

딥러닝 기법을 이용한 차량 연료차단 주행의 감지법

고광호

평택대학교 스마트자동차학과 교수

Detection Method of Vehicle Fuel-cut Driving with Deep-learning Technique

Kwang-Ho Ko

Professor, Division of smart Automobile, Pyeongtaek University

요 약 차량의 변속기어가 체결된 주행 상태에서 가속페달을 방치하는 경우 연료차단 주행이 시작된다. 적극적인 연료차단 주행을 활용하면 차량 연비가 개선된다. 본 연구에서는 차량의 속도, 가속도, 도로구배를 입력데이터로 사용하여 연료차단 주행 여부를 예측할 수 있는 딥러닝 기법을 제안하였다. 약 12km 정도의 도로주행을 통해 측정된 9600개의 데이터에 은닉층 3~10개, 매개변수 10~20개의 딥러닝 연산법을 적용하여 연료차단 주행여부를 예측하였다. 연산 결과, 렐루함수를 활성화함수로 적용하고 은닉층 7개, 매개변수 10개인 경우 정확도 84.5% 수준으로 예측할 수 있었다. 입력데이터인 속도, 가속도, 도로구배의 변화율이 연료소모율 데이터의 변화율에 비해 큰 것이 오차의 원인으로 판단된다. 따라서 입력데이터 정규화 과정을 통해 정확도를 높일 수 있을 것으로 예상된다. 본 연구의 특징은 차량의 연료분사 인젝터나 OBD 데이터를 사용하지 않고 GPS 등에서 쉽게 측정할 수 있는 데이터에 딥러닝을 적용한 방식이다. 또한 연산량이 적어 본 연구에서 제안한 방식으로 친환경 경제운전에 적용하기 용이할 것으로 기대된다.

주제어 : 연료차단, 친환경운전, 딥러닝, 연비, GPS

Abstract The Fuel-cut driving is started when the acceleration pedal released with transmission gear engaged. Fuel economy of the vehicle improves by active fuel-cut driving. A deep-learning technique is proposed to predict fuel-cut driving with vehicle speed, acceleration and road gradient data in the study. It's 3~10 of hidden layers and 10~20 of variables and is applied to the 9600 data obtained in the test driving of a vehicle in the road of 12km. Its accuracy is about 84.5% with 10 variables, 7 hidden layers and Relu as activation function. Its error is regarded from the fact that the change rate of input data is higher than the rate of fuel consumption data. Therefore the accuracy can be better by the normalizing process of input data. It's unnecessary to get the signal of vehicle injector or OBD, and a deep-learning technique applied to the data to be got easily, like GPS. It can contribute to eco-drive for the computing time small.

Key Words : Fuel-cut, Eco-drive, Deep-learning, Fuel economy, GPS

1. 서론

자동차의 엔진속도가 약 1200rpm 이상이고, 변속기

어가 체결된(Drive) 상태에서 가속페달을 방치하면 연료차단(fuel-cut) 주행 상태가 된다[1]. 이는 대부분의 차량에 적용되어 있는 기능으로, 불필요한 연료소비를 방지하

*Corresponding Author : Kwang-Ho Ko(kwangho@ptu.ac.kr)

Received October 10, 2019

Accepted November 20, 2019

Revised November 11, 2019

Published November 28, 2019

기 위한 것이다. 교통신호 준수, 차간거리 유지 등을 위해 감속해야 하는 경우 주로 적용되는 기능으로 연비 향상에 기여하는 바가 크다. 연료차단 주행을 고속도로의 내리막 구간에 적용하는 경우 정속주행 대비 약 4% 상당의 연비향상을 얻을 수 있다[2,3]. 또한, 내리막 구간에서도 지속적으로 가속하는 경우 자칫 과속하게 되는데, 이를 예방할 수 있어 안전성을 확보할 수 있는 좋은 방법이기도 하다.

