energies

MDPI

Article

Comparative Study of Physics-Based Modeling and Neural Network Approach to Predict Cooling in Vehicle Integrated Thermal Management System

Duwon Choi [1,2,†], Youngkuk An [1,†], Nankyu Lee [2], Jinil Park [1] and Jonghwa Lee [1,*]

[1] Department of Mechanical Engineering, Ajou University, 206 World cup-ro, Yeongtong-gu, Suwon 16499, Gyeonggi, Korea; duwon@tenergy.co.kr (D.C.); stingyk@ajou.ac.kr (Y.A.); jpark@ajou.ac.kr (J.P.)

[2] Vehicle Calibration Team, Tenergy, 145 Gwanggyo-ro, Yeongtong-gu, Suwon 16229, Gyeonggi, Korea; nklee@tenergy.co.kr

* Correspondence: jlee@ajou.ac.kr

† These authors contributed equally to this work.

Received: 27 September 2020; Accepted: 8 October 2020; Published: 12 October 2020

check for updates

Abstract: Vehicle integrated thermal management system (VTMS) is an important technology used for improving the energy efficiency of vehicles. Physics-based modeling is widely used to predict the energy flow in such systems. However, physics-based modeling requires several experimental approaches to get the required parameters. The experimental approach to obtain these parameters is expensive and requires great effort to configure a separate experimental device and conduct the experiment. Therefore, in this study, a neural network (NN) approach is applied to reduce the cost and effort necessary to develop a VTMS. The physics-based modeling is also analyzed and compared with recent NN techniques, such as ConvLSTM and temporal convolutional network (TCN), to confirm the feasibility of the NN approach at EPA Federal Test Procedure (FTP-75), Highway Fuel Economy Test cycle (HWFET), Worldwide harmonized Light duty driving Test Cycle (WLTC) and actual on-road driving conditions. TCN performed the best among the tested models and was easier to build than physics-based modeling. For validating the two different approaches, the physical properties of a 1 L class passenger car with an electric control valve are measured. The NN model proved to be effective in predicting the characteristics of a vehicle cooling system. The proposed method will reduce research costs in the field of predictive control and VTMS design.

Keywords: neural network; recurrent neural network; convolutional neural network; temporal convolutional network; deep learning; time series forecasting; vehicle integrated thermal management system; electric control valve; physical modeling; cooling system

1. Introduction

Vehicle integrated thermal management system (VTMS) technologies have demonstrated their potential in a variety of fields, including powertrains, electrical systems such as motors and batteries, passenger comfort systems, and implementation of new powertrain technologies and emission control systems [1,2]. Among the numerous technologies related to improving fuel economy, when efficiency and improvement in fuel economy are considered compared to the increase in production cost of a vehicle, VTMS technologies are expected to cost less than $50 per 1% reduction in fuel consumption [3]; therefore, automakers are showing significant interest in the technology and are actively applying it to their products [4].

Despite the interests of manufacturers and the demands of the market, there are several challenges in the application of this technology. For optimal operation of VTMS, it is necessary to analyze and predict the system behavior. However, adding sensors for system behavior analysis increases the cost. Therefore, manufacturers use several methods to minimize the number of sensors required. For example, CFD-based analysis for prediction [5,6], simplified model-based prediction [7,8], and model-based analysis for optimization by relocating the sensor [9] have been utilized.

Modeling a thermal management system requires considerable time for experimental parameter studies, as well as manpower in various fields, such as heat transfer, combustion engineering, and fluid mechanics, to construct the model and expensive systems such as environmental chambers. This trend will become more complex with the gradual evolution of thermal management systems.

The vehicle's control system requires a heavy software. Passenger vehicles use four times as many lines of code as a commercial aircraft [10]. Software development costs have gradually increased and have risen to the same level as hardware development costs [11]. While model-based software development is being implemented to reduce development time and cost [3], automakers are still looking for opportunities to increase the flexibility of their workforce or gradually reduce the use of their own resources [12,13]. Along with this trend, there is a need for a simplified development method that can further reduce time, manpower, and costs from conventional thermal management system modeling techniques.

The physics-based method is conventionally used for predicting the energy flow. However, it requires several experimental approaches to obtain the required parameters; thus, the cost is high. Neural networks have been proposed to address this limitation of physics-based modeling.

Instead of figuring out the laws of physics as in the physics-based prediction method, the neural network (NN) method determines the causal relationship between the input and the output. Hence, the intermediate experimental parameter study processes can be omitted. The NN method has excellent generalization performance and shows robust results even with noisy or incomplete data, and is also durable against model defects [14–17]. Since the NN method has many processing nodes, defects in several nodes or connections do not cause serious defects in the entire system [18,19]. Therefore, it is suitable for complex models with many variables and auto-regression problems [20]. It can also be a powerful solution to the problem of coupling data with a variety of inputs [21].

VTMS modeling can be defined as a multivariate time series forecasting problem. Traditionally, time series prediction has been dominated by linear methods such as autoregressive integrated moving average (ARIMA) and vector auto-regression (VAR), which are intuitive for variant problems [22]. However, these classical methods have several limitations, such as problems when data are missing or corrupted and difficulties with multi-step prediction [23]. Furthermore, they predict from a linear relationship to a generalized relationship rather than a complex relationship and, because of the temporal dependence problem, it is necessary to diagnose and specify the number of delayed observations.

Multilayer perceptron (MLP) can overcome some of the disadvantages of ARIMA and VAR. The same simple neural network approximates a mapping function from an input variable to an output variable. This general capability is useful for time series because it can solve various problems such as noise, non-linearity, multivariate input, multi-step inputs, and lack of robustness. Feedforward neural networks do offer great capability but still have the limitation of having to specify the temporal dependence during the model design step [24].

In recent time series prediction research, several studies have proposed variants of recurrent neural network (RNN) and convolutional neural network (CNN) [25]. CNN learns by automatically extracting features from raw data using a method known as expression learning. CNN guarantees shift and distortion invariance with local acceptance, shared or replication weight, and spatial and temporal subsampling [26,27]. These CNN characteristics are useful for preprocessing in time series prediction [28].

RNN is a dedicated sequence model that keeps hidden activation vectors propagating through time [27] (pp. 3–7). Long short-term memory (LSTM), a variant of RNN, explicitly handles the ordering between observations when learning the input-to-output mapping function not provided by MLP or CNN. LSTM natively supports sequences and the persistence of state can learn temporal dependence.

LSTM networks eliminate the need for predefined time windows and can accurately model complex multivariate sequences [29].

The most recent studies present a modified architecture of RNNs and CNNs. Convolutional long short-term memory (ConvLSTM) was developed for reading two-dimensional spatial-temporal data [30], but it can be adapted for use with multivariate time series forecasting [27] (pp. 367–393). Temporal convolutional network (TCN) is more robust in learning and memory conservation than reference iterative architectures such as LSTM, GRU, and RNN in a wide range of sequence modeling tasks [31]. TCN is an architecture that was developed for video-based action segmentation in 2017 [32] and the scope of its application has started to expand recently. The tasks in which it has been applied so far have been limited to areas such as traffic flow forecasting [31], the text to speech (TTS) field represented by Google Deepmind's Wavenet architecture [33], and a wide range of areas with variant transform models.

Although time series forecasting for temperature using NNs is widely used in many other fields, attempts to approach NNs are rare in the VTMS field. The reason for this is that several VTMS devices are being applied to commercial vehicles in recent years and are still expanding [34], and the existing physics-based modeling has not yet emphasized the need for other methods. Most of the NN studies are only partially applied to the state of charge [35,36] or temperature-related studies [37] of battery electric vehicles (BEVs), and their application to other VTMS fields such as cooling system has not been considered.

In this study, the cooling system of a 1 L class gasoline vehicle equipped with an electric control valve (ECV) was modeled using physics-based modeling and NN approach, and the prediction results were compared. Through this comparison, we validate the applicability of NN modeling.

2. Physics-Based Modeling

There are various methods of physics-based modeling. We use the model-based method, a simplified model that can be embedded in a controller for establishing a control strategy or for model-based predictive control.

2.1. Model Structure

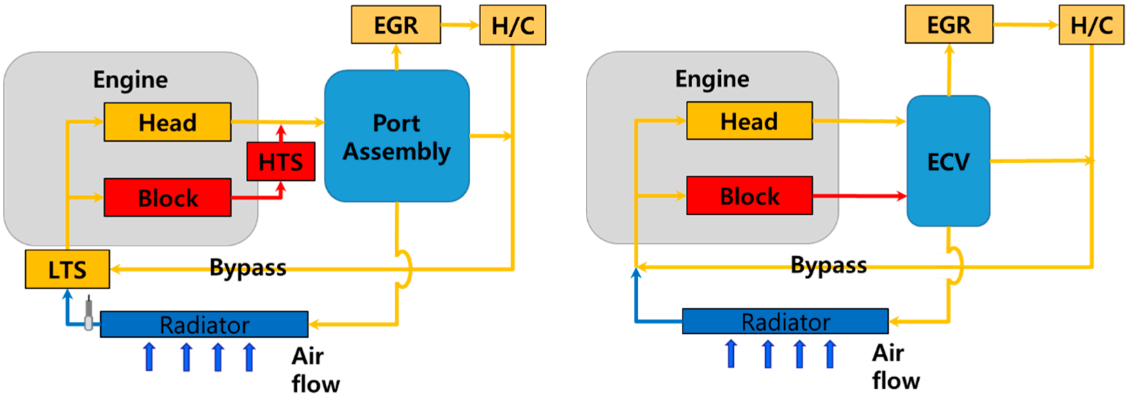
In this study, a small 1 L gasoline vehicle with separate cooling was selected as the target vehicle. Table 1 is a brief specification of the vehicle used in the study. The manual transmission vehicle is equipped with a 98 horsepower gasoline engine, mechanical thermostat, and mechanical coolant pump. However, the mechanical thermostat was replaced with an ECV.

