한국ITS학회논문지 J. Korea Inst. Intell. Transp. Syst.

딥러닝으로 추정한 차량대기길이 기반의 감응신호 연구

Study of the Operation of Actuated signal control Based on Vehicle Queue Length estimated by Deep Learning

이 용 주* · 심 민 경** · 김 용 만*** · 이 상 수**** · 이 철 기*****

* 주저자 : 아주대학교 교통연구센터 연구교수
** 공저자 : 아주대학교 교통공학과 석사과정
*** 공저자 : 도로교통공단 교통과학장비처 차장대우
**** 공저자 : 아주대학교 교통시스템공학과 교수
***** 교신저자 : 아주대학교 교통시스템공학과 교수

Lee Yong-Ju* · Sim Min-Kyung** · Kim Yong-Man*** · Lee Sang-Su**** · Lee Choul-Ki*****

* Dept. of Transportation Research Institute, Univ. of Ajou
** Dept. of Transportation Eng., Univ. of Ajou
*** Dept. of Road Equipment, KoROAD
**** Dept. of Transportation System Eng., Univ. of Ajou
***** Dept. of Transportation System Eng., Univ. of Ajou
† Corresponding author : Lee Choul-Ki, cklee@ajou.ac.kr

Vol.17 No.4(2018)
August, 2018
pp.54~62

ISSN 1738-0774(Print)
ISSN 2384-1729(On-line)
https://doi.org/10.12815/kits.2018.17.4.54

Received 20 July 2018
Revised 7 August 2018
Accepted 21 August 2018

ⓒ 2018. The Korea Institute of Intelligent Transport Systems. All rights reserved.

요 약

본 연구는 인공지능 신호 구현의 일환으로서, 딥러닝을 통해 실시간으로 추정하는 차량대기길이 기반의 감응식 신호 알고리즘을 제시하였다. 알고리즘의 구현을 위해 딥러닝 모형을 구현한 텐서플로우에 미시적 교통시뮬레이터인 Vissim을 제어하는 API, 즉 COM Interface를 구축하였다. Vissim에서 신호주기별로 수집된 링크통행시간과 통과교통량이 텐서플로우에 전달되면 학습이 완료된 딥러닝 모형을 통해 접근로별 차량대기길이가 추정된다. 접근로별 차량대기길이를 기반으로 신호시간을 산정한 후 Vissim 내부의 신호등화를 조정하여 시뮬레이션 한다. 본 연구에서 개발한 알고리즘은 현 TOD 방식에 비해 차량 지체가 약 5% 감소한 것으로 분석되었으며, 이는 네트워크 내 하나의 교차로만 대상으로 적용하여 그 효과가 제한된 것이며, 축 또는 네트워크 제어로의 공간적 확대방안을 향후연구로 제시하였다.

핵심어 : 인공지능 신호, 딥러닝, 차량대기길이, Vissim COM Interface, 텐서플로우

ABSTRACT

As a part of realization of artificial intelligence signal(AI Signal), this study proposed an actuated signal algorithm based on vehicle queue length that estimates in real time by deep learning. In order to implement the algorithm, we built an API(COM Interface) to control the micro traffic simulator Vissim in the tensorflow that implements the deep learning model. In Vissim, when the link travel time and the traffic volume collected by signal cycle are transferred to the tensorflow, the vehicle queue length is estimated by the deep learning model. The signal time is calculated based on the vehicle queue length, and the simulation is performed by adjusting the signaling inside Vissim. The algorithm developed in this study is analyzed that the vehicle delay is reduced by about 5% compared to the current TOD mode. It is applied to only one intersection in the network and its effect is limited. Future study is proposed to expand the space such as corridor control or network control using this algorithm.

Key words : AI Signal, Deep Learning, Vehicle Queue Length, Vissim COM Interface, Tensorflow

Ⅰ. 서 론

1. 배경 및 목적

전 세계적으로 일어나고 있는 제4차 산업혁명의 토대는 다양한 혁신 분야에 적용되고 있는 빅데이터와 인공지능(AI)일 것이다. 교통 분야에서도 빅데이터와 인공지능을 활용한 새로운 접근의 교통 연구가 활발히 추진되고 있는 가운데, 최근 교통신호 부문에 인공지능 신호(이하 'AI 신호') 구현이라는 목표가 제시되고 있다.

