍼지 신경망은 TSK 퍼지 모델을 5개의 계층으로 구성되며, 이 계층 단계 중에 멤버십 함수 및 출력값이 만들어진다[4][8]. 본 방법은 신경망의 학습 기능과 퍼지 추론 시스템의 근사추론 기능의 장점을 합한 뉴로-퍼지 시스템인 것이다. 가우시안 방사기저함수를 가지는 신경망과 가우시안 멤버십 함수를 가지는 퍼지 추론 시스템은 같은 기능을 가지는 특징을 사용하며, OS-ELM에서 멤버십 함수를 다룰 수 있는 OS-퍼지 ELM이 개발되었다[9].

[RBF 신경망 기반 퍼지 신경망 구조]

- (a) 계층 1: 각 노드는 입력 변수를 나타내며, 입력 신호를 계층 2로 보낸다.
- (b) 계층 2: 각 노드는 입력 변수  $x_i$ 의 멤버십 값을 나타내며, 다음과 같은 멤버십 함수를 사용한다.

$$A_{ik}(x_i) = \exp\left(-\frac{(x_i - \mu_{ik})^2}{\sigma_k^2}\right), k = 1, 2, \dots, \tilde{N} \quad (8)$$

여기서,  $\tilde{N}$ 은 가우시안 함수의 수를 나타내며,  $\mu_{ik}$ 는  $i$  번째 입력에 대한  $k$  번째 가우시안 함수의 평균을 의미한다. 또한,  $\sigma_k$ 는 가우시안 함수의 편차를 나타낸다.

- (c) 계층 3:  $k$  번째 규칙의 발화도(firing strength, if-part)는 다음과 같은 식을 사용한다.

$$R_k(X) = \prod_{i=1}^n A_{ik}(x_i) = \exp\left(-\sum_{i=1}^n \frac{(x_i - \mu_{ik})^2}{\sigma_k^2}\right) \quad (9)$$

$$= \exp\left(-\frac{\|x_i - \mu_{ik}\|^2}{\sigma_k^2}\right)$$

- (d) 계층 4: 다음과 같은 식을 사용하여  $k$  번째 노드를 정규화시킨다.

$$\bar{R}_k = \frac{R_k(X)}{\sum_{k=1}^{\tilde{N}} R_k(X)} \quad (10)$$

- (e) 계층 5: 출력값은 각 정규화된 규칙의 출력에 가중치가 적용된 합으로 계산되며, 다음과 같은 식으로 계산할 수 있다.

$$Y = \frac{\sum_{k=1}^{\tilde{N}} R_k(X) w_k}{\sum_{k=1}^{\tilde{N}} R_k(X)} \quad (11)$$

여기서,  $Y = [y_1, y_2, \dots, y_m]^T$ ,  $w_k = [w_{k1}, w_{k2}, \dots, w_{km}]^T$  이다.

[OS 퍼지-ELM]

OS 퍼지-ELM은 조건을 먼저 선정한 후 다음의 2단계로 나누어 수행된다.

- (조건)  $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$ ,  $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$  인  $N$ 개의 이산 샘플  $(x_i, t_i)$  이 존재한다. RBF 신경망을 기반으로 하는 퍼지 신경망은 수학적으로 다음과 같이 모델링 할 수 있다.

$$H\beta = T \quad (12)$$

여기서,  $H$ 는 정규화된 노드의 출력 행렬을 의미한다.

먼저, 가우시안 멤버십 함수  $A$ 와 규칙수  $\tilde{N}$ 을 선정하며, 다음과 같은 신경망의 계층별 단계로 계산된다.

- (단계1) 초기화 단계

학습 데이터

집합  $\mathfrak{N} = \{(x_i, t_i) | x_i \in R^n, t_i \in R^m, i = 1, \dots, \tilde{N}\}$  으로부터,

초기 학습 데이터 집합

$\mathfrak{N}_0 = \{(x_i, t_i) | i = 1, \dots, N_0, N_0 \geq \tilde{N}\}$  를 선정한다.