Table 1. Specifications of the vehicle used in the experiment.

Powertrain	**Engine**	Displacement (cc)	998
		Max Power (PS/RPM)	77/6500
		Max Torque (kg·m/RPM)	9.7/4800
		Cooling System	Thermostat ECV
		Injector	Solenoid
		Water Pump type	Mechanical
		Engine Oil	0W-20
	Transmission	Number of Gear Stages	5th
		FGR×TOP Gear Ratio	3.3074
		ATF Warmer/Cooler	X
Weight		ETW (kg)	1215

ECV is an electrically operated valve that replaces the thermostat to measure the temperature and ensure coolant flow to the desired part. Because mechanical thermostats operate only at a set temperature by opening the valve using the force of the expansion of the wax inside, we used ECV that can variably determine the coolant flow rate according to the desired control temperature and strategy.

Figure 1 shows the schematic of the vehicle cooling system used in this study. The cooling system of a car works as follows. The heat generated by fuel combustion in the engine is cooled by the radiator to keep the engine oil at a high temperature to maintain optimum viscosity. It also regulates the coolant temperature to optimize the operation of emission control devices such as exhaust gas recirculation (EGR). The coolant from EGR is then passed through the heater core (H/C) to heat the air supplied to the vehicle interior, because the heating, ventilation, and air conditioning (HVAC) was not operated, and the influencing factor was ignored in this study.



(**a**) (**b**)

Figure 1. Cooling system schematic comparison. (**a**) Conventional cooling circuit. Conventional cooling system using a mechanical thermostat with separate cooling. Low-temperature Thermostat (LTS) starts opening at 88 °C; high-temperature thermostat (HTS) starts opening at 105 °C. (**b**). Modified cooling circuit. Electric control valve system with both the mechanical thermostats (LTS, HTS) removed. The flow through Exhaust gas recirculation (EGR) and Heater core (H/C) is combined in the bypass.

HVAC is based on the premise that it does not work according to regulations in mode tests (FTP-75, HWFET, WLTC) to measure fuel economy. Therefore, the mode results used in the verification data in this study are not related to the HVAC operation.

The vehicle used in this study has a manual transmission system. In general, manual transmissions do not have a heat exchanger for oil transmission so that modeling can be done in terms of coolant and engine oil.

According to previous studies, energy in vehicles is distributed as follows: About 25–28% of the heat generated inside the engine is transferred to power the spark ignition engine; 17–26% is transferred to the cooling system, and 3–10% in cooling the heat rejected or convection with oil. The rest is exhausted in the form of thermal energy or kinetic energy through the exhaust or converted into enthalpy loss [38]. This is expressed as an energy balance equation as follows:

$$\eta_c \dot{m}_f Q_{LHV} = \dot{Q}_{cool} + \dot{W}_b + \dot{Q}_{oil} + \dot{H}_{exhaust} + \dot{Q}_{misc} \tag{1}$$

Here the expression on the left refers to the energy generated by the fuel combustion. η_c is the combustion efficiency of fuel; $\dot{m}_f$ is the fuel consumption; Q_{LHV} is the low heating value of gasoline. The expression on the right refers to the energy conversion into engine heat transfer rate to the coolant $\dot{Q}_{cool}$, brake power $\dot{W}_b$, exhaust energy loss $\dot{H}_{exhaust}$, engine oil heat transfer $\dot{Q}_{oil}$, and other minor miscellaneous losses $\dot{Q}_{misc}$.

2.1.1. Engine Oil Temperature

The engine oil bulk temperature in oil pan can be simulated by the effect of heat generated inside the engine by fuel combustion ($\dot{Q}_{oil}$), heat exchange with coolant to cool it at engine oil cooler ($\dot{Q}_{EOC,oil}$), and heat exchange with ambient at oil pan and oil filter ($\dot{Q}_{air}$). The energy flow in the engine oil can be expressed as an energy balance equation as follows:

$$\frac{dU_{oil}}{dt} = \dot{Q}_{oil} + \dot{Q}_{EOC,oil} + \dot{Q}_{air} \tag{2}$$

Each factor in Equation (2) is expressed in detail by the heat transfer equation as follows:

$$(mc)_{oil}\frac{dT_{oil}}{dt} = k_{oil} \times P_{gen} + \left(\dot{m}c\right)_{EOC,oil}\Delta T_{EOC,oil} + (\mathrm{hA})_{air}(T_{air} - T_{oil}) \tag{3}$$

Some of the heat generated by combustion and mechanical friction in the cylinder contributes to the temperature rise in the coolant and engine oil. The amount of heat transferred by combustion inside the engine, which is a turbulent flow, should be analyzed for combustion. Rather than a complicated analysis, it is assumed that the ratio of the transfer function k_{oil} due to engine heat generation P_{gen} contributes to the temperature rise in coolant and engine oil, according to Equation (1) [39].

To use Equation (3), both the inlet/outlet temperature when passing through the heat exchanger (engine oil cooler) must be known, but this measurement is difficult under actual conditions. Therefore, in this study, the model was modified using the Effectiveness-NTU method, which expresses effectiveness ε as

$$\varepsilon = \frac{C_h\left(T_{h,in} - T_{h,out}\right)}{C_{min}\left(T_{h,in} - T_{c,in}\right)}\ or,\ \varepsilon = \frac{C_c(T_{c,in} - T_{c,out})}{C_{min}\left(T_{h,in} - T_{c,in}\right)} \tag{4}$$

Equation (3) can be expressed as follows by Equation (4).

$$(mc)_{oil}\frac{dT_{oil}}{dt} = k_{oil} \times P_{gen} + \varepsilon_{EOC,oil} \times C_{min}\Delta T_{EOC,oil} + (\mathrm{hA})_{air}(T_{air} - T_{oil}) \tag{5}$$

$$T_{oil} = \int \frac{k_{oil} \times P_{gen} + \varepsilon_{EOC,oil} \times C_{min}\Delta T_{EOC,oil} + (hA)_{air}(T_{air} - T_{oil})}{(mc)_{oil}}dt \tag{6}$$

2.1.2. Coolant Temperature

The behavior of coolant temperature is similar to that of engine oil but is more complex due to the influence of ECV, radiator, and bypass. However, when several variables are considered, the parameter determination becomes very complex and difficult; thus, the factors with minority influence were ignored. As shown in Equation (2), the factors that affect the coolant temperature can be expressed as:

$$\dot{Q}_{EOC,oil} = -\dot{Q}_{EOC,coolant} \tag{7}$$

$$\frac{dU_{coolant}}{dt} = \dot{Q}_{cool} + \dot{Q}_{EOC,coolant} + \dot{Q}_{rad} + \dot{Q}_{EGR} + \dot{Q}_{H/C} + \dot{Q}_{bypass} \tag{8}$$

One of the most difficult parts in expressing the coolant temperature behavior is expressing the heat exchanger of the radiator. The radiator can be expressed using the NTU-effectiveness method, similar to a general heat exchanger, but it is difficult to measure the coolant and ambient flow rate. The effectiveness of the radiator ε_{rad}, expressed in Equation (4), can also be expressed as a function of temperature because the ambient temperature is colder than the coolant temperature.

$$\varepsilon_{rad} = \frac{C_{coolant}\left(T_{coolant,in} - T_{coolant,out}\right)}{C_{min}\left(T_{coolant,in} - T_{air,in}\right)} \tag{9}$$

The coolant flow rate is affected by the ECV's opening area in the radiator direction and the operating speed of the water pump (N_{pump}). Since the water pump is directly connected to the engine's shaft, its operation is proportional to the engine speed (N). The opening area on the radiator side of the ECV $(A_{ecv,rad})$ is proportional to the defined valve shape and angle of the ECV motor (θ_{ecv}) which is an actuator that operates the valve. Therefore, the flow rate of coolant can be expressed as a function of the ECV's operating angle and engine speed.

$$\dot{m}_{rad,coolant} = f\left(A_{ecv,rad}, N_{pump}\right) = f(\theta_{ecv}, N) \tag{10}$$

The flow rate of air reaching the front of the radiator can be expressed as the sum of the flow rate that reaches the radiator when the vehicle is running and flow rate generated when the radiator electric fan operates. It can be assumed that the running wind is proportional to the vehicle speed $(vspd)$, and the electric fan generates a flow rate proportional to the amount of current supplied $(I_{rad,fan})$.

$$\dot{m}_{rad,air} = f\left(vspd,\ I_{rad,fan}\right) \tag{11}$$

The equation related to coolant can also be summarized as follows:

$$(mc)_{coolant}\frac{dT_{coolant}}{dt} = \begin{bmatrix} k_{coolant} \times P_{gen} + \varepsilon_{EOC,coolant} \times C_{min}\Delta T_{EOC,coolant} \\ +\varepsilon_{rad} \times C_{min,rad}\Delta T_{rad} + \varepsilon_{EGR} \times C_{min,EGR}\Delta T_{EGR} \\ +\varepsilon_{H/C} \times C_{min,H/C}\Delta T_{H/C} + (\mathrm{hA})_{air}\left(T_{air} - T_{bypass}\right) \end{bmatrix} \tag{12}$$

$$T_{coolant} = \int \begin{bmatrix} k_{coolant} \times P_{gen} + \varepsilon_{EOC,coolant} \times C_{min}\Delta T_{EOC,coolant} \\ +\varepsilon_{rad} \times C_{min,rad}\Delta T_{rad} + \varepsilon_{EGR} \times C_{min,EGR}\Delta T_{EGR} \\ +\varepsilon_{\frac{H}{C}} \times C_{min,H/C}\Delta T_{\frac{H}{C}} + (\mathrm{hA})_{air}\left(T_{air} - T_{bypass}\right) \end{bmatrix} \frac{dt}{(mc)_{coolant}} \tag{13}$$

Using these equations, we design the simulation to predict the coolant and engine oil temperatures.