AI 신호에 대해 아직 명확한 방법론이 정립된 바는 없으나, 이를 구현하기 위해서는 가용할 수 있는 데이터를 이용한 단계적인 접근이 필요하다고 판단된다. AI 신호 구현의 기반이 되는 선행 연구로서, 지능형교통체계(ITS)에서 생성되는 실시간 링크통행시간 및 통과교통량을 이용하여 차량대기길이를 딥러닝(Deep Learning) 모형으로 추정하는 연구(Lee et al., 2018)를 들 수 있다. 본 연구는 선행연구에서 개발된 딥러닝 차량대기길이 추정모형을 신호제어에 처음으로 적용하여 딥러닝 모형으로 추정한 차량대기길이 기반의 감응신호를 구현하여 그 효용성과 발전가능성을 증명함으로써 AI 신호 구현의 초석을 마련하는 데 그 목적이 있다.

2. 연구 방법

본 연구에서는 선행 연구에서 개발된 '딥러닝 기반의 차량대기길이 추정모형'을 신호제어 부문에 적용함으로써 실시간으로 추정하는 차량대기길이 기반의 감응식 신호제어(이하 '감응신호') 알고리즘을 구현한다. 감응신호 알고리즘 개발을 위해 딥러닝 모형을 구현한 텐서플로우와 미시적 교통시뮬레이터인 Vissim을 연결하는 COM Interface를 구축한다. Vissim에서 신호주기별로 수집된 링크통행시간과 통과교통량이 텐서플로우에 전달되면 학습이 완료된 딥러닝 모형을 통해 접근로별 차량대기길이가 추정된다. 접근로별 차량대기길이를 기반으로 신호시간을 산정한 후 Vissim 내부의 신호등화를 조정하여 시뮬레이션 한다.

시뮬레이션은 선행연구에서 개발한 딥러닝 모형을 적용하기 위해 선행연구와 동일한 네트워크를 대상으로 시행하였으나 단, 차량대기길이 기반의 감응신호를 적용하는 교차로는 1개소만 대상으로 한다. 그 이유는 텐서플로우를 이용한 Vissim COM Interface 구축의 불확실성과 감응신호 알고리즘 구현의 용이성 등 초기연구에서 겪는 기술적 구현의 문제를 우선적으로 해결해야하기 때문이다.

마지막으로 본 연구에서 개발하는 딥러닝 추정 차량대기길이 기반의 감응신호 알고리즘의 효과를 평가하기 위해 시뮬레이션 결과로 도출되는 평균지체를 평가지표로 하고 현재의 신호운영방식과 비교한다.

Ⅱ. 이론적 고찰

1. 딥러닝 및 텐서플로우

딥러닝(Deep Learning)은 기계학습(Machine Learning) 분야 중 하나인 인공신경망(Artificial Neural Network)[1] 이론을 심화·발전시킨 인공지능(AI)으로서, 복잡한 문제 해결을 위한 신경망의 층수 확대, 심층신경망(Deep Neural Network, DNN)·합성곱 신경망(Convolutional Neural Network, CNN)·순환신경망(Recurrent Neural Network, RNN)·LSTM(Long Short Term Memory) 등 신경망 구조의 다양화, Batch Gradient Descent·Stochastic Gradient Descent(SGD)·Momentum·AdaGrad·RMSProp·ADAM(Adaptive Moment Estimation) 등 파라미터 최적화 과정에

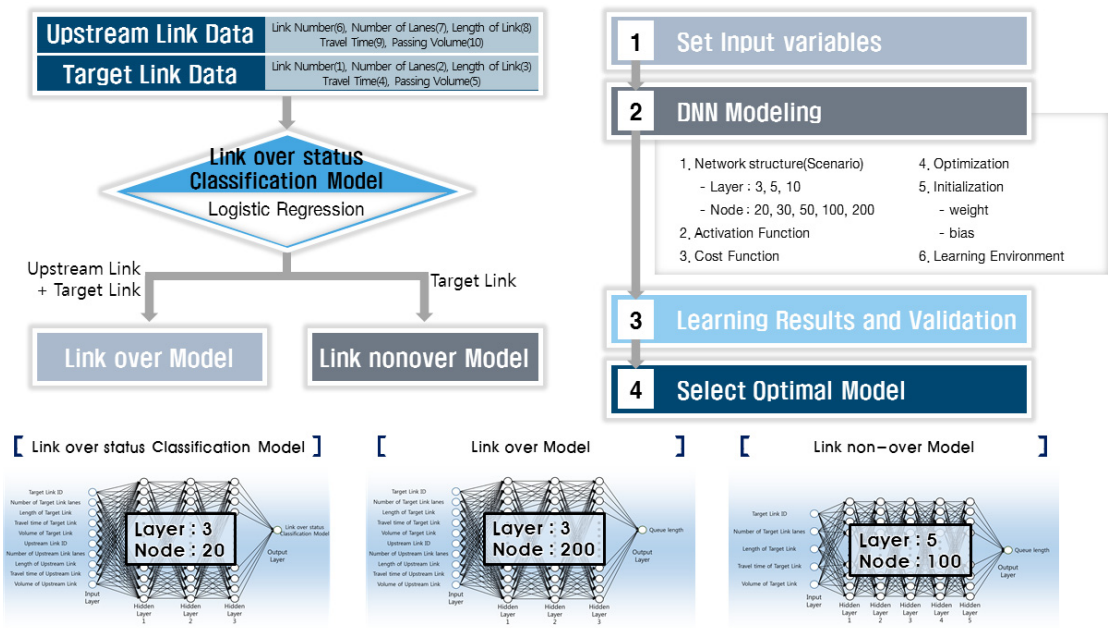
1) 인공신경망(Artificial Neural Network)은 인간의 뇌를 모델로 하여 정보를 처리하는 알고리즘을 뜻함