차량의 계기판에서 실시간 연비를 숫자나 그래프로 표시하는 경우 연료차단 주행 여부를 쉽게 확인할 수 있다. 숫자로 표시되는 경우 연료차단 주행 상태에서 99.9km/L로 도시되며, 그래프의 경우 주로 녹색의 그래프가 최대한 길어지도록 표시된다. 하지만, 이러한 차량 계기판의 연비 표시장치는 완성차업체에서 기본으로 제공하는 것이기 때문에 제3자의(3rd party) 업체에서 연료차단 기능을 효과적으로 도시하거나 활용하는 제품의 개발에는 상당한 어려움이 있다. 제3자의 업체의 제품은, 주로 인젝터 분사파형을 측정하거나, OBD(On Board Diagnostics) 단자의 연료분사 정보를 얻어 연비 및 연료차단 여부를 표시하는 방식으로, 이러한 제품의 경우 인젝터까지의 배선 작업이나 OBD 단자에 별도의 장치를 장착해야 하는 번거로움이 있다[4]. OBD 단자는 주로 차량 정비를 위해 대부분의 차량에 적용되어 있는 표준 입력력 단자이다[5].

이에 본 연구에서는 GPS 데이터(고도, 속도, 가속도)에 딥러닝 기법을 적용하여 연료차단 여부를 판정하는 방법을 제안하고자 한다. 도로 주행 실험을 통해, 10Hz 수준으로 측정된 도로 및 차량 데이터에 딥러닝 기법을 적용하여 연료차단 주행 여부를 분류하는 방식이다. 도로 구배(road gradient), 차속 및 가속도를 입력 데이터로, 연료차단 여부를 레이블(label)로 설정하여 다양한 구성의 딥러닝 기법을 적용하여 그 결과를 비교, 검토 하였다.

본 연구에서 제안된 방식은 추후 스마트폰의 GPS에도 동일하게 적용할 수 있기 때문에, 차량과의 배선 연결, OBD 데이터 없이 간단하게 연료차단 여부를 감지할 수 있다. 이 기법을 활용하여, 연료차단 주행 여부를 운전자에게 효과적인 방식으로(그래프나 소리 등) 알려주고, 주행이 종료되었을 때 이번 주행 중에 적용된 연료차단 횟수와 이를 통한 연비 향상 정도, 이전의 주행에서의 연비와 비교한 결과 등을 운전자에게 알려주는 경우 친환경 경제운전(Eco-drive)에 큰 도움이 될 것이다[6].

2. 실험방법 및 결과

실험도로는 도로구배가 다소 큰 산악지역의 국도이다. 약 12km 정도의 구간을 주행하면서 10Hz의 해상도로 도로구배, 차량속도, 가속도, 연료소모량 등을 측정하였다. 실험차량은 H사의 S모델로 2013년식 RV 차종으로, 공인연비 14.4km/h, 배기량 2000cc의 디젤엔진을 사용한다.

Table 1. Test vehicle specification

Spec.	Data
Vehicle type	RV
Model year	2013MY
Total weight	2245kg
Height	1680mm
Width	1880mm
Engine volume	1995cc
Certified fuel economy	14.4km/L
Fuel type	Diesel

Table 1에 실험 차량의 제원을, Table 2에 도로 주행을 통해 얻은 실험 데이터를 정리하였다. 해당 도로를 평균 속도 48km/h 정도로 주행했을 때 평균 연비가 12.8km/L 수준이고, 최소가속도 $-4.89m/s^2$, 최대가속도 $6.85m/s^2$, 도로의 평균경사각도 0.75도, 최대경사각도 8.6도, 연료소모량은 0~80.6mcc/s 정도임을 알 수 있다.

도로구배, 속도, 가속도, 연료소모량 등은 영국의 레이서로직스(Racelogics)사의 VBOX라는 장비를 사용하였다. Fig. 1의 VBOX는 모터스포츠 분야에서 주로 사용되는 장비로 자체의 GPS에서 도로구배, 속도, 가속도를 측정하고, 차량의 OBD 단자와 연결하여 연료소모량을 측정한다[7]. VBOX의 측정 해상도는 10Hz 수준이고, 관련 사양은 Table 3에 정리하였다. 속도는 0.1km/h, 각도 0.1도, 위치 3m 수준의 정밀도를 나타낸다. 이렇게 측정된 데이터를 가공하여 속도, 가속도, 도로구배 등을 계산한 후 딥러닝의 입력 데이터로 사용한다.