2.2. Experimental Parameter Study

Physics-based modeling requires identifying the parameters of the simulation. Each part must be individually tested to study the experimental parameters. Therefore, in this section, we describe examples of a single unit experimental process for parameter selection of the key factors.

The radiator is the most influential part of the vehicle cooling system. The factors that affect the cooling amount of the radiator are the temperature and flow rate of the coolant, temperature of the surrounding air, flow rate (in proportion to the vehicle running speed), and cooling fan operation.

For experimenting with such a radiator, an environmental chamber is required that can determine the ambient temperature and flow rate. Figure 2 shows the system and measurement device that simulates the cooling system located inside an environmental chamber. The engine, replaced by a heater, determines the radiator inlet coolant temperature, and the external turbofan determines the air flow rate. The air conditioner inside the chamber determines the ambient temperature, and the electric water pump determines the coolant flow rate. Based on the NTU-effectiveness method, the heat exchange in the radiator can be calculated by changing the conditions in response to changes in inlet and outlet temperature and flow rate.

Calculating the engine's heat generation requires more complex tests. Figure 3 shows the actual vehicle engine installed in the engine cell. The engine cell contains a device to control the temperature of the coolant; unlike a real engine, the coolant temperature is controlled through this device instead of a radiator.

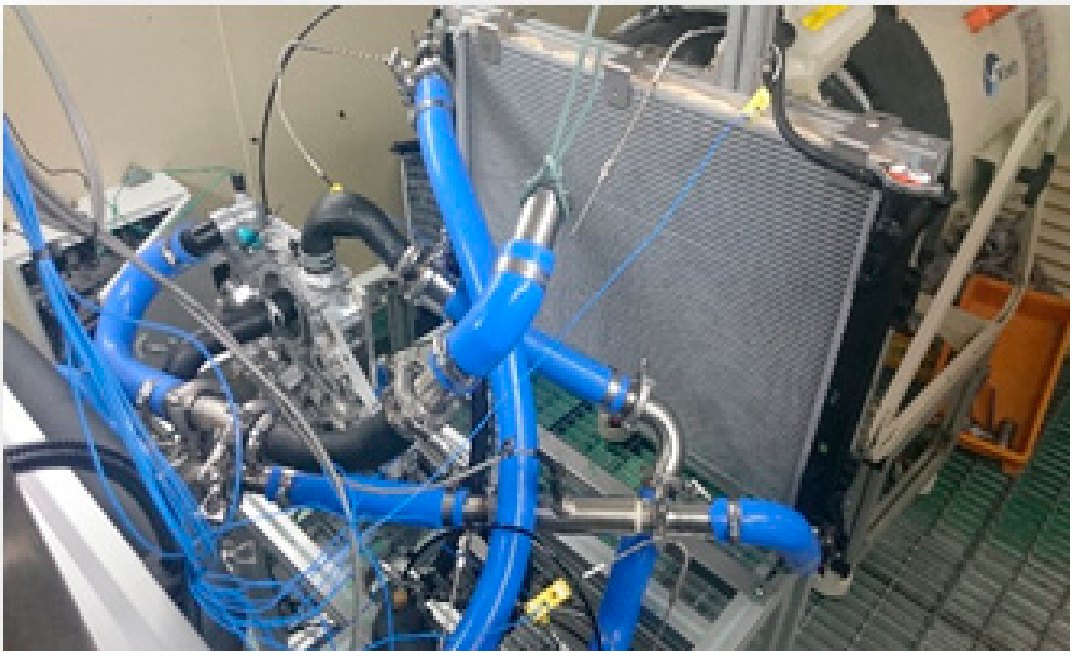


Figure 2. Measurement device that simulates the cooling system located inside the environmental chamber. It is configured to control the external temperature and amount of air blown to the front of the radiator, and to control the coolant temperature and flow rate supplied to the radiator.

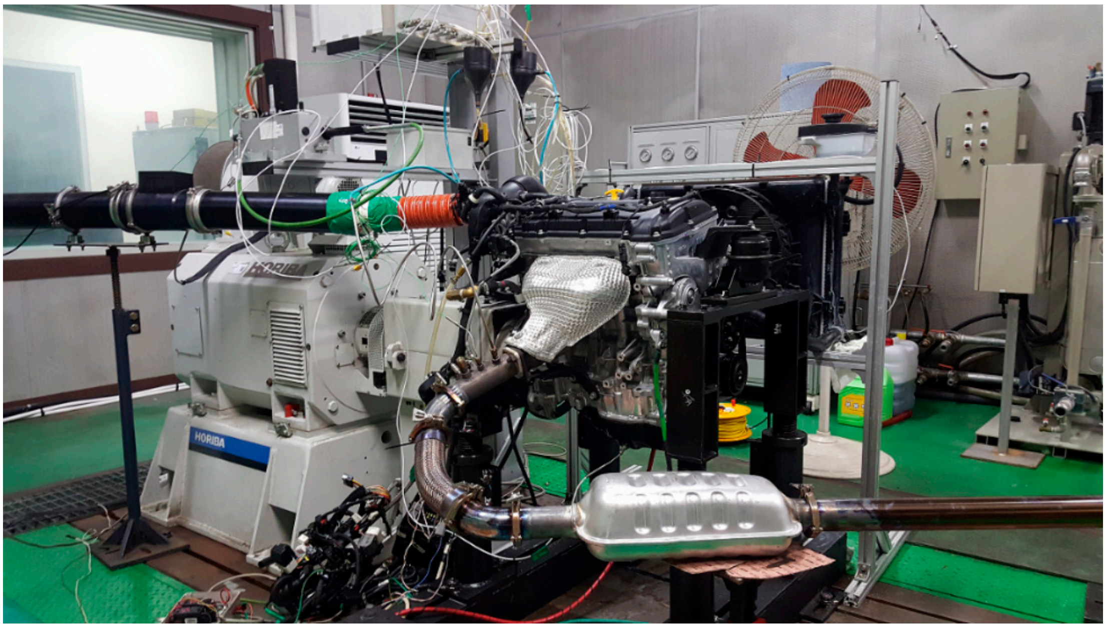


Figure 3. Engine cell test system. Engine generated heat is calculated under various conditions, and the difference in temperature before and after the coolant can be measured.

The heat generated by the engine is affected by many factors, including combustion and turbulent flow. Therefore, it is difficult to predict accurately. In this study, the heat generated by the engine is based on the assumption that a certain portion of the indicated work is transferred to the coolant. The engine measured the operating point according to the engine speed and load (brake torque), and the experiments were conducted while maintaining the steady-state to reduce the influence of control variables such as ignition timing. The heat generated by the engine was calculated by measuring the difference between the coolant temperature and the inlet and outlet flow rate of the engine.

In addition to the experiments mentioned above, physics-based modeling requires further experiments such as the change in coolant flow rate according to the opening amount of the ECV, and engine speed, heat dissipation from each pipe to the atmosphere, heat exchange in EGR, and heat exchange between oil and coolant.

Table 2 lists some typical parameter values that need to be obtained from experimental parameter studies to complete the physics-based model. There is no need to obtain these values in the neural network technique.

Table 2. Parameters necessary for physical models.

	Parameter	Description
#1	$h_{radiator}$	Heat transfer coefficient of radiator
#2	$h_{oil\ pan}$	Heat transfer coefficient of oil pan
#3	$h_{engine\ block}$	Heat transfer coefficient of engine block
#4	$k_{engine\ block}$	Thermal conductivity of engine block
#5	$k_{oil\text{-}coolant}$	Thermal conductivity of wall surface where heat exchange occurs between engine oil and coolant
#6	$C_{p,engine\ oil}$	Specific heat capacity of engine oil
#7	$C_{p,coolant}$	Specific heat capacity of coolant
#8	$C_{p,engine\ crank\ shaft}$	Specific heat capacity of engine crank shaft
#9	$C_{p,engine\ block}$	Specific heat capacity of engine block
#10	$C_{p,radiator}$	Specific heat capacity of radiator
#12	$A_{radiator}$	Surface area where convective heat transfer of radiator occurs
#12	$A_{oil\ pan}$	Surface area where convective heat transfer of oil pan occurs
#13	$A_{engine\ block}$	Surface area where convective heat transfer of engine block occurs
#14	L_{bolck}	Wall thickness where heat conduction occurs between coolant and engine block
#15	$L_{oil\text{-}coolant}$	Wall thickness where heat conduction occurs between coolant and engine oil
#16	$\varepsilon_{radiator}$	Effectiveness for predicting radiator outlet temperature
#17	ε_{EOC}	Effectiveness for engine oil cooler
#18	η_{th}	Efficiency at which fuel is converted from fuel energy to indicated power when internal combustion engine is operated
#19	$\eta_{engine\ heat\ transfer}$	Ratio of the amount of heat transferred to cooling system from heat loss
#20	$A_{oil\text{-}coolant}$	Area of heat conduction between oil and coolant at oil cooler
#21	$A_{engine\ block\text{-}coolant}$	Area of heat conduction between engine block and coolant