적용하는 경사하강법(Gradient Descent Algorithm)의 고도화, 경사감소소멸(Vanishing Gradient Descent) 문제에 대응한 활성화 함수(Activation Function)의 개선, 과적합(Overfitting) 문제를 방지하는 규제화(Regulation) 기법 개발 등 이론적 발전이 지속적으로 이루어지고 있다.

딥러닝 모형을 구현한 텐서플로우(TensorFlow)는 구글에서 만든 오픈소스 소프트웨어 라이브러리로서, Data Flow Graph를 사용하여 수치 연산을 한다. Data Flow Graph에서는 수학 계산과 데이터의 흐름을 노드(Node)와 엣지(Edge)를 사용한 방향 그래프로 표현한다. 노드는 수학적 계산, 데이터 입 · 출력, 그리고 데이터의 읽기 · 저장 등의 작업을 수행한다. 엣지는 노드들 간 데이터의 입출력 관계를 나타내는데, 동적 사이즈의 다차원 데이터 배열(텐서, tensor)을 옮기는 역할을 수행한다.

2. 딥러닝을 활용한 차량대기길이 추정모형 개발(Lee et al., 2018)

Lee et al.(2018)는 교통운영 개선에 필요한 빅데이터 및 인공지능 모델 개발의 일환으로서, 도시부의 링크 통행시간 및 통과교통량 등 가용 데이터 등을 이용하여 교통변수로 활용도가 높은 차량대기길이와의 관계를 딥러닝(Deep Learning)을 통해 학습하고 추정하는 인공지능 모델을 구축하였다. 차량대기길이 추정모형은 우선 로지스틱 회귀모형을 구축하여 차량대기길이의 링크 초과여부를 분류한 후 링크 초과 및 링크 미초과 상황에서 각각 차량대기길이를 추정하는 3개의 모형으로 모델링하였다. 딥러닝 모형은 텐서플로우로 구현하였으며, 모든 모형은 DNN 구조로서 은닉층과 노드 개수를 다양화하여 학습 및 테스트 후 최소 오차를 나타내는 네트워크 구조를 선정하였다. 차량대기길이 링크 초과여부 분류 모형은 약 98%의 정확도를 나타냈으며, 미초과 모형은 15% 미만, 초과 모형은 5% 미만의 오차를 각각 나타내었다. 링크별 평균 오차는 12%로 도출되었다. 이를 기존 검지기 데이터 기반의 방식과 비교한 결과 오차가 약 39% 감소된 것으로 분석되었다. 모형의 구조 및 3개 세부모형별 입력변수와 신경망 구조는 다음 그림과 같다.



〈Fig. 1〉 Development of Vehicle Queue length Estimation Model Using Deep Learning