- (a) 멤버십 함수의 파라미터  $(\mu_n, \sigma_n)$  들을 랜덤하게 할당한다.
- (b) TSK 모델을 위한 정규화 노드의 출력 행렬  $H_0$ 를 계산한다.

$$H_0 = \begin{Bmatrix} \bar{R}(\mu_1, x_1, \sigma_1) & \dots & \bar{R}(\mu_{\tilde{N}}, x_1, \sigma_{\tilde{N}}) \\ \vdots & \dots & \vdots \\ \bar{R}(\mu_1, x_{N_0}, \sigma_1) & \dots & \bar{R}(\mu_{\tilde{N}}, x_{N_0}, \sigma_{\tilde{N}}) \end{Bmatrix}_{N_0 \times \tilde{N}} \quad (13)$$

- (c) 초기 출력 가중치  $\beta^{(0)} = MH_0^T T_0$ , 여기서  $M_0 = (H_0^T H_0)^{-1}$ ,  $T_0 = [t_1, \dots, t_{N_0}]^T$  이 된다.
- (d)  $k = 0$ .

- (단계2) 순차적인 학습 단계

$k+1$  번째 새로운 관측 덩어리(chunk) 데이터가 입력되면 다음과 같이 표현한다.

$$\mathfrak{N}_{k+1} = \{(x_i, t_i)\}_{i = \left(\sum_{j=0}^k N_j\right) + 1}^{\left(\sum_{j=0}^{k+1} N_j\right)} \quad (14)$$

여기서  $N_{k+1}$ 은  $k+1$  번째 관측 데이터의 수를 나타낸다.

- (a) 1차 TSK 모델을 적용하여  $\mathfrak{N}_{k+1}$ 에 대한 부분 은닉층 출력 행렬  $H_{k+1}$ 를 계산한다.

$$H_{k+1} = \begin{Bmatrix} H_{k+1}^1 \\ H_{k+1}^2 \end{Bmatrix}_{\tilde{N}(n+1), N} \quad (15)$$

$$H_{k+1}^1 = \begin{Bmatrix} \bar{R}(\mu_1, x_{(\sum_{j=0}^k N_j) + 1}, \sigma_1) & \dots & \bar{R}(\mu_1, x_{(\sum_{j=0}^{k+1} N_j)}, \sigma_1) \\ \vdots & \dots & \vdots \\ \bar{R}(\mu_{\tilde{N}}, x_{(\sum_{j=0}^k N_j) + 1}, \sigma_{\tilde{N}}) & \dots & \bar{R}(\mu_{\tilde{N}}, x_{(\sum_{j=0}^{k+1} N_j)}, \sigma_{\tilde{N}}) \end{Bmatrix}^T$$

$$H_{k+1}^2 = \begin{Bmatrix} \bar{R}(\mu_1, x_{(\sum_{j=0}^k N_j) + 1}, \sigma_1) x_{(\sum_{j=0}^k N_j) + 1, 1} & \dots & \bar{R}(\mu_1, x_{(\sum_{j=0}^{k+1} N_j)}, \sigma_1) x_{\sum_{j=0}^{k+1} N_j, 1} \\ \vdots & \dots & \vdots \\ \bar{R}(\mu_{\tilde{N}}, x_{(\sum_{j=0}^k N_j) + 1}, \sigma_{\tilde{N}}) x_{(\sum_{j=0}^k N_j) + 1, n} & \dots & \bar{R}(\mu_{\tilde{N}}, x_{(\sum_{j=0}^{k+1} N_j)}, \sigma_{\tilde{N}}) x_{\sum_{j=0}^{k+1} N_j, n} \end{Bmatrix}^T$$

- (b)  $T_{k+1} = [t_{(\sum_{j=0}^k N_j) + 1}, \dots, t_{(\sum_{j=0}^{k+1} N_j)}]^T$  이 된다.