2.3. Simulation Implementation of Coolant and Oil Temperature

We used MATLAB/Simulink® (MathWorks Inc., Natick, MA, USA) to perform the simulations in this study. The system was divided into several sub-systems, each simulating the behavior of the cooling system through the same flow path and calculation structure as the components in the actual vehicle, as shown in Figure 4:

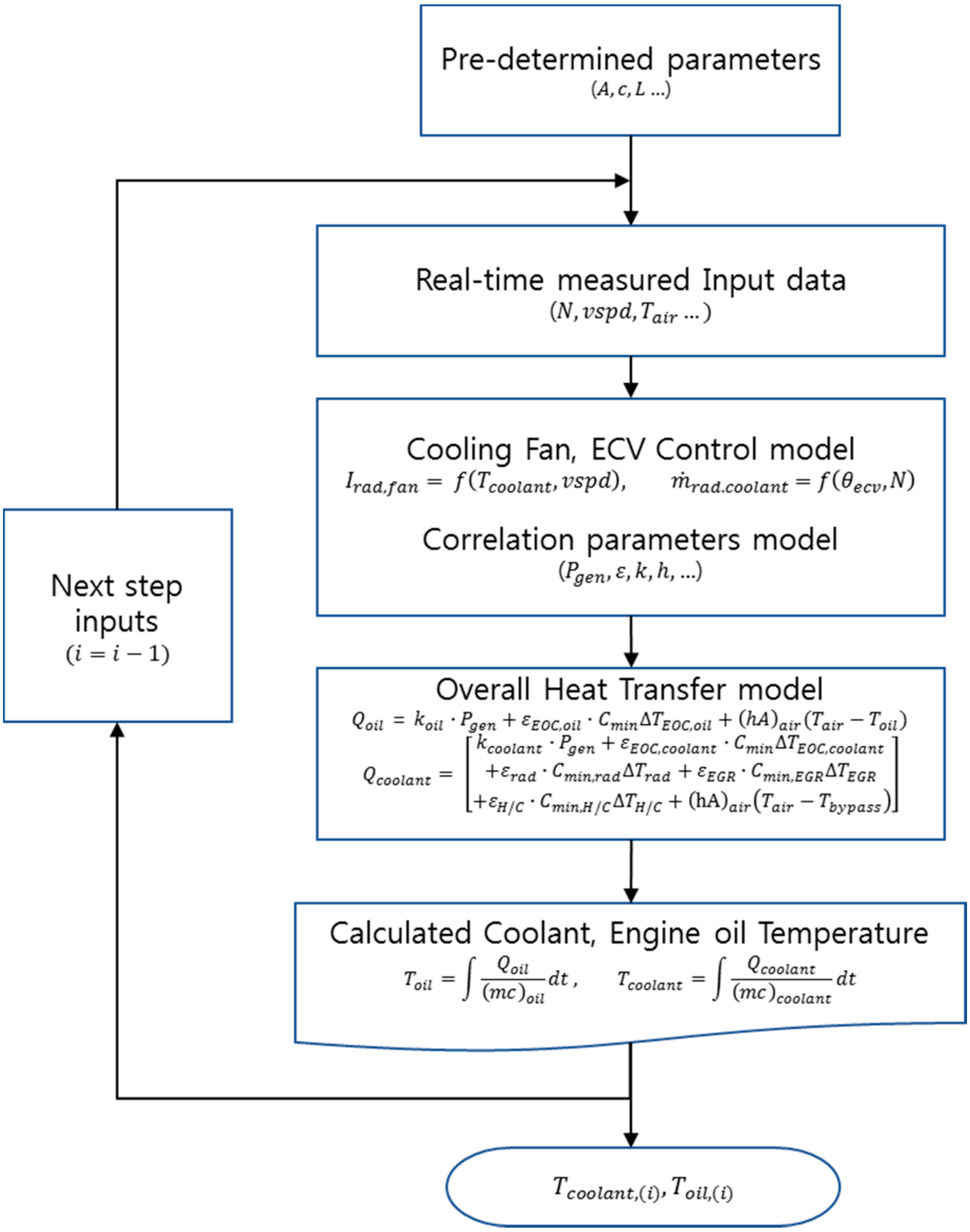

Pre-determined parameters
$(A, c, L \ldots)$
Real-time measured Input data
$(N, vspd, T_{air} \ldots)$
Cooling Fan, ECV Control model
$I_{rad,fan} = f(T_{coolant}, vspd)$, $\dot{m}_{rad.coolant} = f(\theta_{ecv}, N)$
Correlation parameters model
$(P_{gen}, \varepsilon, k, h, \ldots)$
Overall Heat Transfer model
$Q_{oil} = k_{oil} \cdot P_{gen} + \varepsilon_{EOC,oil} \cdot C_{min} \Delta T_{EOC,oil} + (hA)_{air}(T_{air} - T_{oil})$
$Q_{coolant} = \begin{bmatrix} k_{coolant} \cdot P_{gen} + \varepsilon_{EOC,coolant} \cdot C_{min} \Delta T_{EOC,coolant} \\ + \varepsilon_{rad} \cdot C_{min,rad} \Delta T_{rad} + \varepsilon_{EGR} \cdot C_{min,EGR} \Delta T_{EGR} \\ + \varepsilon_{H/C} \cdot C_{min,H/C} \Delta T_{H/C} + (\mathrm{hA})_{air}(T_{air} - T_{bypass}) \end{bmatrix}$
Calculated Coolant, Engine oil Temperature
$T_{oil} = \int \frac{Q_{oil}}{(mc)_{oil}} dt$, $T_{coolant} = \int \frac{Q_{coolant}}{(mc)_{coolant}} dt$
Next step inputs
$(i = i - 1)$
$T_{coolant,(i)}, T_{oil,(i)}$

Figure 4. Flowchart of the computational simulation for prediction by physics-based model.

1. Pre-determined parameters: among the values obtained from experimental parameter studies, a fixed variable that is not a value that changes in real time. It consists of the surface area (A), specific heat (c), and thickness of the cylinder wall (L) of various heat exchangers.
2. Real-time measured input data: input data that changes every moment while the actual vehicle is driving. It is obtained by data acquisition device, and consists of engine speed (N), vehicle speed ($vspd$), ambient temperature ($T_{ambient}$), etc.
3. Cooling fan, ECV control model: a model that calculates the control target value using real-time input data to determine what effect it will have if the control logic is changed, or to verify that the existing control is working well.

4. Correlation parameter model: a model that calculates parameters that change according to the value of real-time input data. Such as Heat transfer coefficient (h), Effectiveness (ϵ), power generation (P_{gen}), etc. are calculated.
5. Overall heat transfer model: a model that integrates the amount of heat transferred by each unit to calculate the temperature of engine oil and coolant, which are the prediction targets.
6. Calculation of coolant temperature and engine oil temperature: a model that predicts the next stage temperature by using the heat quantity calculated in the integrated heat transfer model.
7. Next step inputs: send the final calculated output to the next step input.

The completed simulation predicts coolant and engine oil temperatures under given operating conditions. The results of this step, including coolant and engine oil temperatures, form the inputs of the next step.

3. Neural Network-Based Modeling

The Neural Network method directly learns the relationship between the input and the output. Therefore, the time-consuming study process in most physics-based modeling can be eliminated, and collecting vehicle data should be sufficient to obtain necessary input for the NN models.

The main input and output elements used for prediction of both physics-based and neural network models are shown in Table 3:

Table 3. Main input and output elements for physics-based and neural network (NN) models.

	Parameter	Description	I/O Classification
#1	Vehicle speed	Vehicle speed for calculating wind speed	Input
#2	Engine speed	Engine rotating speed	Input
#3	Ambient temperature	Air temperature outside the vehicle	Input
#4	Indicated torque	Indicated torque/maximum torque ratio	Input
#5	ECV current angle	Angle at which ECV is currently operating	Input
#6	Radiator cooling fan current	Electric current for cooling fan operation (A)	Input
#7	Sampling rate	Data acquisition interval time	Input
#8	Coolant temperature	Predicted coolant temperature	Output
#9	Engine oil temperature	Predicted engine oil temperature	Output

Figure 5 is a graph showing some of the training data for a neural network-based model. Some of the input and output signals presented in Table 3 were acquired by mounting only the Controller Area Network (CAN) and a simple temperature sensor. The learning data were obtained by driving repeatedly on the actual road. The acquired data were obtained through repetitive operations by non-professional personnel. The acquired data were used for training and verification via several neural network models.