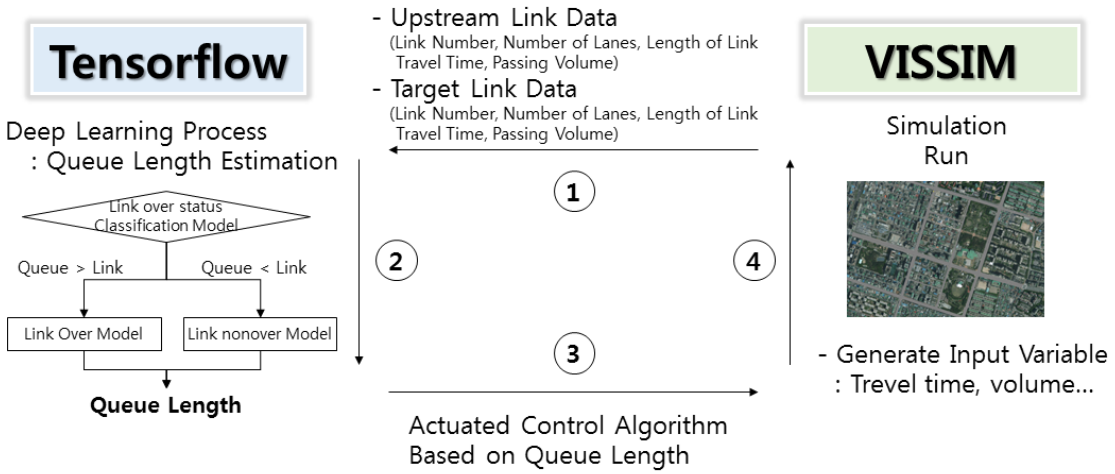
Ⅲ. 딥러닝과 Vissim 인터페이스 구현

1. 딥러닝의 텐서플로우와 Vissim의 인터페이스

Vissim은 다양한 API(Add-on modules Programming Interface)를 제공하는데, 이를 통해 별도의 응용프로그램을 Vissim에 통합할 수 있다. 이는 Vissim의 COM(Component Object Model) Interface 기능으로서, 데이터 준비 및 전처리, 제어알고리즘이 포함된 시나리오 실행을 위한 시퀀스 제어, 모든 네트워크 개체 속성에 대한 접근 등이 가능하다. COM 개체는 VBA, VBS, Python, C, C ++, C #, Delphi 및 MATLAB을 비롯한 광범위한 프로그래밍 및 스크립팅 언어에서 구현될 수 있다.

본 연구에서는 텐서플로우에 차량대기길이를 추정하는 딥러닝 모델과 python 코드로 작성된 COM Interface를 구축하여 Vissim을 제어한다. 이는 텐서플로우가 Python을 지원함에 따라 가능하다.

딥러닝과 감응신호 등 알고리즘의 시퀀스는 다음과 같다. Vissim 시뮬레이션이 실행되면 신호주기별로 각 접근로의 링크통행시간과 통과교통량이 수집되고, 이를 포함하여 접근로 ID 및 차로수, 길이 등의 딥러닝 입력변수에 필요한 데이터가 텐서플로우에 전달된다. 텐서플로우에서는 전달 받은 데이터를 학습이 완료된 최적 딥러닝 모형에 입력하여 접근로별 차량대기길이를 추정한다. 이후 추정된 접근로별 차량대기길이 기반의 감응신호 알고리즘을 통해 신호시간을 산정한 후 Vissim 내부의 신호등화를 조정하여 시뮬레이션한다. 이러한 시퀀스가 분석시간 동안 반복 실행되며, 주기별로 접근로별 차량지체가 도출된다.



〈Fig. 2〉 Data Processing Flow Through Interface Between Vissim and Tensorflow

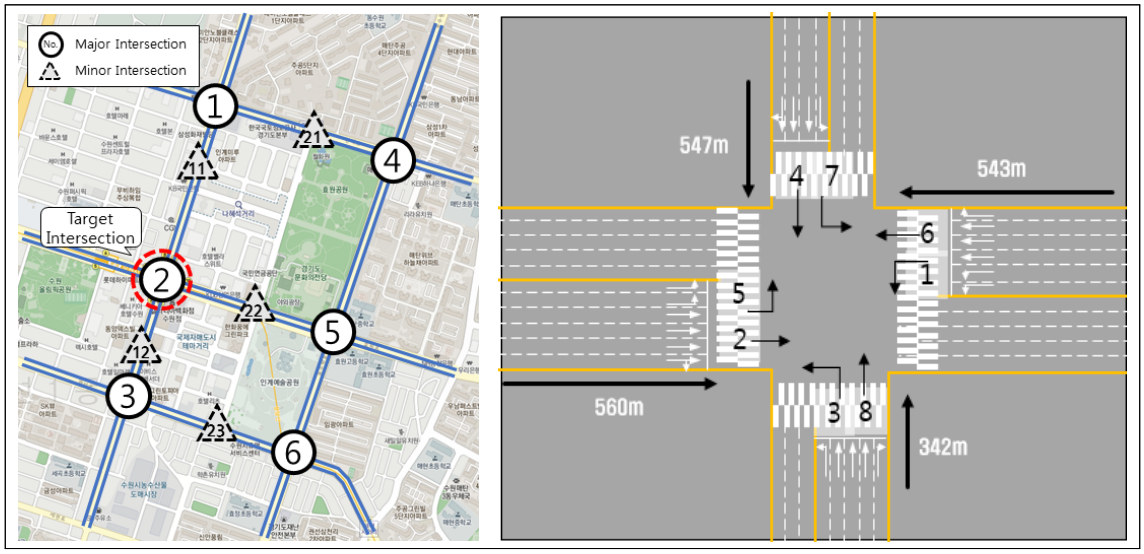
2. 시뮬레이션 구현 환경

시뮬레이션 대상 네트워크는 선행연구의 학습데이터 생성지역과 동일하게 설정하며, 수원시 내 상습적으로 교통혼잡이 발생하는 CBD지역인 인계동에 위치해 있다. 6개의 중요교차로(CI)로 구성된 3X2 구조이며, 또한 5개의 비중요교차로가 위치해 있어 총 11개의 교차로로 구성되어 있다. 단, 연구의 성격상 딥러닝 모형을 통해 차량대기길이를 추정하고 이를 기반으로 감응신호를 적용하는 교차로는 1개소만 대상으로 한다.

