



LSTM 기반의 차선 정보와 조향 각을 활용한 LKAS 요각속도 예측 모델

김 현 우 · 박 상 현*

전자부품연구원 모빌리티플랫폼센터

LSTM Based LKAS Yaw Rate Prediction Model Using Lane Information and Steering Angle

Hyunwoo Kim · Sanghyun Park*

Mobility Platform Research Center, Korea Electronics Technology Institute, 22 Daewangpangyo-ro, 712beon-gil, Bundang-gu, Seongnam-si, Gyeonggi 13488, Korea

(Received 2 February 2018 / Revised 11 February 2018 / Accepted 19 February 2018)

Abstract : This paper proposes the LSTM(Long short-term memory) model, which offers the advantage of considering data sequence, for estimating the yaw rate value based on the dynamic motion of the vehicle in the camera sensor as represented by the steering angle data. In addition, the use of data table in the learning data resolves the time lag problems caused by the different output cycles of the sensors. For learning, the input value was selected from the image data and output from the camera sensor to the coefficient value of the road model and the vehicle steering angle. The target value was selected as the yaw rate value. We have completed the yaw rate forecasting model by learning Many to One LSTM predictive models connected with the LSTM and Fully-Connected-Layer(FC) series. The results were confirmed by test-data after completion.

Key words : Lane keeping assistance system(차선 유지 시스템), Yaw rate prediction(요각속도 예측), Long short-term memory(장 단기 순환신경망), Steering control(조향 제어)

Nomenclature

$\dot{\psi}$: yaw rate, rad/s	y_t	: memory block output
$\hat{\psi}$: prediction yaw rate, rad/s	h_t	: the value passed to the memory block for 't+1'
δ	: steering angle, rad	b_y	: RNN output bias
c_0	: lateral offset	W_{hh}	: RNN h_{t-1} weight
c_1	: heading angle	W_{xh}	: RNN input data weight
c_2	: curvature	W_{yh}	: RNN output weight
c_3	: curvature rate	W_f	: LSTM input x_t weight of f_t
x_t	: input data	W_c	: LSTM input x_t weight of \tilde{C}_t
		W_i	: LSTM input x_t weight of i_t

*Corresponding author, E-mail: shpark@keti.re.kr

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.

- W_o : LSTM input x_t weight of o_t
- U_f : LSTM input h_{t-1} weight of f_t
- U_c : LSTM input h_{t-1} weight of \tilde{C}_t
- U_i : LSTM input h_{t-1} weight of i_t
- U_o : LSTM input h_{t-1} weight of o_t
- b_f : LSTM f_t bias
- b_c : LSTM \tilde{C}_t bias
- b_i : LSTM i_t bias
- b_o : LSTM o_t bias
- C_t : LSTM cell-state at the 't' time

Subscripts

- LKAS : lane keeping assistance system
- IMU : inertia measurement unit sensor
- RNN : recurrent neural network
- LSTM : long short-term memory
- MSE : mean square error
- CNN : convolution neural network
- RMSE : root mean square error

1. 서론

오늘날, 자동차는 다양한 안전 및 편의 시스템들이 개발되고 있다. 그 중 차선 유지 시스템(Lane Keeping Assistance System, LKAS)은 차량이 차선을 유지하도록 보조해 주는 시스템으로 운전자의 의도가 아닌 차선이탈이 발생할 경우, 운전자에게 위험을 알리고, 차선을 이탈하지 않도록 조향 장치를 제어하는 시스템이다.^{1,2)}

이러한 차선 유지 시스템은 카메라 센서로부터 도로정보를 취득하고, 차량 동특성 센서로부터 차량의 동특성을 파악하여^{3,4)} 횡 방향 제어기에서 차량 조향각^{5,6)} 또는 조향 휠 토크를 생성⁷⁻⁹⁾해 차선을 유지한다. 하지만 정확한 제어를 위해서는 고성능의 관성 측정 장치 센서가 필요하다. 이와 같은 방법은 비용 문제로 인해 실제 차량에 장착이 어려움이 있다. 또한, 센서의 주기가 달라 시간 지연 문제¹⁰⁾가 발생한다.

따라서 본 논문에서는 딥 러닝 모델 중 하나인 순환신경망(Recurrent Neural Network, RNN)¹¹⁾ 모델의

변형 모델인 장 단기 순환신경망(Long Short-Term Memory, LSTM)¹²⁻¹⁵⁾를 사용하여 센서 주기가 서로 다른 카메라 센서와 차량 조향각만을 통해 고성능의 관성 측정 장치 센서를 대체할 수 있는 강인한 요각속도 예측 기법을 제안한다. 먼저 장 단기 순환신경망 모델을 학습하기 위해 주행 데이터를 취득한다. 취득한 데이터에서 카메라 센서 데이터로부터 추출한 도로 방정식의 도로 모델의 계수 값¹⁶⁾과 차량의 조향각을 입력 값으로 선정한다. 목표값은 관성 측정 장치 센서(IMU sensor)에서 측정된 요각속도 값으로 선정하여 학습시킨다. 학습이 완료된 이후에는 관성 측정 장치 센서 없이 차량 내부의 조향각 센서와 카메라 센서만으로 요각속도 예측값을 얻을 수 있다. 무엇보다 장 단기 순환신경망은 데이터의 시간 정보를 포함하여 학습하기 때문에 차량의 과거로부터 현재까지 움직임을 고려한 요각속도를 예측하여 차선 유지 시스템에서 요구되는 강인한 요각속도 예측 모델을 설계할 수 있다는 장점이 있다.