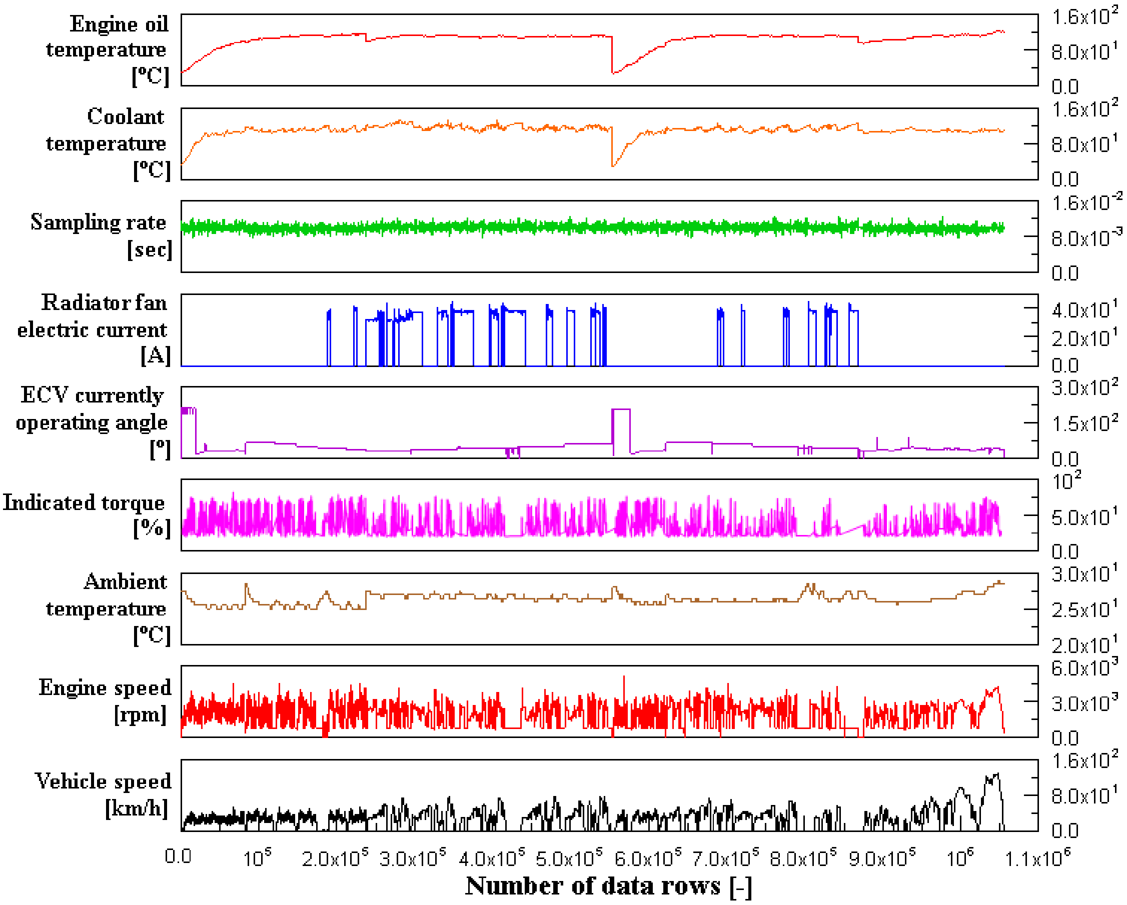


Figure 5. Graph representing samples of the training data for neural network-based model.

3.1. Model Framework Determination

The VTMS modeling problem can be defined as a time series forecasting problem. The model calculates the desired output by inferring the sequence and physical association of several measured input variables. Traditionally, a variety of methods for predicting time-series have been developed, ranging from statistically-based modeling methods such as ARIMA, VAR, and exponential smoothing to the more recent NN deformation models.

The NN model used in this study aims to minimize the type of input variable. Sophisticated and complex models guarantee high accuracy, but it is difficult to collect input data. This study aims to set up an unsophisticated predictive model by using known physical causal relationships as inputs, rather than including elements with subtle influences. Hence, this problem can be defined as a problem with endogenous variables.

The input variable used is obtained from the Controller Area Network (CAN) given by the vehicle, assuming that the installation of additional sensors is avoided as much as possible. The output variable uses engine oil and coolant temperatures as in physics-based modeling. However, due to resolution problems, the temperature of the coolant is measured using the installed thermocouple.

The given problem is a typical regression problem and corresponds to time series forecasting. In addition, since the objective is to find a physical correlation, it is an unstructured problem (without seasonality) and a multivariate problem that predicts two outputs by finding correlations from multiple input variables. Following recent trends, we applied ConvLSTM—a variant of RNN, and TCN—a variant of CNN, as the NN architecture that fits the study characteristics.

To assess the model's prediction efficiency, the differences between the actual and predicted model values were compared statistically using two performance metrics: mean absolute error (MAE) and mean square error (MSE).

3.2. Convolutional LSTM

RNN has several variants, but recently LSTM, a variant in which a gate structure is added to the RNN to solve the long-term dependency problem, has been gaining popularity. In general, the time series forecasting that requires sequence processing was considered in RNN and its variants [40]. However, recent research reveals the shortcomings of the RNN method [41,42]. There are several challenges in predicting the temperature in this study. First, various operating conditions or control changes affect the temperature, with a time delay of about 3–10 s for this output to change. The data sampling rate is about 10–100 ms, and the change is relatively fast compared to the time delay. Hence, it takes about 1000 or more sequences of data for changes in the input data to affect the output. However, recent studies have shown that RNN variants such as LSTM may not be suitable for training very long sequences of 1000 or more [43]. Hence, we need other methods to deal with long sequence time-series data.

To reduce the cost of LSTM or utilize pretreatment, variants of LSTM are being developed, including using CNN for down-sampling. CNNs have excellent performance in extracting features from large data volumes, and several modified CNN-LSTM techniques have been studied. A further extension of the CNN-LSTM approach is the ConvLSTM that performs convolution of the CNN as part of the LSTM for each time step. Similar to CNN-LSTM, ConvLSTM is used for spatiotemporal data. ConvLSTM was developed for reading two-dimensional space-time data, but can be tuned for use with multivariate time series forecasting [27] (pp. 367–393) and is expressed as follows, where '∘' denotes the Hadamard product [44,45]:

$$\begin{aligned} i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci} \circ c_{t-1} + b_i) \\ f_t &= \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf} \circ c_{t-1} + b_f\right) \\ c_t &= f_t \circ c_{t-1} + i_t \circ \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\ o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co} \circ c_t + b_o) \\ h_t &= o_t \circ \tanh(c_t) \end{aligned} \tag{14}$$

This is quite similar to the LSTM, but with the matrix multiplication replaced with the convolution operations. This means that the number of weights present in every W in each cell can be extremely small than the fully-connected LSTM. This is similar to replacing the fully connected layer with the convolutional layer, thereby reducing the overall number of model weights [46].

The structure of the ConvLSTM is shown in Figure 6. Bi-directional LSTMs consistently perform better than unidirectional layers, reducing the weakness of convolutional LSTMs' vulnerability to overfitting [47]. Therefore, in this study, a layer of bidirectional LSTM was added after the layer of convolutional LSTM.

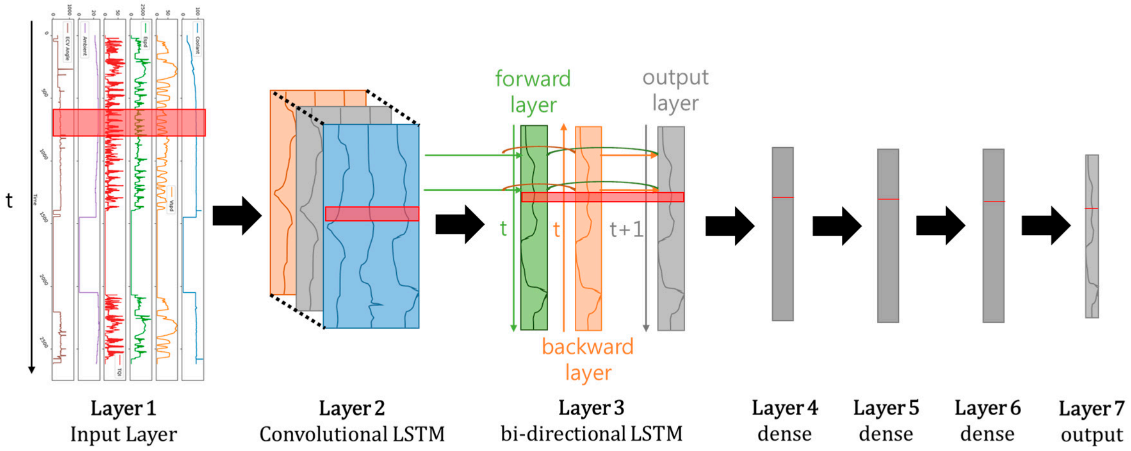


Figure 6. Description of the convolutional long short-term memory (ConvLSTM) architecture. For time-series forecasting, ConvLSTM and bidirectional long short-term memory (LSTM) were stacked.

3.3. *Temporal Convolutional Network*

Temporal Convolutional Network is a specialized architecture developed for time-series forecasting. TCN can extract long-term patterns using dilated causal convolutions and residual blocks, and can also be more efficient in terms of accuracy and calculation speed than variants of LSTMs that are treated as modern architectures in the field [48]. TCN is more robust in learning and memory conservation than reference iterative architectures such as LSTM, GRU, and RNN in a wide range of sequence modeling tasks [30,31].

Unlike RNNs, TCN has no explicit time dependency between predictions for adjacent time steps. Hence, TCN can perform convolutions in parallel to process long inputs as a whole sequence for training and evaluation. In TCN, the sequence information learned in the local layer is propagated to the upper layer through the temporal layer. This is due to the introduction of a temporal convolutional filter layer called a temporal layer structure; because of this characteristic, TCN can capture long sequence patterns [49].

The extended causal convolution used by TCN was more effective in capturing temporal dependencies than repetitive LSTM units. Additionally, TCN was less sensitive to parameter selection than LSTM models, providing more stable performance.