오후첨두(17~19시) 시간대를 대상으로 하여 비포화에서부터 과포화까지 다양한 교통상황이 발생하는 교차

로를 대상으로 하였으며, 실제 교차로별 신호DB 및 교통여건을 반영하고, 해당 시간대의 ITS 통행속도를 토대로 유사한 소통상황을 Vissim으로 구현한다.

시뮬레이션을 통한 데이터 분석시간은 2시간으로 설정하며, 초기 네트워크 균형(Network Equilibrium)을 위한 1시간을 포함하여 총 3시간 동안 시뮬레이션이 진행된다.

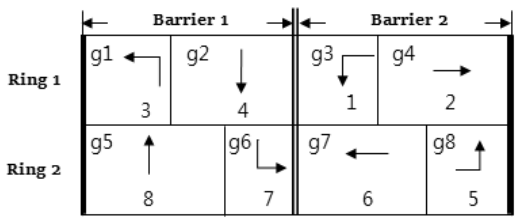


〈Fig. 3〉 Road Network Structure

Ⅳ. 차량대기길이 기반의 감응신호 알고리즘 구현

1. 교차로 신호운영현황(TOD 방식)

해당 교차로를 비롯한 전체 네트워크(11개 교차로)는 TOD(Time of Day) 방식으로 운영 중이며, 분석 시간대(17~19시)를 포함한 17~22시에는 180초의 공통주기로 운영되고 있다. 신호현시는 이중링(Dual-Ring) 구조이나 감응제어가 적용되지 않아 단일링(Sing-Ring) 구조의 중복현시(Lead and Lag Left Turns Protected with Overlap Phase)로 구성된 6현시 체계이다. 해당 교차로의 신호현시구조 및 신호시간 등 신호DB 현황은 다음과 같다.



<Signal Phase Structure>

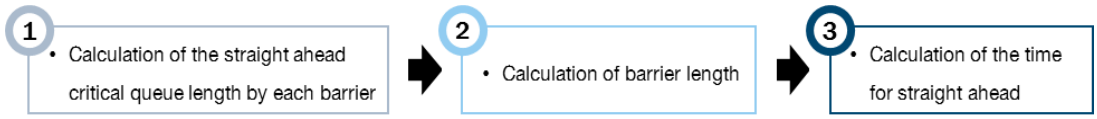
Phase Sequence						Cycle Length	Offset
1	2	3	4	5	6		
						180	120
35	29	46	27	15	28		

<Signal Phase Time Plan>

〈Fig. 4〉 Signal Phase Time

2. 차량대기길이 기반의 감응신호 알고리즘

차량대기길이 기반의 감응신호 알고리즘의 신호시간 산정방식은 COSMOS에서 제시한 포화도(DS)를 이용한 방식에서 차량대기길이로 대체하여 Barrier 구분 및 Split을 수행한 것으로서, 먼저 접근로별로 딥러닝에 의해 추정된 차량대기길이를 입력 받고 각 Barrier 내에서 직진 이동류의 임계 차량대기길이를 선정한다. Barrier 1의 경우 4번과 8번 직진 이동류 간의 차량대기길이를 비교하고 Barrier 2에서는 2번과 6번 직진 이동류 간의 차량대기길이를 비교하여 각각 임계 차량대기길이를 선정한다. 이후 Barrier 1과 2의 임계 차량대기길이의 비율에 따라 각 Barrier의 길이를 산정한다. 그리고 각 Barrier 길이에 대해 보행신호를 고려한 최소녹색시간 이상인지를 체크하고 미만일 때에는 보정을 수행한다. Barrier의 최소녹색시간은 임계 좌회전 시간과 횡단보도 녹색시간을 합한 시간으로서, Barrier1과 Barrier2의 최소녹색시간을 각각 85초, 60초로 설정한다. 마지막으로 각 Barrier에서 현 좌회전현시 시간을 제외한 시간을 직진현시(g2, g4, g5, g7)에 할당한다. 알고리즘의 절차를 비롯하여 임계이동류 설정, Barrier 구분 및 Split 방식은 다음과 같다.