- (c) 출력 가중치를 다음과 같은 식으로 계산한다.

$$P_{k+1} = P_k - P_k H_{k+1}^T (I + H_{k+1} P_k H_{k+1}^T)^{-1} H_{k+1} P_k \quad (16)$$

$$\beta^{(k+1)} = \beta^{(k)} + P_{k+1} H_{k+1}^T (T_{k+1} - H_{k+1} \beta^{(k)})$$

- (d)  $k = k+1$ .

### 3. 제안된 전체적인 차량 검출 시스템

본 장에서는 2장에서 사용한 방법을 이용하여 실시간 차량 검출을 위한 알고리즘 과정에 대하여 기술한다. 차량 검출을 위한 알고리즘은 그림 1과 같이 크게 학습단계와 검출단계로 나뉘어진다.

학습단계에서는 긍정영상(Positive image)에 해당되는 차량 영상들과 부정영상(Negative image)에 해당되는 배경영상들을 취득하여 저장한다. 긍정영상은 영상에서 차량의 뒷부분에 해당되는 부분을 바퀴가 포함되도록 수작업으로 잘라서 저장한다.

저장된 영상은 그레이 영상으로 변환시킨 후 HOG 특징을 계산한다. 여기서, 블록당 특징수가 81개가 되도록 3×3 셀영역을 하나의 블록(block)으로 정한다(=9빈×9셀). 학습영상으로 취득된 영상을 72×72 픽셀이 되도록 스케일링하고 블록 9개가 포함되도록 하여 해당 영상의 특징 수가 729개인 벡터형태로 추출하여 사용한다(=81개×9블럭).

OS 퍼지-ELM을 사용하여 추출된 특징벡터들을 구분하기 위한 분류기를 제작한다. 본 논문에서 제안한 분류기는 RBF 신경망 기반이므로, 출력행렬  $H$ 를 계산하기 위하여 가우시안 멤버쉽 함수를 사용한다. 평균값은 [-1,1] 사이의 값을 각 단계마다 랜덤하게 선택하며, 편차값의 계산을 위한 분산값은 [0.01, 9.00] 사이의 값을 랜덤하게 선택한다.

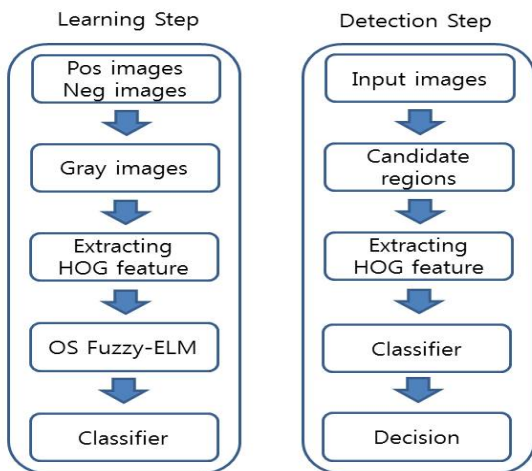


그림 2. 제안된 검출 알고리즘  
Fig. 2. the proposed algorithm for detecting vehicles

분류기의 제작은 초기 단계와 순차적인 학습 단계로 나뉘어진다. 초기 단계는 배치 학습의 형태로 퍼지 추론 시스템을 사용하며, 해당 단계에서 사용된 데이터는 초기 단계의 수행이 종료되자마자 삭제된다. 이 경우 학습 데이터의 수는 최소한 퍼지 규칙의 수와 동일하게 설정하도록 한다. 순차적인 학습 단계에서는 OS 퍼지-ELM은 하나씩 혹은 덩어리 단위로 학습시킬 수 있으며, 순차적인 덩어리의 사이즈는 고정시킬 수도 있고 시간에 따라 변경시켜가면서 입력할 수도 있다. 입력된 데이터들에 대한 학습 과정이 완료되어 분류기가 완성된 이후에는 모든 학

습 데이터는 향후 학습에서 사용하지 않으므로 삭제할 수 있다. 덩어리 데이터를 만들 때는 같은 종류의 차량을 한 덩어리로 만들어 제작하는 방식을 택했다. 이는 덩어리 데이터의 특징들의 유사성을 출력 행렬에 반영하기 위함이다.