본 논문에서 먼저 차량의 요각속도를 예측하기 위해 텐서플로우(Tensorflow)를 활용하여 여러 개의 입력을 받아 하나의 출력을 주는(Many to one) 장 단기 순환신경망 예측 모델을 구현하고, 차량을 주행하며 얻은 데이터를 7:3의 비율로 나누어 각각 학습과 시험(Test)을 진행하여 요각속도 예측 모델을 완성시킨다. 마지막으로 성능을 확인하기 위해 예측된 요각속도 값과 실제 차에 장착된 관성 측정 장치 센서에서 측정한 요각속도 값을 비교하여 성능을 확인한다.

본 논문의 구성은 다음과 같다. 제2장에서는 순환신경망과 장 단기 순환신경망의 기본적인 원리와 도로 모델을 나타낸다. 3장에서는 요각속도 예측 모델을 위한 모델 구성과 학습방법을 설명한다. 4장에서는 데이터에 대한 설명과 학습 및 시험을 진행하여 시뮬레이션 결과를 확인하였다. 5장에서는 본 연구의 결론과 향후 연구 방향을 제시한다.

2. 관련 이론

2.1 순환신경망

순환신경망의 모델을 나타내면 다음 Fig. 1과 같

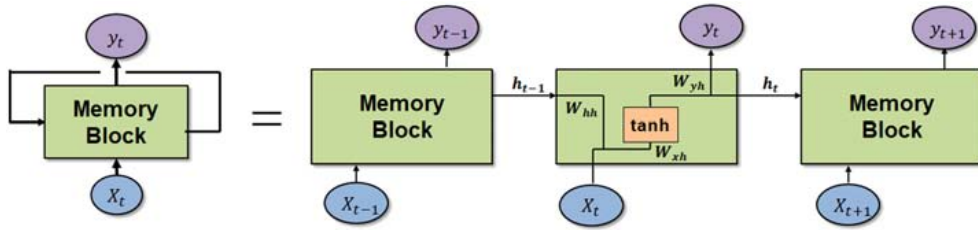


Fig. 1 RNN basic structure

다. Fig. 1에서 보이 것과 같이 순환신경망은 메모리 블록들을 서로 연결하여 계층을 쌓아 만든 모델이다. 각 입력 메모리 블록에는 특정 시점의 데이터 정보를 입력한다. 입력받은 메모리 블록은 연결된 메모리 블록으로 데이터 정보를 전달한다. 이처럼 순환신경망은 데이터를 주고받는 순환형식으로 되어 있다. 그러므로 합성곱 신경회로망(Convolution Neural Network, CNN)과 다르게 데이터의 시간 정보까지도 고려하여 학습하는 모델임을 알 수 있다.¹³⁾

위 순환신경망 모델을 나타낸 Fig. 1을 식으로 정리하면 다음 식 (1)과 같다.

$$\begin{aligned} h_t &= \tanh(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \\ y_t &= W_{yh}h_t + b_y \end{aligned} \quad (1)$$

위 식에서 보면 활성화 함수인 Tanh 함수를 통해 데이터가 전달되기 때문에 시계열(Sequence)이 길어질수록 데이터의 손실이 발생한다. 이것을 기울기 소실 문제(Banishing gradient problem)에 의한 데이터의 장기 의존성 문제(The problem of long-term dependency)라고 한다. 이 때문에 본 연구에서는 데이터의 장기 의존성을 문제를 해결하기 위해 제안된 순환신경망의 변형 모델인 장 단기 순환신경망을 사용한다.¹³⁻¹⁶⁾

2.2 장 단기 순환신경망

장 단기 순환신경망의 구조는 위 Fig. 1의 순환신경망 메모리 블록에서 메모리 블록 상태(Cell-state, C_t)가 추가된다. 이 메모리 블록 상태는 3개의 게이트 층으로 구분하여 구성된다. 그리고 이 3개의 게이트 층의 입력 노드에 시간 단위로 들어오는 데이터를 입력, 저장, 출력할 수 있도록 제어한다. 이 셀 상태의 제어 때문에 데이터의 장기 의존성 문제를 해결할 수 있다. 이 메모리 블록 상태를 포함한 장