〈Fig. 5〉 Algorithm for Calculation of the Signal Phase

$$MQ_1 = Max(Q_{NB}, Q_{SB}) \quad (1)$$
$$MQ_2 = Max(Q_{EB}, Q_{WB}) \quad (2)$$

$$Barrier_1 = C \times \frac{MQ_1}{MQ_1 + MQ_2} \quad (3)$$
$$Barrier_2 = C \times \frac{MQ_2}{MQ_1 + MQ_2} \quad (4)$$

$$Min.Barrier_1 = Max((\min.g5 + \min.g6), (\min.g1 + \min.g2)) \quad (5)$$
$$Min.Barrier_2 = Max((\min.g8 + \min.g7), (\min.g3 + \min.g4)) \quad (6)$$

i) $Barrier_1 > Barrier_2$ 일 때,
$$Barrier_1 = C - Min.barrier_2 \quad (7)$$
ii) $Barrier_1 < Barrier_2$ 일 때,
$$Barrier_2 = C - Min.Barrier_1 \quad (8)$$

$$g2 = B_1 - g1 \quad (9)$$
$$g4 = B_2 - g3 \quad (10)$$
$$g5 = B_1 - g6 \quad (11)$$
$$g7 = B_2 - g8 \quad (12)$$

여기서, $Q_{NB,SB,EB,WB}$ = Queue Length by Direction(NB, SB, EB, WB)
MQ_n = Max of Queue (m), n = Barrier number(1, 2)
C = Cycle length
$Min.Barrier_n$ = Minimum Barrier time, n = Barrier number(1, 2)
$\min.gN$ = Minimum green time of phase N, (N = 1 ~ 8)

Ⅴ. 시뮬레이션 및 평가

1. 시뮬레이션 결과

접근로별로 추정된 차량대기길이와 이를 기반으로 한 감응신호 알고리즘에 의해 산출된 방향별 직진현시 시간 등의 시뮬레이션 결과는 다음과 같다.

〈Table 1〉 The Results of Simulation

number of Cycle	North Bound(g5)				South Bound(g2)				West Bound(g7)				East Bound(g4)			
	Queue		Phase time		Queue		Phase time		Queue		Phase time		Queue		Phase time	
	Before	After	Before	After	Before	After	Before	After	Before	After	Before	After	Before	After	Before	After
1	51.4	64.5	75	54	54.1	126.2	64	65	43.0	70.1	42	53	129.2	100.2	43	52
2	61.9	81.9	75	67	58.1	147.6	64	78	37.5	52.1	42	40	120.7	87.4	43	39
3	52.1	106.4	75	68	81.4	131.1	64	79	39.7	49.4	42	39	86.9	75.5	43	38
4	46.0	79.4	75	74	137.5	211.6	64	85	40.6	65.2	42	33	111.9	86.5	43	32
5	46.5	70.2	75	74	147.8	182.3	64	85	40.0	60.6	42	33	103.7	81.2	43	32
6	37.5	59.6	75	74	149.5	267.5	64	85	29.6	59.6	42	33	110.8	80.4	43	32
7	48.7	58.4	75	74	189.8	407.6	64	85	34.6	48.6	42	33	113.2	84.2	43	32
8	26.4	61.0	75	74	185.1	262.7	64	85	33.4	42.9	42	33	93.0	63.2	43	32
9	58.7	68.4	75	74	156.9	360.4	64	85	41.2	61.1	42	33	86.7	69.7	43	32
10	48.0	59.9	75	74	176.4	275.6	64	85	42.7	61.6	42	33	85.1	62.5	43	32
11	22.7	68.9	75	74	189.5	201.3	64	85	36.2	60.1	42	33	96.7	80.2	43	32
12	44.0	59.4	75	74	126.3	243.6	64	85	37.6	64.4	42	33	60.6	79.8	43	32
13	42.1	55.7	75	74	100.3	249.0	64	85	34.2	51.7	42	33	68.8	65.8	43	32
14	57.4	55.3	75	71	144.4	123.7	64	82	35.4	51.1	42	36	62.7	66.6	43	35
15	39.8	71.3	75	74	182.6	205.8	64	85	35.0	49.9	42	33	66.3	66.1	43	32
16	48.2	85.2	75	74	160.7	196.0	64	85	28.6	56.9	42	33	51.7	57.8	43	32
17	41.1	72.7	75	74	141.9	211.6	64	85	68.8	60.7	42	33	58.0	68.1	43	32
18	64.3	96.6	75	74	125.2	173.2	64	85	24.8	61.2	42	33	56.2	63.8	43	32
19	38.6	61.3	75	60	77.9	157.9	64	71	36.3	52.6	42	47	99.3	109.6	43	46
20	35.4	86.4	75	65	47.6	104.7	64	76	22.0	46.2	42	42	83.4	65.1	43	41
21	34.6	85.4	75	61	43.1	127.8	64	72	42.2	36.7	42	46	50.5	86.8	43	45
22	18.1	93.2	75	68	82.8	128.6	64	79	26.0	52.6	42	39	71.0	74.8	43	38
23	50.2	79.5	75	74	75.6	160.4	64	85	41.9	37.6	42	33	50.3	58.1	43	32
24	39.1	54.8	75	74	59.8	184.7	64	85	27.2	59.3	42	33	105.5	60.2	43	32
25	56.0	72.9	75	74	76.7	143.9	64	85	30.2	50.4	42	33	135.1	66.7	43	32
26	35.3	57.1	75	57	75.4	153.3	64	68	36.6	45.7	42	50	171.0	114.7	43	49
27	34.9	69.6	75	40	49.1	123.2	64	51	36.6	40.9	42	67	179.1	134.1	43	66
28	47.5	89.6	75	52	48.8	141.3	64	63	42.6	45.6	42	55	176.9	117.5	43	54
29	32.2	780.8	75	74	37.4	269.8	64	85	25.4	40.0	42	33	183.3	119.4	43	32
30	58.2	88.1	75	74	48.3	185.0	64	85	28.1	35.6	42	33	164.6	76.1	43	32
31	51.3	60.8	75	40	65.9	105.2	64	51	41.3	43.1	42	67	114.0	113.7	43	66
32	50.3	131.0	75	66	78.7	94.1	64	77	48.7	47.9	42	41	54.6	78.7	43	40
33	43.6	781.0	75	74	46.0	259.4	64	85	40.6	65.6	42	33	65.2	61.5	43	32
34	47.3	72.1	75	74	34.0	156.2	64	85	42.4	41.9	42	33	57.8	69.2	43	32
35	62.3	57.6	75	51	90.9	79.7	64	62	33.1	42.0	42	56	64.9	67.9	43	55
36	52.1	100.6	75	58	165.7	87.0	64	69	27.4	37.0	42	49	65.1	73.0	43	48
37	39.4	127.8	75	74	175.7	226.4	64	85	26.4	38.6	42	33	54.9	66.5	43	32
38	59.7	106.5	75	74	220.7	244.0	64	85	25.1	42.9	42	33	57.2	51.9	43	32
39	50.0	57.7	75	74	331.3	141.9	64	85	42.4	39.4	42	33	53.6	52.7	43	32
40	18.3	100.6	75	56	230.7	87.0	64	69	45.4	37.0	42	49	111.7	74.9	43	48