검출단계에서는 실시간으로 입력되는 영상으로부터 학습단계에서 제작된 분류기를 사용하여 차량을 검출하는 단계이다. 먼저 실시간으로 입력된 영상을 그레이 영상으로 변환시킨 후 소벨 에지 검출기를 사용하여 추출된 수직 에지정보와, 색 정보를 사용하여 추출된 차량 그림자 영역 정보를 조합하여 차량 후보군 영역을 선정한다.

2장에서 언급된 차량 후보군 검출의 세부 과정은 다음과 같다.

- (1) 차량의 그림자로 판단된 영역( $g > 40$ )의 양쪽 끝 부분의 위치 정보를 수집한다.
- (2) 후보군 영역의 좌, 우 길이 = (양쪽 끝(실선 원)의 히스토그램 값 - 한 칸 안쪽(점선 원)의 히스토그램 값)
- (3) 좌, 우 길이 중에서 큰 값을 선택하여 설정
- (4) 후보군 영역의 위, 아래 길이 = 그림자 영역의 길이
- (5) 위, 아래 길이를 동일하게 설정

후보군 검출 후 학습단계와 동일한 방법으로 추출된 특징점을 OS 퍼지-ELM을 사용하여 분류한다. 차량 검출 분류는 차량인지 아닌지를 판단하는 이진 분류에 해당되므로 다음과 같은 식을 사용하였다.

$$f(x) = h(x)\beta = \text{sign}\left(h(x)\left(\frac{I}{C} + H^T H\right)^{-1} H^T H\right) \quad (17)$$

여기서,  $C$ 는  $[2^{-5}, 2^5]$  사이값을 사용하였으며,  $f(x)$  값이 양수이면 차량, 음수이면 배경으로 판단한다.

### 4. 실험 결과 및 분석

본 논문의 실험에서는 초기단계에서 2000개의 긍정영상(차량 영상)과 3000개의 부정영상(배경영상)을 사용하였다. 낮에 다양한 각도에서 취득한 차량영상을 수작업으로 자른 후 72×72크기로 스케일링하여 학습을 위하여 사용한다.

검출단계에서는 640×480의 해상도와 초당 30프레임의 속도를 가지는 도로영상이 저장된 AVI 동영상 파일을 사용하여 차량 검출 실험을 수행한다. 학습 및 검출을 위해서 사용된 컴퓨터 하드웨어 사양은 CPU 3.30 GHz, RAM 8G이며, MATLAB을 사용하여 알고리즘을 구현하였다.

#### 4.1 차량 검출 결과 영상

그림 3은 본 논문에서 제안한 OS 퍼지-ELM을 사용하여 실시간 차량 검출을 수행한 실험 결과 영상들의 예를 보여준다. 검출된 차량 영역을 사각형으로 표시하였으며, 주변 상황의 복잡한 정도에 따라 차량 검출 영역 표현에 미세한 차이가 존재하였



다. 후보군 검출을 차량 중심으로 정확히 수행할수록 차량 검출 성능에 영향을 미친다.

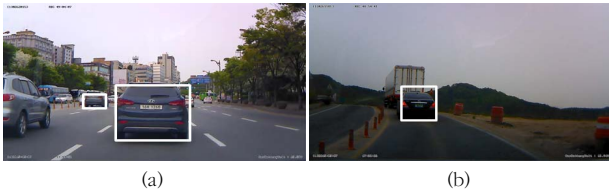


그림 3. 제안된 알고리즘을 사용한 검출 영상  
Fig. 3. the proposed algorithm for detecting vehicles