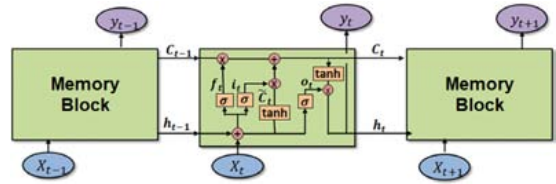


Fig. 2 LSTM basic structure

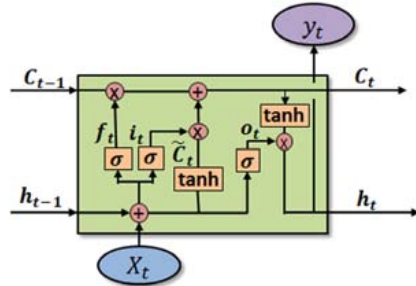


Fig. 3 LSTM memory block

단기 순환신경망의 구조를 보면 다음 Fig. 2와 같다.

위 장 단기 순환신경망을 나타내는 Fig. 2에서 중요한 메모리 블록 상태를 포함한 메모리 블록을 확인하면 다음 Fig. 3과 같다.

위 Fig. 3의 메모리 블록은 3단계로 나뉘어져 있다. 첫 단계는 이전 정보인 C_{t-1} 의 기억 유무를 결정하는 단계이다. 이 단계에서는 h_{t-1} 과 x_t 를 입력받아 시그모이드를 통과시켜 f_t 를 출력한다. 이 출력 값이 0이면 이전 상태의 정보인 C_{t-1} 을 기억하지 않고, 1이라면 C_{t-1} 을 온전히 기억하게 된다. 이러한 이유로 이 단계에서 구성된 층을 잊어버리는 계층(Forget-gate-layer)이라고 부르기도 한다. 이것을 수식으로 나타내면 다음 식 (2)와 같다.

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (2)$$

두 번째 단계는 이전 정보인 C_{t-1} 을 전달 받아 현

재 정보인 C_t 로 갱신하는 단계이다. 갱신 과정은 두 부분으로 나뉜다. 첫 번째 부분은 첫 단계와 같은 역할로써 시그모이드 층으로 이루어져 있고, 이전 정보와 현재 정보의 기억 유무를 결정하는 i_t 를 출력한다. 두 번째 부분은 Tanh 계층으로 현재 정보를 입력 받아 현재 상태인 C_t 에 더해줄 후보 \tilde{C}_t 를 만드는 부분이다. 이렇게 계산 되어진 두 부분의 결과 값을 합쳐 새로운 현재 메모리 블럭 상태 C_t 로 갱신시킨다. 식으로 나타내면 다음 식 (3)과 같다.

$$\begin{aligned} \tilde{C}_t &= \tanh(W_c x_t + U_c h_{t-1} + b_c) \\ i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i) \\ C_t &= f_t \circ C_{t-1} + i_t \circ \tilde{C}_t \end{aligned} \quad (3)$$

여기서 \circ 은 아다마르 곱(Hadamard product)¹⁷⁾을 나타낸다. 위 식 (3)을 보면 첫 단계에서 구한 f_t 를 이전 상태(C_{t-1})에 곱한 값과 i_t 를 현재 상태(C_t)에 곱한 값의 합을 나타내고 있다. 이것을 직관적으로 보면 이전 상태와 현재 상태의 기억유무를 정하는 값(f_t, i_t)을 통해 이전과 현재 상태의 기억유무를 결정하는 것을 알 수 있다. 마지막 단계는 결과 출력을 결정하는 단계이다. 이 출력 값은 필터처리 된 셀 상태를 기반으로 한다. 첫 번째로 앞 두 번째 단계에서 구한 현재 상태 C_t 을 Tanh 함수를 통과시켜 -1과 1의 사이 값을 갖도록 한다. 두 번째로 이 출력 값을 이전 상태 C_{t-1} 와 현재 셀에 입력받은 값 x_t 를 받아 시그모이드를 취해준 출력 값과 곱해준다. 결과적으로 설계자가 원하는 출력 값만 선택할 수 있게 되며, 다음 식 (4)와 같다.

$$\begin{aligned} o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o) \\ h_t &= o_t \circ \tanh(C_t) \end{aligned} \quad (4)$$

이러한 과정으로 모든 입력에 대해 모두 기억하는 것이 아니라, 선택적으로 기억하여 재귀적으로 학습하는 모델임을 알 수 있다. 그러므로 순환신경망의 문제점인 데이터 장기 의존성 문제(The problem of long-term dependency)^{15,16)}를 해결할 수 있다.