Figure 7 shows the structure of the dilated causal convolutional layers. The complete dilated causal convolution operation over consecutive layers can be formulated as follows [50]:

$$x_l^t = \mathrm{g}\left(\sum_{k=0}^{K-1} w_l^k x_{(l-1)}^{(t-k\times d)} + b_l\right) \tag{15}$$

where x_l^t is the output of the neuron at position; (t) in the layer number l; K is the width of the convolutional kernel; w_l^k stands for the weight of position (k); d is the dilation factor of the convolution, and b_l is the bias term.

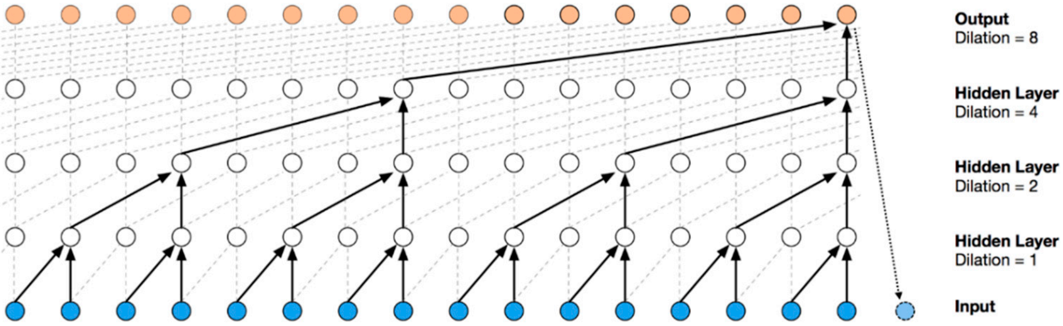


Figure 7. Architecture of a stack of dilated causal convolutional layers [33].

TCN has fewer trainable parameters than RNN but takes longer to train. However, the prediction time after training is much faster than that of conventional RNNs. Despite parameter tuning, large models need not always give good results owing to overfitting. Deep networks lose generalization capacity when trained using large models since they often converge to sharp minimizers; hence, choosing a smaller model can be beneficial. For TCNs, a smaller kernel was able to extract the underlying patterns more accurately [51].

4. Comparison of Predicted Results and Actual Measurements

For verifying the prediction accuracy of the system, we compared the predicted value with the actual values obtained by driving the vehicle under different driving conditions.

Figure 8 shows the actual vehicle installed on the chassis dynamometer to test the fuel efficiency authentication mode. The verification of the entire system consisted of the model experiment, a vehicle fuel economy measurement experiment on chassis dynamometer, and an on-road driving of the vehicle. The vehicle's verification mode was tested with EPA Federal Test Procedure (FTP-75), Highway Fuel Economy Test cycle (HWFET), and Worldwide harmonized Light duty driving Test Cycle (WLTC).

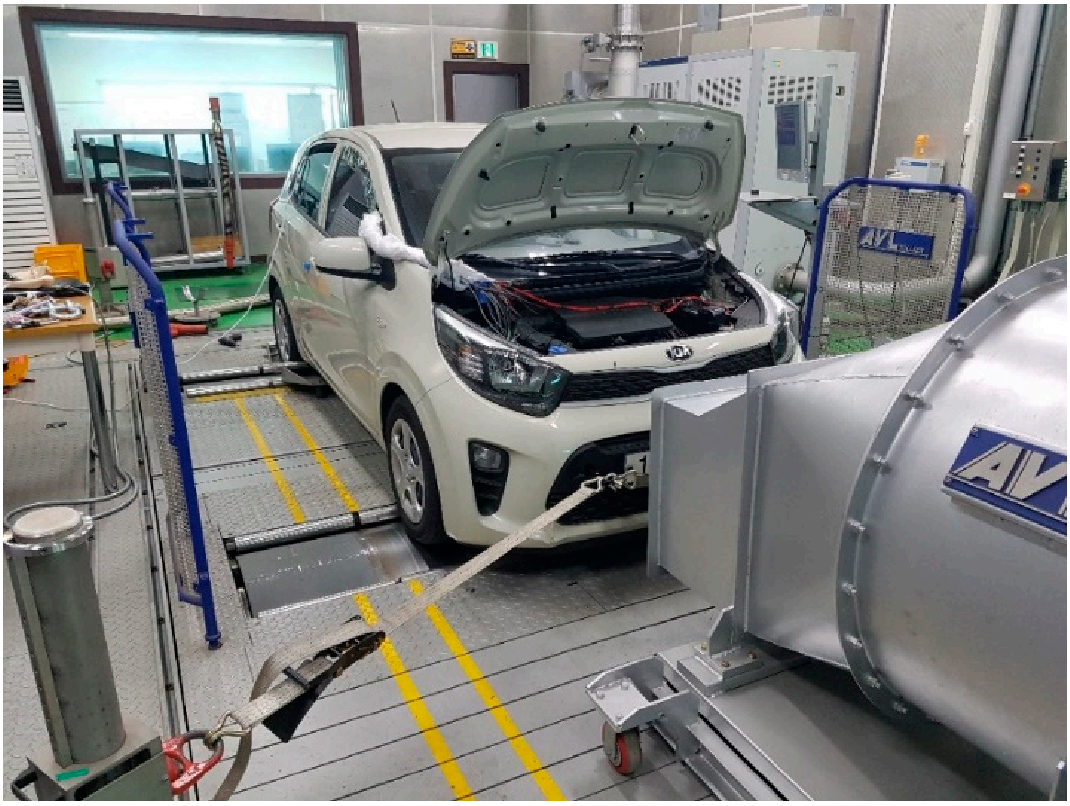


Figure 8. Test on the chassis dynamometer for verification. By comparing the experimental and simulation results the feasibility of the modeling method can be determined.

Most of the chassis dynamometer fuel economy authentication models are good structures to judge whether simulations are well made. The driving pattern of the fuel economy authentication model is designed to reflect the driving conditions of various vehicles that are not monotonous within a short time.

FTP-75 is a model designed to test urban driving conditions. The test procedure measures emissions and fuel economy as defined by the US Environmental Protection Agency (EPA). Phases 1 and 2 of the FTP-75 simulate the traffic conditions during rush hour in downtown LA in 1972 [52].

HWFET is part of FTP-75 and simulates highway driving conditions [53]. Most of them are high-speed areas and do not stop while driving.

WLTC is a European standard driving model [54]. After the low-speed zone that simulates urban driving, the high-speed zone that simulates highway driving appears. This model covers a large driving range, which is good for determining whether predictions are well made in the neural network method.

Actual on-road driving generated random inputs that were not repeated when driving in a laboratory setup. These random driving conditions prevent overfitting in both physics-based and NN approaches and, in some cases, also help to secure additional training data.

5. Results

This section presents and discusses the experimental results of the physics-based modeling and NN approaches presented in the previous section. The predicted results were compared to actual experimental data (ground truth) to determine their suitability.

5.1. Comparison Between Prediction and Actual Measurement

The comparison between the simulated prediction results and actual data in different driving conditions is shown in Figure 9.

Physics-based simulation results show substantially similar behavior to actual measured data under similar operating conditions. Using physical-based modeling, the vehicle performance can be predicted even in areas or conditions that are otherwise difficult to experiment with real vehicles. It can be used to establish a control strategy such as model-based predictive control of controllable VTMS devices.

Table 4 lists some of the parameter tuning values for optimizing ConvLSTM. The activation functions ReLU and SoftPlus show excellent performance in terms of accuracy [52]. In all cases, the dropout and recurrent dropout were set to 0.2. Based on the results, Model #9 is determined to have the least error and better computational speed.