Table 2. Test driving data

Data	Result
Driving Distance	12.76km
Driving fuel economy	12.85km/L
Speed (average/max.)	48.1 / 97.4 km/h
Road gradient (average/max.)	0.75 / 8.6 deg
Acceleration (min./max.)	-4.89 / 6.85 m/s ²
Fuel consumption (min./max.)	0.0 / 80.6 mcc/s

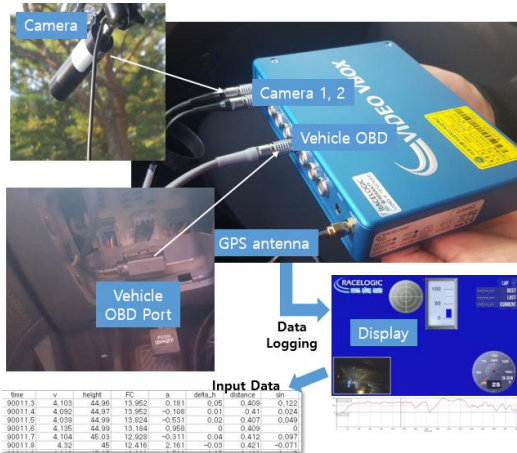


Fig. 1. Data logger(VBOX)

Table 3. Specification of data logger(VBOX)

Spec.	Accuracy	Resolution	Unit
Heading	0.1 °	0.01 °	degree
Position	3m(X, Y), 10m(Z)		meter
Distance	0.05%	1cm	cm
Velocity	0.1km/h	0.01km/h	km/h
Acceleration	1%	0.01G	G

Fig. 2에 주행 후 약 400초 부근의 측정 데이터를 도시하였다. 그래프에서 각 데이터가 서로 겹치지 않도록 스케일을 조정하여 도시하였고, 단위는 그래프에 표시하였다. 주행 개시 후 399.5~400.5초 사이에 연료소모량이 제로로 떨어지는 연료차단 주행이 지속되었음을 알 수 있다. 이 구간에서 연료차단 상태가 유지되는데도 불구하고 속도, 가속도, 도로구배는 그 크기의 변화가 상당함을 알 수 있다. 즉, 입력데이터의 변화율이 큰 조건에서 결과 데이터가 일정하게 유지되는 특징이 있는 것을 알 수 있다. 이는 추후 딥러닝의 예측 결과에 나타나는 주요

오차의 원인으로 판단된다.

또한 Fig. 2에서 연료소모량 데이터가 연속적으로 변화한다는 것을 알 수 있다. 즉 공회전 상태를 유지하기 위한 최소한의 연료를 분사하고 있다가 연료차단 조건에서 불연속적으로 제로가 되는 것이 아니고 부드럽게 변화하면서 제로로 떨어지고, 다시 연료분사가 재개될 때에도 연속적으로 연료소모량이 증가하여 공회전을 유지하는 정도의 연료소모량 상태가 되는 특징을 보임을 알 수 있다. 일정 수준의 최소 분사량을 유지하다가 제로로 떨어지는 경우 연료차단 주행 조건은 확실히 구분이 될 것인데, 실험 데이터와 같이 그 변화가 연속적이고 상당히 부드럽기 때문에 이를 뚜렷하게 구분하는 것도 상당히 어려워지는 것으로 판단된다.

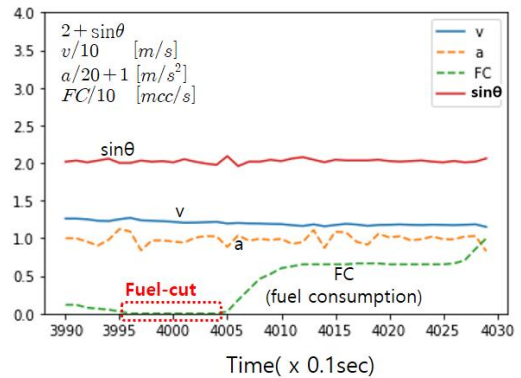


Fig. 2. Measured sample data from road driving

3. 계산방법 및 결과

3.1 딥러닝 연산 구조 및 순서

딥러닝에서는 입력데이터에 매개변수(가중치와 편향)를 적용하여 연산한 결과와, 미리 정해진 레이블(label)이 일치하도록 하는 과정을 반복 수행한다. 이때 연산 결과와 레이블의 차이가 오차인데, 이 오차를 최소화하는 매개변수를 경사하강법(gradient method)으로 결정하는 것이 핵심이다[8,9]. 경사하강법은 매개변수에 대한 오차의 미분값을 기준으로 매개변수를 증감시켜 매개변수를 최적화하는 방식이다. 매개변수를 적용하는 단계를 여러 개로 나누는 경우 각 단계가 하나의 은닉층(hidden layer)이 된다[10]. 본 연구에서는 3~5개의 은닉층을 적용하여 그 결과를 비교하였다. 또한 각 은닉층 사이에 적용되는 활성화함수(activation function)를 렐루(Relu)

함수와 시그모이드(Sigmoid)함수로 적용시켜 그 결과를 비교하였다[11].