2.3 퍼셉트론

퍼셉트론(Perceptron)은 단층과 다층으로 구분할 수 있다. 단층은 입력층과 출력층으로 이루어져 있고,

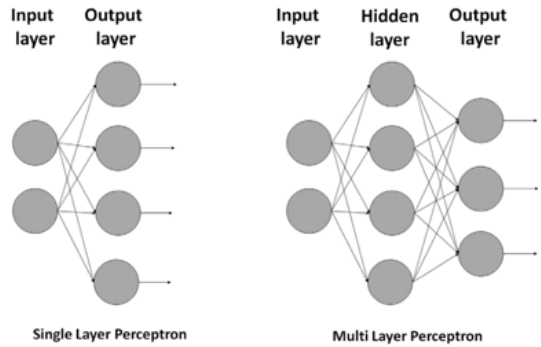


Fig. 4 Perceptron structure

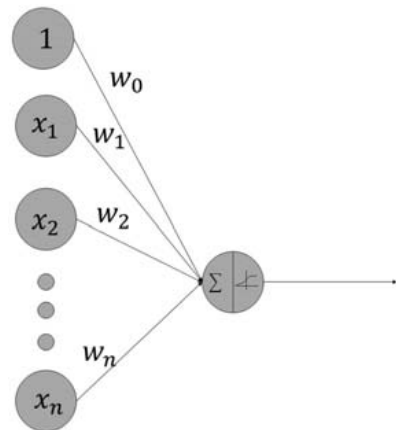


Fig. 5 Perceptron internal structure

다층은 입력층, 은닉층, 출력층으로 이루어져 있다. 퍼셉트론의 구조를 나타내면 다음 Fig. 4와 같다.

위 Fig. 4의 퍼셉트론의 구조에서 각 계층을 구성하는 것을 노드라고 부르며 선형결합과 활성화 함수 기능을 가지고 있다. 또한 노드별로 연결하는 선은 가중치를 가지고 있다. 이것을 그림으로 나타내면 다음과 같다.

위 Fig. 5를 보면 입력으로 받은 $x_1, x_2 \dots x_n, 1$ 의 입력 값은 각각 가중치가 곱해져서 노드로 전달된다. 전달된 값은 모두 더해서 활성화 함수를 거쳐 다음 노드로 전달한다. 이것을 식으로 나타내면 다음과 같다.

$$a = \sum_{i=0}^n w_i x_i \quad (5)$$

$$o = \sigma(a)$$

이러한 구조를 가진 네트워크를 퍼셉트론(Per-

ception)이라고 부르며 딥러닝에서 완전 접속망(Fully connected layer)라고 불리고 있다.

2.4 도로모델

차선 유지 시스템에서는 카메라 센서로부터 얻는 차선 정보가 가장 중요하다. 이 차선 정보를 얻기 위해 차량과 차선의 상대적인 운동을 도식화 한 도로 방정식 $f_L(s)$ 을 사용하여 차선정보 데이터를 얻는다. 도로 방정식을 종 방향 거리 s 의 3차 다항식으로 표현 하였을 때 각 계수 값들이 의미하는 바는 다음과 같다. c_0 (lateral offset), c_1 (heading angle), c_2 (curvature), c_3 (curvature rate)이다.¹⁶⁾ 식으로 나타내면 다음 식 (6)과 같다.

$$f_L(s) = c_0 + c_1s + c_2s^2 + c_3s^3 \quad (6)$$

3. 요각속도 예측 모델

기존 차선 유지 시스템에서 요각속도를 얻고자 곡률 반경과 차량 속도를 사용하여 요각속도를 예측 하거나 고가의 관성 측정 장치 센서를 사용하여 요각속도 값을 얻는다. 이러한 두 방법을 대체하고자 LKAS에서 사용되는 변수들인 도로모델 계수 값(c_0, c_1, c_2, c_3)과 조향각(δ) 데이터를 기반으로 요각속도($\dot{\psi}$)를 예측하는 인공지능 기반의 요각속도 예측 모델을 제안한다.

3.1 장 단기 순환신경망 요각속도 예측 모델 설계

위에서 언급한 바와 같이 입력 값은 여러 개의 센서 값이 들어온다. 그리고 출력은 요각속도 예측값인 하나의 값을 출력한다. 그러므로 장 단기 순환신경망의 여러 모델 중 여러 개의 입력을 받아 하나의 출력을 하는 모델을 선택하여 설계한다. Fig. 6은 설계한 장 단기 순환신경망 기반의 예측 모델을 나타내고 있다.

기존, 장 단기 순환신경망에서 사용되는 모델들은 일반적으로 장 단기 순환신경망 단독으로 사용된다. 하지만 설계자에 따라 장 단기 순환신경망에 합성곱 계층(Convolution layer)이나 단층 퍼셉트론(Single layer perceptron) 혹은 다층 퍼셉트론(Multi