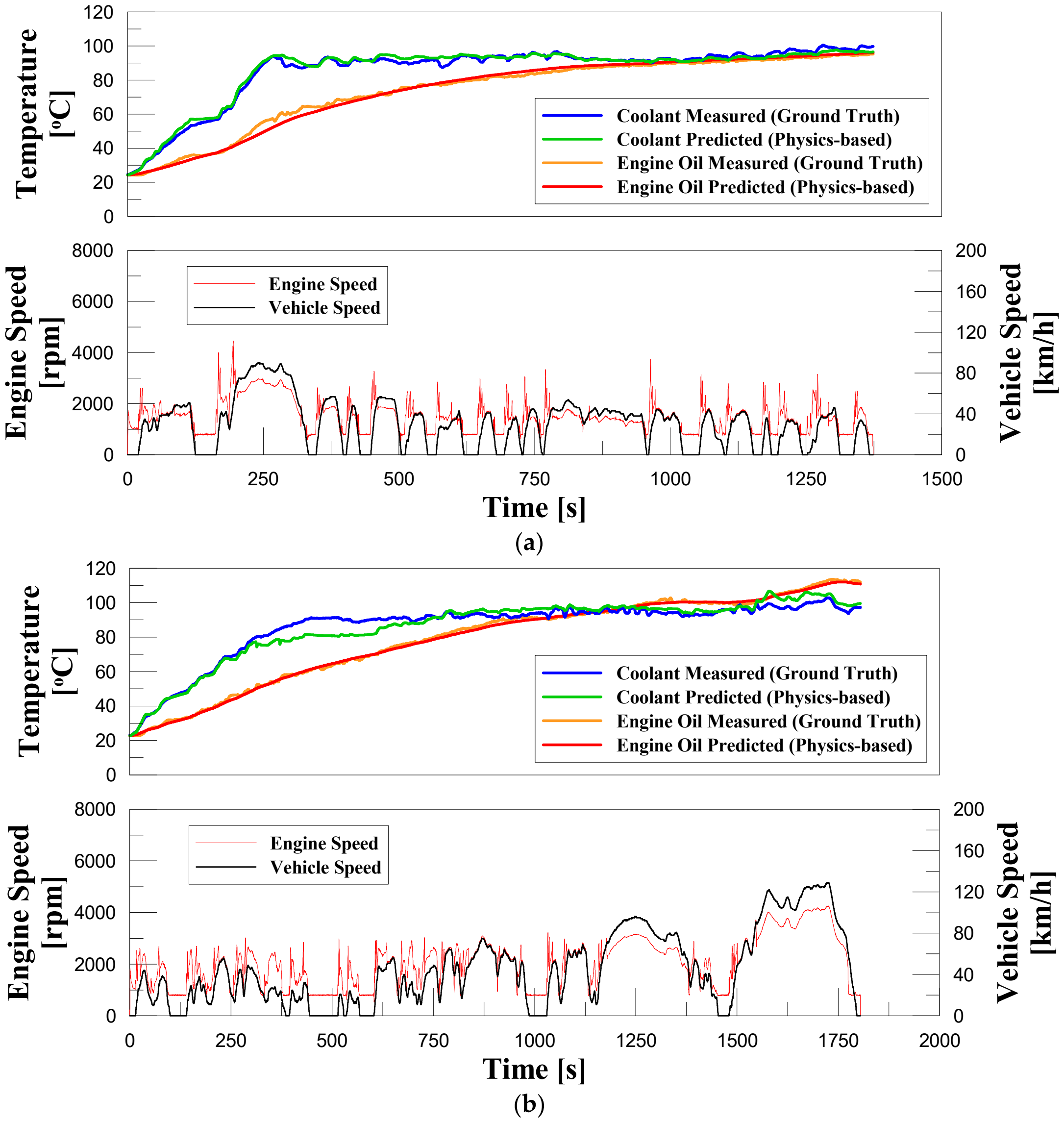
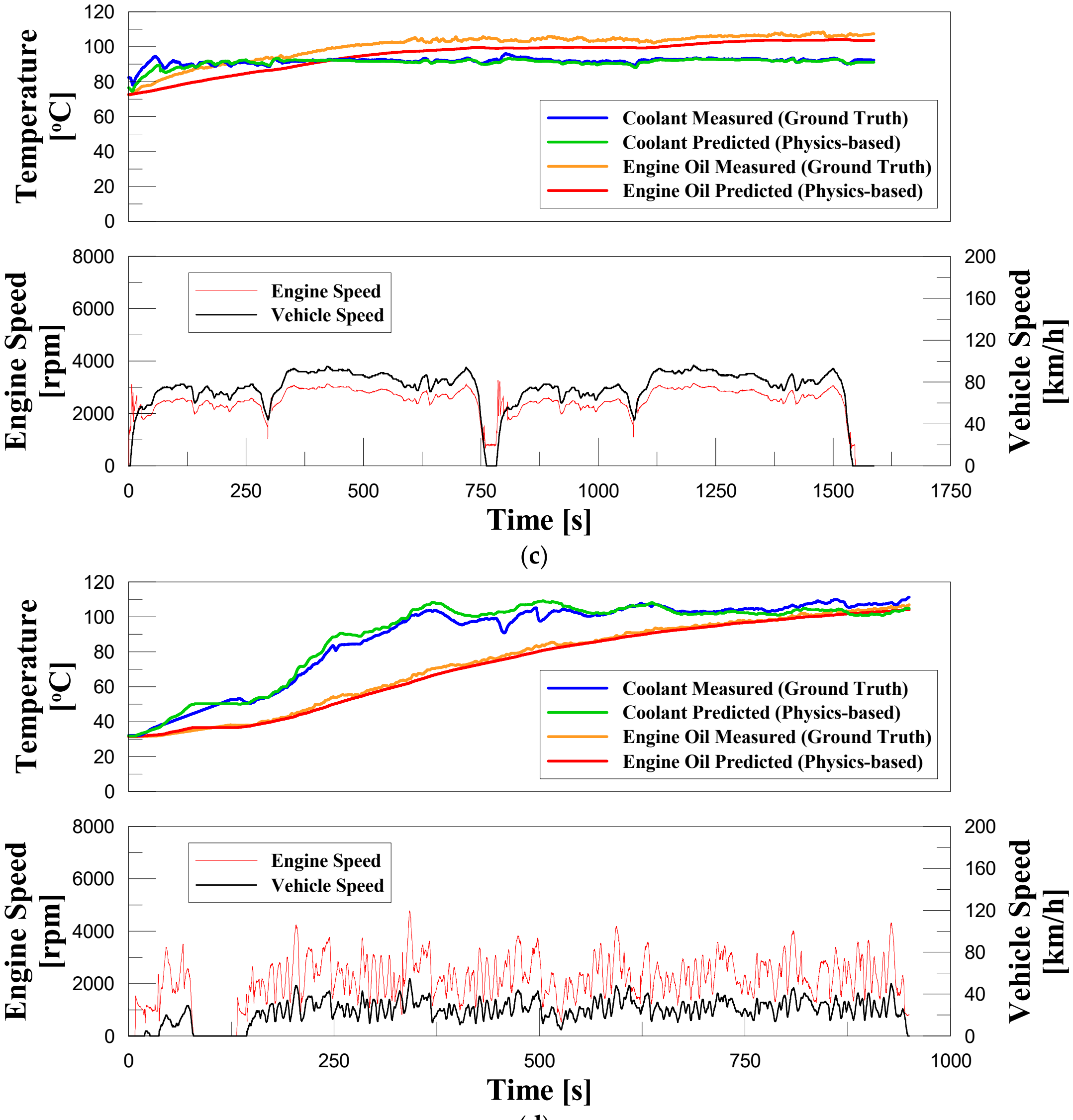


Figure 9. *Cont.*



(**c**)

(**d**)

Figure 9. Coolant and engine oil temperatures for various driving conditions predicted by physics-based modeling. (**a**) EPA Federal Test Procedure (FTP-75) Phase 1,2; (**b**) Worldwide harmonized Light duty driving Test Cycle (WLTC); (**c**) Highway Fuel Economy Test cycle (HWFET); (**d**) On-road test.

For comparing the prediction performance of NN-based model, the training model is divided into four driving conditions (FTP-75 Phase 1, 2, HWFET, WLTC, and on-road) similar to physics-based modeling in the previous section. An on-road example of the result predicted by ConvLSTM is presented in Figure 10 with similar ground truth as in Figure 9d. The reason for choosing an actual on-road experiment as an example is that it has the most fluctuations and disturbances, making it difficult to predict the randomness.

Table 4. Parameter optimization for ConvLSTM models.

	Model Description					MAE	MSE
	Filters	Kernel	Strides	Trainable Parameters	Activation		
#1	256	20	5	22,194,737	SoftPlus	3.2889	20.0674
#2	256	3	5	17,581,617	ReLU	6.1641	61.4168
#3	256	1	5	17,038,807	ReLU	15.5647	355.0674
#4	128	5	5	8,925,489	ReLU	4.3266	34.8466
#5	128	10	1	42,044,209	SoftPlus	4.4020	37.5245
#6	128	20	3	15,507,249	SoftPlus	3.3207	19.8283
#7	128	40	5	16,910,129	ReLU	2.7086	15.2777
#8	64	20	3	7,616,945	SoftPlus	3.4380	25.7581
#9	64	20	3	7,616,945	ReLU	2.6124	13.3402
#10	32	20	3	3,917,553	ReLU	2.7728	15.7787

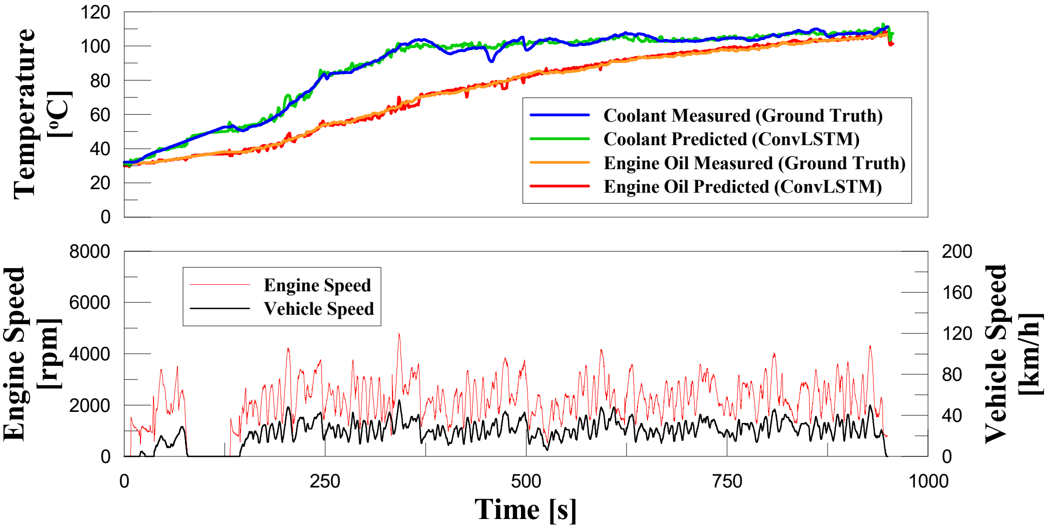


Figure 10. On-road example of Neural Network (Convolutional LSTM) based coolant and engine oil temperature prediction and comparison of actual measurements. Data used in the on-road experiment of physics-based modeling are considered here.