딥러닝에서 주요한 인자(parameter)로는 매개변수의 증감 정도를 결정하는 학습율(learning rate), 1회에 적용하는 학습 데이터의 개수에 해당하는 미니배치(mini batch)의 크기 등이 중요한데, 본 연구에서는 학습율 0.01, 미니배치 크기를 100으로 지정하여 계산을 수행하였다. 전체 학습 데이터가 약 10,000개 수준이므로 미니배치의 크기인 100개를 적용하면 1회 학습에 100회씩의 배치 계산이 수행되며, 이를 다시 500회 반복하여 총 연산횟수(epoch)는 50,000회가 된다.

Table 4. Number of variables for each layer

Layer	No. of variables	Data
Output	1	Fuel consumption
Softmax+CEE ¹⁾	10x2 ²⁾	Matrix of Weights & Bias
Layer5	10x10	
Activation function	Relu / Sigmoid	
Layer4	10x10	
Activation function	Relu / Sigmoid	
Layer3	10x10	
Activation function	Relu / Sigmoid	
Layer2	10x10	
Activation function	Relu / Sigmoid	
Layer1	3x10	
Input	3	<i>v, a, sinθ</i>

¹⁾ CEE : Cross Entropy Error

²⁾ Label number = 2 (One-hot encoding)

매개변수의 개수도 중요한 요소 중의 하나인데, 매개변수의 개수가 많을수록 대체적으로 정확도는 증가하고 계산량은 늘어난다[12]. 본 연구에서는 각 층에서 10개 혹은 20개의 데이터를 출력하도록 고정하여 그 결과를 비교하였다. Table 4에 10개의 출력 데이터 조건에 해당하는 각 층의 매개변수의 형태를 정리하였는데, 이는 일반적인 이미지 분류에 적용하는 딥러닝 구조와 동일하다. 입력 층에서 속도, 가속도, 도로구배의 3개 데이터가 입력되면 중간 은닉층에서 10개의 데이터가 출력되도록 매개변수 행렬의 구조를 정하였고, 최종 결과로는 하나의 데이터를 출력하여 연료소모량을 예측하도록 딥러닝 연산구조를 구성하였다. 또한 각 층의 연산결과에 적용되는 활성화함수로 렐루 혹은 시그모이드를 선택하여 결과를 비교하였다. 20개씩 출력하는 경우에는 최초 입력 및 최종 출력 데이터는 앞의 경우와 같이 각각 3과 1로 유지

되고 나머지 개수를 20개로 적용한다.

Table 4에 정리된 입력데이터, 은닉층 및 출력층, 오차(loss) 등에 해당하는 딥러닝 연산 수식은 다음과 같다.

$$Input\ data = X(v, a, \sin\theta) \tag{1}$$

$$H_i = AF(X \cdot W_i + B_i) \tag{2}$$

$$Output = Softmax(H_{last}) \tag{3}$$

$$L = CEE = -Label \times \ln(Output) \tag{4}$$

$$G_W = \frac{\partial L}{\partial W} \tag{5}$$

$$G_B = \frac{\partial L}{\partial B} \tag{6}$$

$$W_{next} = W - \eta \times G_W \tag{7}$$

$$B_{next} = B - \eta \times G_B \tag{8}$$

여기에서,

H_i Result of *i*-th hidden layer

AF Activation Function(*Relu* or *Sigmoid*)

W_i Weight coefficient

B_i Bias Coefficient

L Loss or error from output and label

η Learning rate

G_W Gradient of loss for weight coefficient

G_B Gradient of loss for bias coefficient

$$Sigmoid(x) = \frac{1}{1 + e^{-x}}$$

$$Relu(x) = Max(0, x)$$

$$Softmax(x) = \frac{\exp(x_j)}{\sum \exp(x_j)}$$

식 (5)와 식 (6)의 G_W, G_B 에 해당하는 값들이 경사하강법에서 사용하는 기울기(gradient)이다. 오차를 최소화하는 매개변수 (W_i, B_i)를 이 기울기로 계산하며, 이 때 적용되는 변화의 크기는 학습율(η)에 반영되어 식 (7)과 식 (8)의 계산이 진행된다.