방향별 직진 현시시간의 변화를 살펴보면, 북쪽 방향 직진현시가 기존 64초에서 평균 68초로, 남쪽 방향도 75초에서 평균 79초로써 남북 방향의 직진 현시는 모두 약 4초 증가하였으며, 대신 서쪽 방향 직진현시는 42초에서 평균 39초로, 동쪽 방향도 43초에서 평균 38초로써 동서 방향의 직진 현시는 약 4초 감소하였다.

2. 알고리즘 평가

링크ID, 링크길이, 차로수, 링크통행시간과 통과교통량 등을 이용한 딥러닝 모형에 의해 추정된 차량대기길이 기반의 감응신호 알고리즘에 대한 평가는 제어지체(Control Delay)를 평가지표(MOE)로 하여 수행한다. 시뮬레이션 분석시간인 2시간(40주기) 동안 주기별 접근로별 평균지체를 산출하고 이를 이용하여 교차로 전체의 평균지체를 도출하며, 현재의 TOD 방식과 비교하여 지체감소 효과를 분석한다.

분석 결과 남쪽 방향(SB)으로의 추정된 차량대기길이가 길어서 신호시간이 증가한 남북 방면은 지체가 약 7~13% 감소한 반면 신호시간이 줄어든 동서 방면은 -1~3%로 지체가 증가한 곳도 있었다. 교차로 전체적으로는 지체가 62초/대에서 59초/대로 약 3초/대 감소하였다. 2시간 동안의 교통상황에서 네트워크 내 1개 교차로의 신호체계를 차량대기길이 기반의 감응신호로 운영한 결과 약 5%의 지체가 감소한 것이다.

〈Table 2〉 Travel speed estimation result

	Before(TOD)		After(Development Model)		Delay Reduction Rate(%)
	Delay [s/veh]	Volume [veh]	Delay [s/veh]	Volume [veh]	
NB	55.85	2,943	52.07	2,951	6.77
SB	45.89	3,798	39.90	3,840	13.05
WB	85.93	2,547	86.99	2,550	-1.23
EB	67.95	3,862	65.67	3,862	3.36
Total Delay	62.35	-	59.25	-	4.97

이러한 제약된 효과 즉 현 신호운영방식과의 차이가 유의미한지 분석하기 위해서 접근로별로 신뢰수준 95%로 Paired t-test를 수행한 결과, 다음과 같이 남쪽 방향(SB)은 차이가 유의미한 것으로 분석되었다.