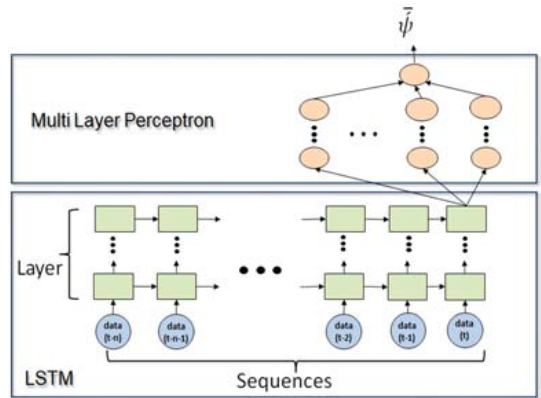


Fig. 6 LSTM based many to one prediction model

layer perceptron)을 연결하여 사용하기도 한다. 본 논문에서는 데이터의 예측을 목표로 하므로 회귀 성능이 높은 퍼셉트론을 장 단기 순환신경망 모델에 직렬 연결시켜 적은 계층의 장 단기 순환신경망 모델에서도 회귀 성능을 높이고자 하였다. 본 논문에서는 장 단기 순환신경망의 계층 수는 1, 시계열은 15로 설계하였으며, 퍼셉트론(Multi layer perceptron) 입력층의 노드는 10개 출력층의 노드는 1개로 설계하였다. 여기서, 노드의 개수와 계층의 개수는 설계자의 임의대로 설계할 수 있으며 발견법(Heuristics)한 방법으로 찾아야 한다. 이처럼 설계한 모델을 차선정보와 조향각 기반의 요각속도를 예측하는 모델 만들기 위해 학습 과정은 밑 3.2에서 설명한다.

3.2 장 단기 순환신경망 요각속도 예측 모델 학습

학습을 위해 차량을 운행하여 데이터를 취득한 후, 입력 값은 카메라 영상으로부터 취득한 도로모델의 계수 값(c_0, c_1, c_3, c_4)과 차량 내부정보인 조향각(δ)으로 선정하고, 목표값은 관성 측정 장치 센서로부터 취득한 요각속도($\dot{\psi}$)값을 선정하여 학습을 진행하였다. 이처럼 입력 값으로 도로 모델의 계수 값과 조향각을 선정한 이유는 곡률 반경과 조향각 요각속도의 데이터를 그래프로 비교하면 비슷한 개형을 나타내기 때문에 가장 직접적인 연관성이 있을 것이라 생각하였기 때문이다. 또한, 카메라 센서와 차량 내부 정보 이외의 다른 센서를 사용하지 않고 요각속도를 추정하기 위함이다. 다음 Fig. 5는 장

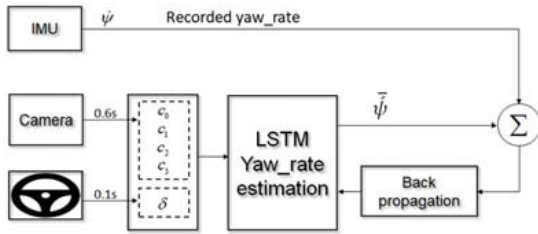


Fig. 7 LSTM based many to one prediction model learning structure

단기 순환신경망 요각속도 예측 모델을 위한 학습 구조를 나타내고 있다.

위 Fig. 7과 같이 도로모델의 계수 값(c_0, c_1, c_3, c_4)과 차량 조향각(δ)을 입력 값으로 전달하면 예측 요각속도($\bar{\psi}$)이 출력되고, 이 값은 관성 측정 장치 센서로부터 측정된 요각속도(ψ)와 오차를 제공하여 평균하는 평균 제곱 오차(MSE)기반의 비용함수(Cost function)가 0으로 수렴하도록 학습된다. 여기서, 학습 알고리즘은 퍼셉트론은 역전파 알고리즘이고, 장 단기 순환신경망은 시간에 따른 역전파 알고리즘으로 학습하였다. 최적화 기법은 아담 최적화 기법(Adam solver)을 사용하여 학습하였다.

4. 시뮬레이션

4.1 데이터 수집

본 논문에서 요각속도 예측 모델 학습을 위해 도로교통안전공단의 주행 시험장 코스 중 고속 주회로를 주행하며 카메라 센서와 차량 내부 정보인 조향각, 관성 측정 장치 센서로부터 데이터를 취득하였다. 주행 데이터는 차량 속도 90 km/h로 5분간 주행하였으며 데이터의 양은 이미지 4,300개, 조향각과 요각속도 값은 25,795개이다.

4.2 학습 및 시험 데이터 구축

데이터는 크게 카메라 데이터와 차량 내부 조향각, 관성 측정 장치 센서 데이터로 나누어져 있다. 관성 측정 장치 센서와 차량 내부정보 조향각은 10 ms마다 요각속도($\dot{\psi}$)와 조향각(δ)을 각각 출력시키며, 이 값들은 전처리 과정 없이 저장된다. 다음으로, 카메라 센서로부터 출력되는 이미지 데이터는 전처리 과정을 거쳐 도로 모델의 계수 값들을 추출