또한 본 연구에서는 연료소모량이 1.0mcc/s 미만에 해당하는 경우 연료차단 주행으로 간주하여 학습을 진행하였다. 이 조건에 레이블 제로(0)를 부여하여 딥러닝 기법으로 분류하였다. 일반적인 딥러닝 기법이 적용되는 이미지 분류 작업에서처럼 최종 결과에 레이블을 부여해야 하는데 이를 1.0mcc/s를 기준으로 한 것이다[13]. 즉, 레이블이 제로인 경우 최대 1.0mcc/s 정도의 연료소모

량이 존재할 수 있으나 이를 무시하고 연료차단 주행으로 판정하였다. 이는 연료소모량이 연료차단 상태로 바로 떨어지는 것이 아니고 점진적으로 감소하기 때문이다 [14]. Fig. 2의 그래프에서 400.5초 이후 연료소모량이 연속적으로 증가하면서 연료차단 구간에서 벗어나는 것을 확인할 수 있다.

3.2 딥러닝 연산 결과

위에서 설명한 방식으로 딥러닝 연산구조를 구성한 후 우선은 은닉층의 최대 개수가 5인 경우까지만 일단 계산하여 그 경향을 살펴보았다. 계산한 결과를 Table 5에 정리하였는데, 매개변수의 개수와 은닉층의 개수 및 활성화함수의 종류에 따라 조금씩 정확도가 달라지나 약 80% 수준으로 연료차단 상태를 감지할 수 있음을 알 수 있다. 0.1초 간격으로 측정된 연료소모량 실험데이터에서 연료차단에 해당하는 데이터는 1863개인데, 이 중의 80% 수준인 1490개 정도를 해당 시점에서 연료차단으로 예측하는 경우 정확도가 80% 수준임을 의미한다.

매개변수 10개, 활성화함수 Relu, 은닉층 개수가 5인 경우 정확도가 가장 높았고, 매개변수 10개, 활성화함수 Sigmoid, 은닉층 개수가 3인 경우 정확도가 가장 낮았다. 매개변수가 10개 수준인 경우 은닉층 개수가 증가할수록 다소 정확도가 올라가는데 이는 렐루함수를 적용한 경우 뚜렷하게 나타난다.

매개변수가 20개 수준인 경우에도 렐루함수를 적용하는 경우에는 은닉층의 개수에 비례하여 그 정확도가 증가함을 알 수 있다. 시그모이드함수의 경우 전반적으로 은닉층의 개수, 매개변수의 개수에 비례하여 그 정확도가 증가하기는 하나 렐루함수에 비해 그 경향은 다소 약한 것을 알 수 있다.

Table 5. Fuel-cut detection result with deep-learning

No. of variables	Activation function	No. of hidden layers	Accuracy of Fuel-cut detection
10	Relu	3	80.1%
		4	82.3%
		5	83.9%
	Sigmoid	3	72.4%
		4	72.7%
		5	72.7%
20	Relu	3	75.2%
		4	79.2%
		5	80.0%
	Sigmoid	3	78.4%
		4	83.4%
		5	78.6%

Table 6. Fuel-cut detection result with deep-learning (for the number of hidden layers more than 5)

No. of variables	Activation function	No. of hidden layers	Accuracy of Fuel-cut detection
10	Relu	3	80.1%
		4	82.3%
		5	83.9%
		6	84.2%
		7	84.5%
		8	84.2%
		9	84.1%
		10	84.0%

이상의 계산 결과에서 매개변수 10개, 렐루함수를 적용한 경우 은닉층의 개수에 비례하여 정확도가 증가함을 알 수 있었다. 이에 은닉층의 개수를 증가시키는 경우 연료차단 예측 정확도가 계속해서 증가하는지를 확인하였다. 그 결과를 Table 6에 정리하였는데, 은닉층의 개수가 7인 경우 84.5% 까지 증가하지만 그 이상의 경우에 대해서는 정확도가 미미하지만 감소함을 알 수 있었다. 은닉층의 개수가 5인 경우와 7인 경우 그 결과가 각각 83.9%, 84.5% 수준으로 약 84% 정도의 정확도를 나타낸다고 할 수 있다.