4.2 학습 데이터 수에 따른 시간 및 RMSE 비교

본 논문의 실험에서는 전체 학습 및 테스트 데이터 개수에 관계없이 긍정영상과 부정영상 수의 비율을 항상 4:6으로 조정하면서 학습을 수행하였다. 표 1과 표 2는 데이터의 수에 따른 분류기들의 성능비교 결과를 보여주고 있으며, ELM, OS-ELM 그리고 OS 퍼지-ELM 분류기들을 학습 데이터의 데이터 수가 6000개와 7000개일 때 학습시간과 RMSE, 그리고 테스트 데이터의 수가 2000개와 3000개일 때 RMSE를 비교하였다.

초기단계에서 학습 데이터 수를 학습모드에 따라 6000개, 7000개 까지 늘려가면서 실험을 수행하였으며, 노드의 수는 25개로 설정하였다. 학습데이터의 수가 6000개인 경우 긍정영상 2400개, 부정영상 3600개를 사용하였으며, 학습데이터의 수가 7000개인 경우 긍정영상 2800개, 부정영상 4200개를 사용하였다. 그리고, 테스트 데이터의 수는 2000개에서 3000개로 늘려가면서 실험을 수행하였다. 데이터의 수가 2000개인 경우 긍정영상 800개, 부정영상 1200개를 사용하였으며, 데이터의 수가 3000개인 경우, 긍정영상 1200개와 부정영상 1800개를 사용하였다.

먼저 학습시간을 살펴보면 ELM의 [배치모드]와 OS 방식의 [모드1]에서 학습시간이 많이 소요되는 것을 알 수 있다. [모드1]은 초기단계와 학습단계로 나뉘어 덩어리 사이즈(=1)별로 출력 행렬을 계산하는 방식을 수행하므로 [배치모드]보다 학습시간이 더 소요된다. 하지만 OS 방식의 [모드2]과 [모드10,30]은 초기단계 이후의 학습 데이터를 덩어리 사이즈별로 출력 행렬로 만들어 사용하며, 이전의 데이터는 학습할 때 사용하지 않으므로 학습 시간 측면에서 많은 장점이 있음을 확인할 수 있다.

표 1. 학습 데이터의 수가 6000개이고 테스트 데이터의 개수가 2000개일 때, 분류기들의 학습시간과 RMSE 비교

Table 1, RMSE Comparison of classifiers when the number of training data is 6000 and the number of testing data is 2000

Classifier	mode	Time (sec)	RMSE	
			Training	Testing
ELM	batch	25,3	0,0424	0,0444
OS-ELM	1	40,7	0,0427	0,0445
	20	13,2	0,0380	0,0400
	[10,30]	14,0	0,0395	0,0410
OS Fuzzy-ELM	1	42,0	0,0428	0,0448
	20	14,0	0,0382	0,0402
	[10,30]	14,2	0,0380	0,0390

표 2. 학습 데이터의 수가 7000개이고 테스트 데이터의 개수가 3000개일 때, 분류기들의 학습시간과 RMSE 비교

Table 2, RMSE Comparison of classifiers when the number of training data is 7000 and the number of testing data is 3000

Classifier	mode	Time (sec)	RMSE	
			Training	Testing
ELM	batch	35,0	0,0624	0,0644
OS-ELM	1	50,2	0,0627	0,0645
	20	24,0	0,0581	0,0600
	[10,30]	25,5	0,0597	0,0610
OS Fuzzy-ELM	1	51,0	0,0628	0,0648
	20	25,6	0,0571	0,0604
	[10,30]	27,0	0,0570	0,0582

표 3. 학습 데이터의 수가 7000개이고 테스트 데이터의 개수가 3000개일 때, 분류기들의 검출률 비교

Table 3, Detection rate comparison of classifiers when the number of training data is 7000 and the number of testing data is 3000.