하고, 이 값들은 60 ms마다 저장된다. 이렇게 전처리 과정을 거치는 이유는, 기존 차선 유지 시스템에서 쓰이는 데이터를 기반으로 요각속도($\bar{\psi}$)를 예측하기 위함이다. 전처리 과정은 이미지로부터 차선을 인지하고, 인지된 차선을 3차 방정식으로 근사(Fitting)하여 도로 모델 계수 값(c_0, c_1, c_3, c_4)을 추출한다. 하지만 이 데이터들을 저장할 때 카메라 센서와 관성 측정 장치 센서, 차량 내부정보의 출력 주기가 서로 다르므로 문제가 발생한다. 문제를 해결하고자 데이터를 구성할 때, 가장 빠르게 출력하는 10 ms 단위를 기본으로 하되 카메라 센서 데이터는 새로운 데이터가 들어 올 때 가지 이전 값을 계속 유지하도록 데이터 표를 작성한다. 즉, 60 ms 동안 카메라 센서값은 같은 값을 유지하지만, 관성 측정 장치 센서값인 요각속도($\dot{\psi}$)와 센서 내부 정보 값인 조향각(δ)은 10 ms마다 갱신된 값을 사용한다. 데이터 표를 확인하면 다음 Table 1과 같다.

각 센서로부터 저장된 데이터를 위와 같이 구성하면, 센서로부터 출력되는 주기가 다른 것에 대한 문제는 해결할 수 있으나, 도로 모델의 계수 값이 변화하지 않는 구간에서 학습이 올바르게 되지 않아 잘못된 출력 값이 나올 수 있다. 하지만 장 단기 순환신경망은 데이터의 시계열을 고려할 수 있으므로, 다음 Fig. 8과 같이 입력 값과 목표값을 선정하여 이 문제를 해결하여, 요각속도를 예측하고자 한다.

Table 1 The collected data table

Time(ms)	c_0	c_1	c_2	c_3	δ	$\dot{\psi}$
0	0	0	0	0	0	0
10	0	0	0	0	-0.01043	-0.04747
20	0	0	0	0	-0.01043	-0.04782
30	0	0	0	0	-0.01043	-0.04869
40	0	0	0	0	-0.01043	-0.0473
50	0	0	0	0	-0.01043	-0.04695
60	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01043	-0.04835
70	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01054	-0.04835
80	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01054	-0.04957
90	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01065	-0.04992
100	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01077	-0.05009
110	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01088	-0.05027
120	-0.01367	3.49E-05	-0.00156	1.35E-07	-0.01099	-0.05009
130	-0.01367	3.49E-05	-0.00156	1.35E-07	-0.01122	-0.04852

	Time(ms)	c_0	c_1	c_2	c_3	δ	$\dot{\psi}$	
Data(t-9)	0	0	0	0	0	0	0	
Data(t-8)	10	0	0	0	0	-0.01043	-0.04747	
Data(t-7)	20	0	0	0	0	-0.01043	-0.04782	
Data(t-6)	30	0	0	0	0	-0.01043	-0.04869	
Data(t-5)	40	0	0	0	0	-0.01043	-0.0473	
Data(t-4)	50	0	0	0	0	-0.01043	-0.04695	
Data(t-3)	60	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01043	-0.04835	
Data(t-2)	70	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01054	-0.04835	
Data(t-1)	80	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01054	-0.04957	
Data(t)	90	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01065	-0.04992	Target Data
	100	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01077	-0.05009	
	110	-0.00977	3.49E-05	-0.00156	1.37E-07	-0.01088	-0.05027	
	120	-0.01367	3.49E-05	-0.00156	1.35E-07	-0.01099	-0.05009	
	130	-0.01367	3.49E-05	-0.00156	1.35E-07	-0.01122	-0.04852	

Fig. 8 The input data and the target data indicated in the data configuration

위 Fig. 8에서 보이듯이 10 ms마다 저장된 데이터에서 도로 모델의 계수 값(c_0, c_1, c_3, c_4), 조향각(δ)을 하나의 행으로 구성한다. 구성된 각 행은 장 단기 순환신경망의 각각의 메모리 블록에 순차적으로 입력된다. 그리고 입력된 데이터 중에 가장 최근의 시간과 일치하는 요각속도($\dot{\psi}$)를 목표값으로 선정하여 학습한다. 또한, 도로모델의 계수 값은 60 ms 동안 변화가 없으므로, 시계열 길이를 6 이하로 선정하면 데이터의 변화를 인지 할 수 없기 때문에 시계열을 길이를 6 이상으로 설정한다. Fig. 8에서의 시계열 길이는 t 부터 $t-9$ 까지 총 10을 나타내고 있다. 이렇게 입력 값과 목표값을 선정함으로써, 먼 과거로부터 현재까지의 모든 변화를 확인하여 요각속도 값 예측이 가능하다. 본 논문에서는 시계열 길이를 15로 설정 하여 학습함으로써 과거 150 ms 동안의 차량 움직임을 기반으로 요각속도를 예측한다. 저장된 데이터는 7:3의 비율로 각각 학습과 시험 데이터로 사용한다.