연료차단 주행에 대한 예측에 있어 활성화함수는 렐루함수가 유리하고 은닉층의 개수와 매개변수의 개수는 각각 7개, 10개 수준일 때 그 결과가 우수함을 알 수 있다. 실험에서 얻은 400초 부근의 연료차단 주행 그래프에 딥러닝으로 예측한 결과의 일부를 Fig. 3에 도시하였다. 측정된 연료소모량(FC Measured)은 지속적으로 연료차단 상태를 유지하는데 반해 딥러닝 예측 결과(FC Predicted)는 연료차단 상태에서 벗어나는 것을 알 수 있다. 이는 본 연구의 딥러닝 연산과정에서 사용된 입력데이터인 속도, 가속도, 도로구배의 변화율과 레이블에 해당하는 연료소모량의 변화율 차이에 의해 발생하는 것으로 판단된다.

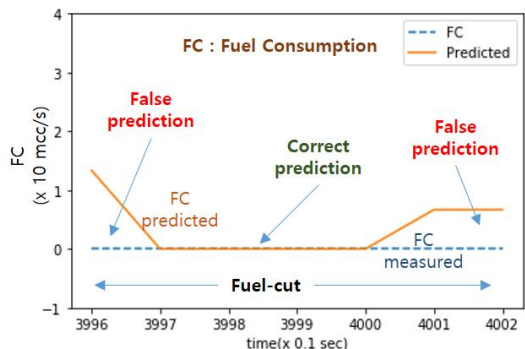


Fig. 3. Deep-learning prediction result sample

Fig. 3에서 속도, 가속도, 도로구배 등의 변화율이 상당히 큰데 반해 연료소모량 데이터는 변화 없이 제로를 유지하는 것을 통해 확인할 수 있다. 자동차 엔진의 안정적인 제어를 위해 연료소모량 제어는 상당히 안정적인 반면 연료소모량 예측을 위해 사용된 입력데이터인 속도, 가속도, 도로구배 등은 가감없이 그대로 적용되었고, 따라서 입력데이터의 변화율이 지나치게 크게 작용하면서 예측 결과에 오차가 발생하는 것으로 판단된다. 이는 추후 입력데이터 정규화(normalization) 과정에서 그 변화율을 적절히 감소시킬 수 있는 기법을 적용하여 개선할 수 있을 것으로 예상된다[15].

4. 결론

본 연구에서 일반적인 이미지 분류에 사용되는 딥러닝 기법을 자동차 연료차단 여부의 판단에 적용하여도 상당한 정확도로 예측할 수 있음을 알 수 있었다.

주행거리 12km, 평균 주행속도 48km/h 정도에 해당하는 실험 데이터를 사용하였다. 약 16분 정도에 해당하는 데이터로 10Hz 수준으로 측정하였기 때문에 전체 데이터 세트의 개수는 약 9600개 수준이다. 이 데이터에 딥러닝 기법을 적용하였는데, 황철화함수로 렐루와 시그모이드를 사용하였다. 매개변수를 10개와 20개, 은닉층의 개수를 3~10개로 변경하면서 예측 정확도를 평가하였다. 연료소모량 1.0mcc/s 이하의 경우 연료차단 주행상태로 설정하고 이에 대해 레이블 0을 부여하여 분류하는 방식을 적용하였다.

연산 결과 매개변수 10개, 렐루함수, 은닉층 개수 7인 경우 약 84.5% 수준으로 연료차단 주행을 예측할 수 있었다. 매개변수와 은닉층의 개수에 비례하여 예측 정확도가 증가하지만 이는 렐루함수를 적용한 경우 좀더 명확하게 비례하였다.

본 연구에서 제안한 방식은 차량의 인젝터 혹은 OBD 신호를 사용하지 않고 GPS에서 간단히 측정할 수 있는 신호만을 입력데이터로 하여 연료차단 여부를 정확도 84.5% 수준으로 예측할 수 있는 방법이다. 이 기법은 연비를 향상시킬 수 있는 다양한 친환경 경제운전 장치 개발에 기여할 것으로 기대된다.