Classifier	mode	Detection rate	
		Training	Testing
ELM	batch	95,2 %	95,0 %
OS-ELM	1	95,5 %	95,0 %
	20	96,8 %	96,3 %
	[10,30]	96,2 %	96,0 %
OS Fuzzy-ELM	1	95,7 %	95,2 %
	20	96,8 %	96,5 %
	[10,30]	97,2 %	97,0 %

RMSE의 결과를 살펴보면 ELM의 [배치모드]나 OS 방식들의 [모드1]에서는 성능의 큰 차이가 없으며, 데이터의 수(학습데이터 6000개->7000개, 테스트 데이터 2000개->3000개)가 증가할수록 ELM의 [배치모드]보다 OS 방식들의 [모드2]과 [모드10,30]에서 RMSE 결과가 더 좋으면서 성능의 차이가 더 커지는 것을 확인할 수 있다. 이는 OS 방식의 순차적인 학습단계에서 덩어리 데이터를 사용할 때 같은 종류(승용차, SUV, 대형차량 별)의 차량 정보만을 덩어리로 이용하여 출력 행렬을 제작함으로써, 현 단계에서 가장 최적의 출력 가중치를 계산할 수 있기 때문으로 보인다. 또한, 덩어리 데이터가 [모드10,30]과 같이 랜덤하게 발생하는 경우는 본 논문에서 제안한 OS 퍼지-ELM 방식의 RMSE 성능이 우수한 것을 확인할 수 있다. 이는 랜덤하게 발생하는 가우시안 멤버십 함수의 파라미터가 데이터의 수가 많아지면서 랜덤하게 입력되는 덩어리 데이터의 학습에 적합한 역할을 하는 것으로 판단된다.

표 3은 학습 데이터의 수가 7000개이고 테스트 데이터의 수가 3000개일 때, 각 분류기의 검출율을 비교한 결과이다. RMSE 실험결과와 마찬가지로 ELM의 [배치모드]와 OS 방식의 [모드1]은 검출율에서 큰 차이가 없으나, 데이터 개수(=학습데이터 7000개, 테스트 데이터 3000개)가 많아지고 덩어리 데이터가 랜덤하게 발생하는 경우는 본 논문에서 제안된 OS 퍼지-ELM 방

식이 우수한 검출율을 나타내는 것을 알 수 있다.

### 5. 결론 및 향후 연구

본 논문은 도로환경에서 안정적으로 실시간으로 차량을 검출하는 알고리즘을 제안하였다. 학습단계에서는 차량 뒷면 영상에 대해서 HOG 특징을 추출하여 특징 벡터를 생성하였다. 이 특징 벡터를 OS 퍼지-ELM을 사용하여 분류기를 제작하였다. 검출단계에서는 입력영상에 대해서 수직 에지 정보와 그림자 정보를 사용하여 후보군을 검출하였으며, 이 후보군 영역에 대하여 HOG 특징 벡터를 생성한 후 OS 퍼지-ELM에 입력하여 분류를 수행한 후, ELM 및 OS-ELM과의 성능을 실험을 통하여 비교하였다.

실험 결과로써, 데이터의 수가 증가하거나 학습데이터가 랜덤하게 발생하는 경우는 본 논문에서 제안한 OS 퍼지-ELM이 학습 시간 측면에서는 손실이 크지 않으면서 RMSE와 검출율에서 우수한 성능을 나타내는 것을 알 수 있었다.

향후, 더욱 성능이 우수한 실시간 차량 검출을 위해서는 낮과 밤 등 조명의 영향에 강인한 성질을 보이는 차량 검출 알고리즘을 전처리 부분에 적용한다면 더 좋은 성능을 나타낼 것으로 예측된다.

### References

[1] J. H. Yu, Y. J. Han, and S. H. Han, "Fast Vehicle Detection based on Haarlike and Vehicle Tracking using SURF Method," *Korean Society of Computer Information*, vol. 17, no. 1, pp. 71-80, 2012.