〈Table 3〉 The Result of Paired t-test

	Mean	Standard Deviation	Standard Error of Means	95% Confidence Interval		*t*	Degree of Freedom	*p*-value
				Min	Max			
SB	8.14058	22.61840	3.57628	0.90687	15.37430	-2.276	39	0.028

Ⅵ. 결 론

본 연구는 AI 신호 구현의 일환으로서, 실시간의 교통수요 파악을 위해 한계와 문제점이 드러난 검지기 방식이 아닌 선행연구에서 개발된 딥러닝 기반의 실시간 링크통행시간 및 통과교통량을 이용한 차량대기길이 추정모형을 신호제어에 적용한 실시간 차량대기길이 기반의 감응신호 알고리즘을 제시하였다.

알고리즘의 구현을 위해 딥러닝 모형을 구현한 텐서플로우에 미시적 교통시뮬레이터인 Vissim을 제어하는 API, 즉 COM Interface를 구축하였다. Vissim에서 신호주기별로 수집된 링크통행시간과 통과교통량이 텐

서플로우에 전달되면 학습이 완료된 딥러닝 모형을 통해 접근로별 차량대기길이가 추정된다. 접근로별 차량대기길이를 기반으로 신호시간을 산정한 후 Vissim 내부의 신호등화를 조정하여 시뮬레이션 하였다.

11개의 교차로로 구성된 네트워크 내 하나의 교차로만 대상으로 본 연구에서 개발한 알고리즘을 적용한 결과 현 TOD 방식에 비해 차량 지체가 약 5% 감소한 것으로 분석되었다. 이는 교차로 그룹(SA, Sub Area)으로 묶여 있고, 네트워크 전체적으로 조율된 신호시간으로 운영 중인 상황에서 하나의 교차로만 감응신호로 운영함으로써 효과가 제약될 수밖에 없었다. 또한 기존의 교통상황을 유지한 채 최적의 TOD 플랜 보다 더 효과적인 신호운영에는 한계가 있었다. 따라서 돌발상황 발생, 교통패턴 변화 등 기존 교통상황과 변동이 있거나 또는 제어의 확장 즉, 축 또는 네트워크 제어로 확장에 대해 향후 연구가 필요하며, 본 연구는 AI 신호구현의 과정에 있어 초기 연구로써의 기술적 구현과 적용 가능성에 의의를 찾을 수 있었다.

ACKNOWLEDGEMENTS

이 논문은 2016년도 정부(국토교통부)의 재원으로 국토교통과학기술진흥원의 지원을 받아 수행된 연구임. (No.16TLRP-C105654-02, V2X 통신 인프라를 활용한 네트워크 신호운영 알고리즘 개발 및 성능검증)

REFERENCES

Bottou L.(1998), *Online Algorithms and Stochastic Approximations, Online Learning and Neural Networks*, Cambridge University Press.

Doh(2017), *Transportation Engineering*, ChungMoonGak.

Duchi et al.(2011), “Adaptive subgradient methods for online learning and stochastic optimization,” *The Journal of Machine Learning Research*, vol. 12, pp.2121-2159.

https://tensorflowkorea.gitbooks.io/tensorflow-kr/

Jeong et al.(2005), “Development of The Signal Control Algorithm Using Travel Time Informations of Sectional Detection Systems,” *The Journal of Korean Society of Transportation*, vol. 23, no. 8, pp.181-191.

Kingma D. P. and Ba J. L.(2014), *ADAM: A Method for Stochastic Optimization*, arXiv:1412.6980.

Lee et al.(2018), “Decelopment of Vehicle Queue Length Estimation Model Using Deep Learning,” *The Journal of the Korea Institute of Intelligent Transportation Systems*, vol. 17, no. 2, pp.39-57.

McCulloch W. S. and Pitts W.(1943), “A logical calculus of the ideas immanent in nervous activity,” *Bulletin of mathematical biophysics*, 5, pp.115-133.

Minsky M. and Papert S.(1969), *Perceptrons : An Introduction to Computational Geometry*, The MIT Press.

Roger P.(2010), *Traffic Engineering*, PEARSON.

Rosenblatt F.(1958), “The perceptron : A probabilistic model for information storage and organization in the brain,” *Psychological Review*, vol. 65, no. 6, pp.386-408.

Russell S. J. and Norvig P.(2003), *Artificial Intelligence : A Modern Approach, 2nd ed.*, Prentice Hall.

Tieleman T. et al.(2012), *Lecture 6.5-RMSProp, COURSERA : Neural Networks for Machine Learning*, Technical report.