REFERENCES

- [1] M. Setiyo & S. Munahar. (2017). AFR and fuel cut-off modeling of LPG-fueled engine based on engine, transmission, and brake system using fuzzy logic controller (FLC). *Journal of Mechatronics, Electrical Power and Vehicular Technology*, 8(1), 50-59. DOI : 10.14203/j.mev.2017.v8.50-59.
- [2] K. H. Ko & S. C. Choi. (2012). A study on the improvement of vehicle fuel economy by fuel-cut driving. *JKAIS*, 13(2), 498-503. UCI : G704-001653.2012.13.2.019
- [3] S. C. Choi., K. H. Ko. & I. S. Jeung. (2013). Optimal fuel-cut driving method for better fuel economy. *International Journal of Automotive Technology*, 14(2), 183-187. DOI : 10.1007/s12239-013-0020-4
- [4] K. H. Ko.. (2010). The Change Rate of Fuel Consumption for Different IRI of Paved Roads. *Int. J. Highw. Eng.*, 12(1), 55-59. UCI : G704-001414.2010.12.1.009
- [5] J. G. Yun. & J. Y. You. (2019). An Experimental Study on the Measurement of Fuel Consumption Using Fuel Flowmeter and CAN DATA. *KSMT*, 21(1), 63-68. DOI : 10.17958/ksmt.21.1.201902.63
- [6] G. H. Lee., S. J. Lim. & S. J. O. (2017). Comparing the Effects of Visual and Visual-auditory Feedback on Eco-driving and Driving Workload. *J. Korea Inst. Intell. Transp. Syst.*, 16(3), 120-131. DOI : 10.12815/kits.2017.16.3.120
- [7] J. C. Park., J. H. Kim., W. T. O., J. H. Choi. & J. J. Park. (2017). Reliability Evaluation of EDR Data Using PC-Crash & Vbox. *Transactions of KSAE*, 25(3), 317-325. DOI : 10.7467/KSAE.2017.25.3.317
- [8] S. H. Lee., I. S. Kim. & M. Robert. (2002). Model Reference Adaptive Control Using Non-Euclidean Gradient Descent. *IJAS*, 4(4), 330-340. UCI : G704-000903.2002.4.4.010
- [9] R. Ehsan. & E. Abbas. (2018). A Finite-time Adaptive Fuzzy Terminal Sliding Mode Control for Uncertain Nonlinear Systems. *IJAS*, 16(4), 1938-1950. DOI : 10.1007/s12555-017-0552-x
- [10] W. J. Lee., J. W. Lee. & S. R. Park. (2018). Instance Weighting Domain Adaptation Using Distance Kernel. *IE&MS*, 17(2), 334-340. DOI : 10.7232/iems.2018.17.2.334
- [11] S. H. Rou. & J. B. Yun. (2017). The Effect of regularization and identity mapping on the performance of activation functions. *JKAIS*, 18(10), 75-80. DOI : 10.5762/KAIS.2017.18.10.75
- [12] S. G. Inn. & J. H. Park. (2018). Inferring Method of Height and Weight from Smartphone Sensor Data Using Deep Learning and Clustering. *KTCP*, 24(9), 456-467. DOI : 10.5626/KTCP.2018.24.9.456
- [13] J. K. Sung. & Y. K. Kim. (2017). Deep learning-based

product image classification system and its usability evaluation for the O2O shopping mall platform. *JIIBC*, 17(3), 227-234

DOI : 10.7236/JIIBC.2017.17.3.227

- [14] C. H. Peng, I. C. Yeh & L. C. Lien. (2009). Modeling strength of high-performance concrete using genetic operation trees with pruning techniques. *CAC*, 6(3), 203-223

UCI : G704-SER000008569.2009.6.3.005

- [15] M. G. Ji., J. C. Chun. & N. G. Kim. (2018). An Improved Image Classification Using Batch Normalization and CNN. *Journal of Internet and Services*, 19(3), 35-42

고 광 호(Kwang-Ho Hong) [장학원]



- 1991년 2월 : 서울대학교 항공공학과(공학사)
- 1993년 2월 : 서울대학교 항공공학과(공학석사)
- 2012년 8월 : 서울대학교 기계자동차공학부(공학박사)
- 1998년 3월 ~ 2018년 2월 : 아주자

동차대학 자동차학부 교수

- 2018년 3월 ~ 현재 : 평택대학교 스마트자동차학과 부교수
- 관심분야 : 차량제어, 친환경운전, 차량동역학
- E-Mail : kwangho@ptu.ac.kr