[2] M. S. Choi, H. J. Lee, M. T. Noh, and J. C. Sim "Vehicle Detection Scheme Based on a Boosting Classifier with Histogram of Oriented Gradient (HOG) Features and Image Segmentation," *Journal of KIISE*, vol. 16, no. 10, pp. 955-961, Oct. 2010.

[3] Y. H. Lee, J. Y. Ko, J. H. Suk, T. M. Roh, and J. C. Shim, "Pedestrian Recognition using Adaboost Algorithm based on Cascade Method by Curvature and HOG," *Journal of KIISE : Computing Practices and Letters*, vol. 16, no. 6, pp. 654-662, Jun. 2010.

[4] G. B. Huang, N. Y. Liang, H. J. Rong, P. Saratchandran, and N. Sundararajan, "On-Line Sequential Extreme Learning Machine," *The IASTED International Conference on CI*, Calgary, Canada, July 4-6, 2005.

[5] Z. Saad, M. K. Osman, Z. I. Zulkafli, S. Ishak, "Vehicle Recognition System Using Singular Value Decomposition (SVD) and Levenberg-Marquardt," *Computational Intelligence, Modelling and Simulation, 2009. CSSim '09. International Conference on*, pp. 187-191, 7-9 Sept. 2009.

[6] N. Y. Liang, G. B. Huang, P. Saratchandran, and N. Sundararajan, "A Fast and Accurate Online Sequential Learning Algorithm for Feedforward Networks," *IEEE Trans. on Neural Networks*, vol. 17, no. 6, pp. 1411-1423, Nov. 2006.

[7] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 886-893, 2005.

[8] J.-S. R. Jang and C.-T. Sun, "Functional equivalence between radial basis function networks and fuzzy inference systems," *IEEE Trans. Neural Netw.*, vol. 4, no. 1, pp. 156-159, Jan. 1993.

[9] H. J. Rong, G. B. Huang, N. Sundararajan and P. Saratchandran, "Online Sequential Fuzzy Extreme Learning Machine for Function Approximation and Classification Problems," *IEEE Trans. Systems, Man, And Cybernetics-Part B: Cybernetics*, vol. 39, no. 4, pp. 1067-1072, Aug. 2009.

[10] H. M. Eum, S. Y. Jang, H. J. Lee, M. Y. Park and C. Y. Yoon, "Human Detection and Fuzzy Temperature Control System for Energy Reduction of Cooling Device in Elevator," *Journal of Korean Institute of Intelligent Systems*, vol. 25, no. 2, pp. 147-154, April. 2015.

### 저 자 소 개



#### 윤창용(Changyong Yoon)

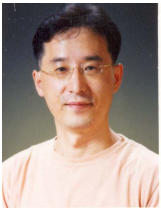
1997년 : 연세대학교 전자공학과 공학사  
 1999년 : 연세대학교 전자공학과 공학석사  
 2010년 : 연세대학교 전기전자공학과 공학박사

1999년~2005년 : LG전자 선임연구원  
 2005년~2006년 : LG노텔 선임연구원

2010년~2012년 : LG디스플레이책임연구원  
 2012년~현재 : 수원과학대학교 전기과 조교수  
 2013년~현재 : 한국지능시스템학회 이사

관심분야 : 컴퓨터비전, 얼굴인식, 지능제어, 퍼지이론, 지능형자동차

Phone : +82-31-350-2262  
 E-mail : cyoon@ssc.ac.kr



**이희진(Heejin Lee)**

1987년 : 연세대학교 전자공학과 학사

1989년 : 연세대학교 전자공학과 석사

1989년~1993년 : (주) 대우통신 종합연구소  
연구원

1993년~1995년 : (주) 에스원 종합연구소 연  
구원

1998년 : 연세대학교 전자공학과 박사

1998년~현재 : 국립한경대학교 전기전자제어공학과 교수

관심분야 : 로봇틱스, 강인제어, 지능제어, 전력제어 시스템

Phone : +82-31-670-5294

Fax : +82-31-670-5299

E-mail : lhjin@hknu.ac.kr