4.3 시뮬레이션 결과

장 단기 순환신경망의 시계열은 25, 계층은 1이며, 퍼셉트론 입력층 노드는 10개, 출력층 노드는 1개로 설계했다. 또한 순서쌍의 데이터(Tuple)를 사

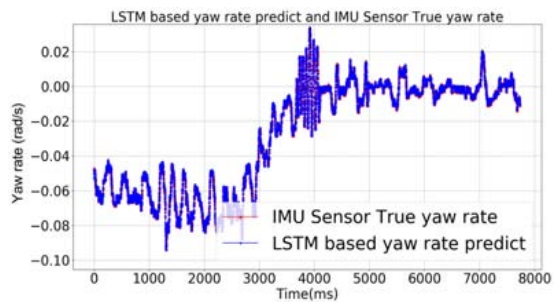


Fig. 9 Comparison of true yaw rate and LSTM based yaw rate

용한다. 학습을 위한 초기 학습률 변수는 0.0005이며 학습 수(Iteration) 5000번마다 0.1씩 곱해져 더 작아진다. 총 학습 수는 100000번으로 하였다. 그 결과, 비용함수(Cost fuction)의 최종값은 0.0013이 되는 것을 확인하였다. 학습된 모델에 시험 데이터로 실제 요각속도($\dot{\psi}$)값과 장 단기 순환신경망 기반의 예측 요각속도($\dot{\psi}$)값을 비교하면 다음 Fig. 9와 같다. 보다 자세한 결과를 확인하기 위해 실제 요각속도($\dot{\psi}$)값과 장 단기 순환신경망 기반의 예측 요각속도($\dot{\psi}$)의 오차를 나타내면 Fig. 8과 같다.

Fig. 9와 10을 통해 장 단기 순환신경망기반의 요각속도 예측 모델이 실제 관성 측정 장치 센서에서 측정된 요각속도 값을 예측할 때 오차 값이 0.005 rad/s

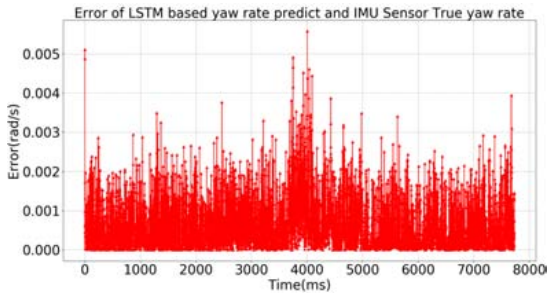


Fig. 10 Error of LSTM based yaw rate predict and true yaw rate

를 넘기지 않는 것을 확인할 수 있다. 또한, 평균 제곱근 편차(RMSE)값을 계산하면 0.0517이 나오는 것을 확인하였다.

5. 결론

본 논문은 기존 차선 유지 시스템에서 요각속도를 얻고자 곡률 반경과 차량 속도를 사용하여 수학적 모델로 요각속도 예측하거나 고가의 관성 측정 장치 센서를 사용하여 요각속도 값을 얻는 방법을 대체하고자 카메라 센서와 차량 내부 정보 조향각을 사용하여 딥러닝 모델 중 하나인 장 단기 순환신경망 모델을 기반의 관성 측정 장치 센서를 대체할 수 있는 요각속도 예측 모델을 제안하였다.

장 단기 순환신경망 기반의 요각속도 예측기를 설계함에 있어서, 기존 여러 개의 입력을 받아 하나의 출력을 주는 장 단기 순환신경망 모델에 퍼셉트론을 직렬연결한 모델을 설계하여 사용하였으며, 학습과 시험 데이터는 도로교통안전공단의 주행 시험장의 고속 주회로를 주행하며 데이터를 취득하였다. 데이터는 카메라 센서와 관성 측정 장치 센서와 차량 내부 조향각 센서로부터 취득하였으며, 센서의 주기가 달라 발생하는 시간 지연 문제는 시간 주기를 맞춰 타임 테이블을 작성하고 시계열을 길게 설정한 후 학습하여 해결하였다. 이렇게 학습이 완료된 모델에 카메라 센서와 차량 내부 조향각 센서를 입력 값으로 인가하면 차량의 과거부터 현재까지의 움직임을 고려하여 요각속도의 예측값이 출력되는 것을 확인하였으며, 예측값이 고가의 관성 측정 장치 센서에서 측정되어 나온 요각속도와 비슷한 값을 출력한 것을 확인하였다.

결과적으로 카메라 센서와 차량 조향각만으로 차선 유지 시스템에서 요구하는 요각속도 값을 딥러닝 기법의 하나인 장 단기 순환신경망을 사용하여 고가의 관성 측정 장치 센서의 기능을 하는 요각속도 예측 모델을 완성할 가능성을 확인하였다.

하지만 요각속도의 값은 차량의 속도, 차량의 탑승 인원 도로의 상태 등에 따라 값이 변화하며, 차량의 외부 여건에 따라 변화한다. 즉, 요각속도에 직/간접적 영향을 주는 변수를 파악하고 중요한 변수를 선정하여 입력 변수로 추가해야 한다. 예로써 차량의 속도, 차량의 무게, 타이어 공기압 등을 들 수 있다. 또한, 이에 따른 데이터를 다양한 운전자들을 통해 구축하고 학습하여 일반적인 모델을 완성해야 한다.

본 논문에서는 차선 유지 시스템에서 쓰이는 기본적인 변수만을 선정하여 학습을 통해 요각속도 추정을 해보았다. 추후, 데이터의 분석을 통해 데이터의 상관관계를 파악하여 입력 변수를 추가로 선정하여 차선 유지 시스템에서 요구되는 요각속도를 관성 측정장치 센서 없이 정확하게 얻을 수 있는 모델을 연구할 예정이며, 추후에는 차선 유지 시스템 뿐만 아닌, 차량 운행 데이터를 분석하여 관성 측정 장치 센서를 대체할 모델을 만드는 연구를 진행할 예정이다.

후 기

본 연구는 산업통상자원부 및 한국산업기술진흥원이 지원하는 경제협력권산업육성사업(비즈니스협력형 R&D, R0004529)의 지원으로 수행된 연구입니다.

References

- 1) H. Kawazoe, T. Murakami, O. Sadano, K. Suda and H. Ono, "Development of a Lane-Keeping Support System," SAE 2001-01-0797, 2001.
- 2) P. Raksincharoensak, M. Nagai and M. Shino, "Vision Based Lane Keeping Assistance Control by Using Drive-by-Wire Electric Vehicle," 12th World Congress on ITS, 2005.
- 3) E. Dickmanns and A. Zapp, "A Curvature-based

- Scheme for Improving Road Vehicle Guidance by Computer Vision,” Proceedings of SPIE Conference on Mobile Robots, Vol.727, pp.161-198, 1986.
- 4) E. Dickmanns, “The Development of Machine Vision for Road Vehicles in the Last Decade,” IEEE Intelligent Vehicle Symposium, Vol.1, pp.268-281, 2002.
 - 5) S. Horiuchi and K. Sunada, “Synthesis of Driver Assistance System for Lane-Following Using Generalized Predictive Control,” Proceedings of AVEC, pp.467-472, 1998.
 - 6) C. Taylor, J. Kosecka, R. Blasi and J. Malik, “A Comparative Study of Vision-based Lateral Control Strategies for Autonomous Highway Driving,” International Journal of Robotics Research, Vol.18, No.5, pp.442-453, 1999.
 - 7) P. Leelavansuk, K. Shitamitsu, H. Mouri and M. Nagai, “Study on Cooperative Control of Driver and Lane-Keeping Assistance System,” Proceedings of the International Symposium on Advanced Vehicle Control, pp.219-224, 2002.
 - 8) J. P. Switkes and J. C. Gerdes, “An Energy Based Performance Bound for Lanekeeping Assistance with Force Feedback,” Proceedings of AVEC, AVEC060096, 2006.
 - 9) M. Shimakage, S. Satoh, K. Uenuma and H. Mouri, “Design of Lane-keeping Control with Steering Torque Input,” JSAE Review, Vol.23, pp.317-323, 2002.
 - 10) S. -J. Ahn and M. -H. Han, “Research of the Lane Recognition for an Advanced Vehicle System,” Journal of Korean Institute Information Technology, Vol.5, No.1, pp.136-142, 2007.
 - 11) J. Y. Chung, C. Gulcehre, K. H. Cho and Y. Bengio, “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling,” <http://arxiv.org/abs/1412.3555>, 2014.
 - 12) S. Hochreiter and J. Schmidhuer, “Long Short-term Memory,” Neural Computation, Vol.9, No.8, pp.1735-1780, 1997.
 - 13) Y. Bengio, P. Simard and P. Frasconi, “Learning Long-term Dependencies with Gradient Descent is Difficult,” IEEE Transactions on Neural Networks, Vol.5, No.2, pp.157-166, 1994.
 - 14) F. A. Gers, J. Schmidhuber and F. Cummins, “Learning to Forget: Continual Prediction with LSTM,” Proceedings of International Conference on Artificial Neural Networks, pp.850-855, 1999.
 - 15) H. Sak, A. Senior and F. Beaufays, “Long Short-term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling,” Proceedings of the Annual Conference of the International Speech Communication Association, pp.338-342, 2014.
 - 16) E. Dickmanns, “The Development of Machine Vision for Road Vehicles in the Last Decade,” Proceedings of IEEE Intelligent Vehicle Symposium, Vol.1, pp.268-281, 2002.
 - 17) E. Million, The Hadamard Product, <http://buzzard.ups.edu/courses/2007spring/projects/million-paper.pdf>, 2007.