In contrast, for the same input data, TCN shows better performance than ConvLSTM in extracting the features.

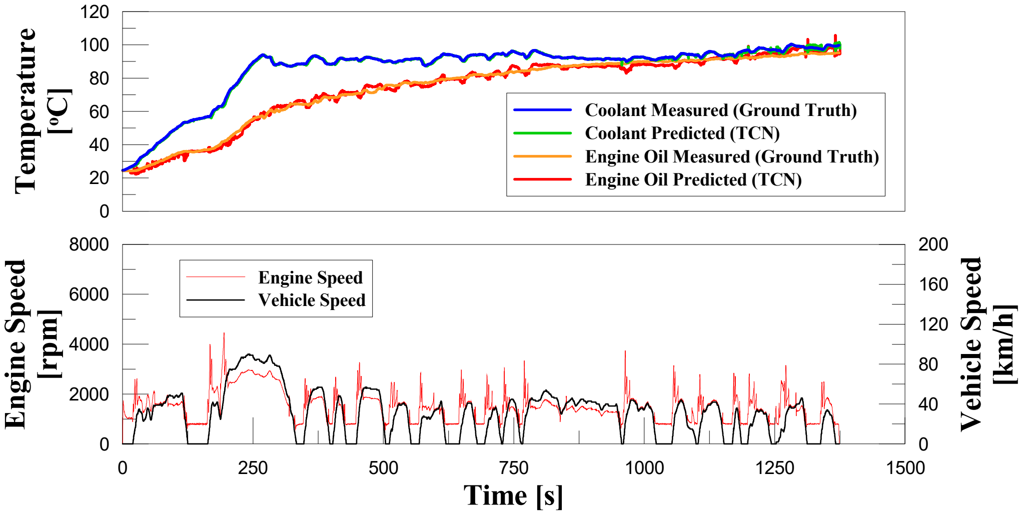
The results of TCN show that, if the number of parameters is large, the accuracy is adversely affected. This phenomenon is presumed to be due to overfitting caused by several learnable parameters. Complex models with many parameters can be considered as insufficiently trained to make accurate predictions, as they require more input data to find the physical correlation.

The TCN model shows the best result among parameter tuning of the ConvLSTM and TCN models based on Tables 4 and 5, and model #3 shows the optimum performance. The final model TCN #3 has 5 hidden layers because it consists of dilations of [1,2,4,8,16,32]. The dilated casual convolution with TCN is better at capturing long-term dependencies than recurrent units. Hence, the TCN model can outperform the prediction accuracy of ConvLSTM and physics-based models. TCN showed stable performance as it was less affected by the selected architecture and parameters. ConvLSTM model is found to be less accurate when the input is too large [44]. This indicates that the sequential processing in the recurrent network is not optimal for handling very long input sequences. Conversely, TCN has excellent results for handling long input sequences.

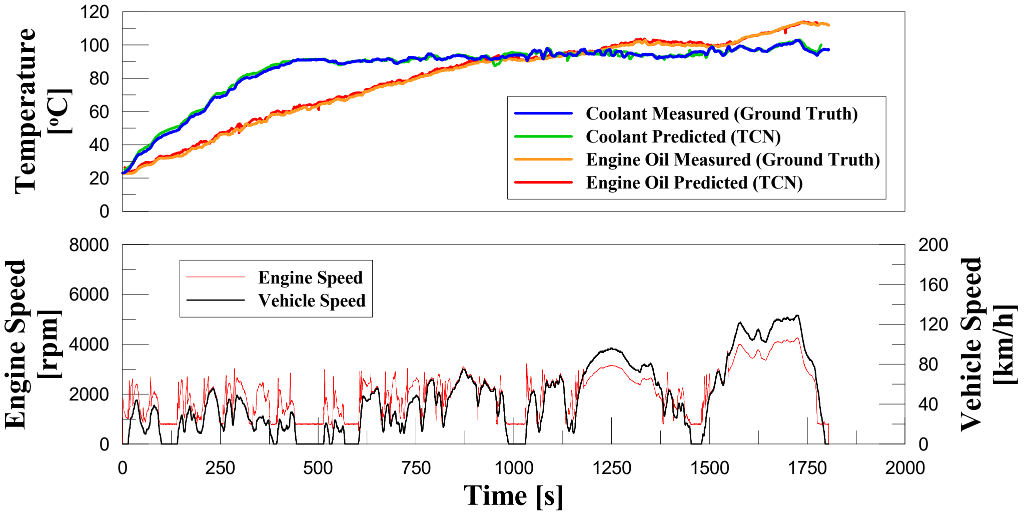
Table 5. Parameter optimization for TCN models.

	Model Description				MAE	MSE
	Filters	Kernel	Dilations	Trainable Parameters		
#1	128	2	[1, 2, 4, 8, 16, 32]	365,826	3.7536	23.8921
#2	64	2	[1, 2, 4, 8, 16, 32]	92,802	0.9148	2.8306
#3	32	2	[1, 2, 4, 8, 16, 32]	23,874	0.2269	0.7441
#4	32	4	[1, 2, 4, 8, 16, 32]	46,978	0.3325	0.9843
#5	32	3	[1, 2, 4, 8, 16, 32]	23,874	0.2885	0.9369
#6	32	2	[1, 2, 4, 8, 16]	19,714	0.3225	1.0207
#7	64	4	[1, 2, 4, 8, 16, 32]	184,066	1.0216	3.0262
#8	64	2	[1, 2, 4, 8, 16, 32, 64]	109,506	1.1830	3.7137
#9	81	3	[1, 3, 9, 27, 81]	181,118	2.0541	14.3222

In summarizing the results, the better performing TCN model was selected as the representative NN model. The temperature change prediction result of TCN model is shown in Figure 11.

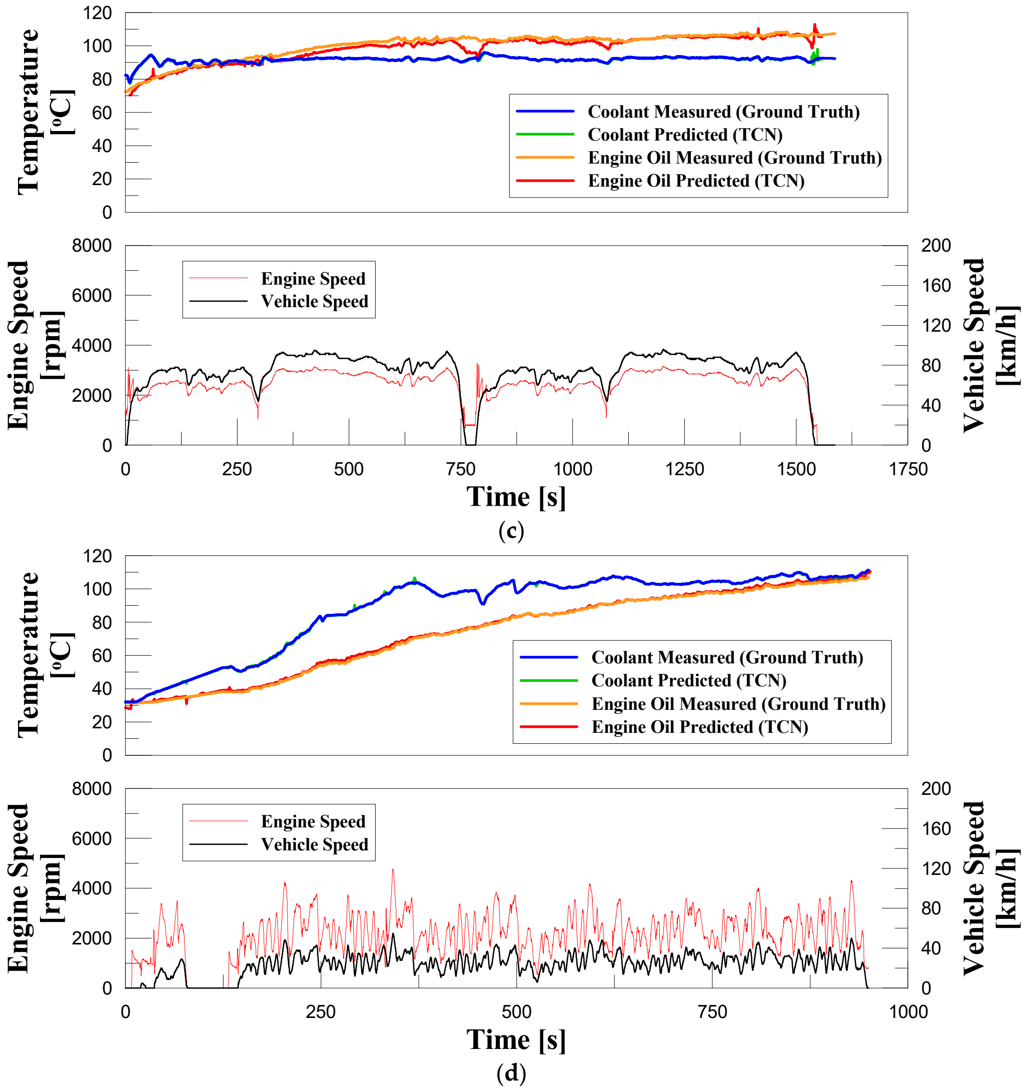


(**a**)



(**b**)

Figure 11. *Cont.*



(**c**)

(**d**)

Figure 11. Neural network method (TCN) prediction of coolant and engine oil temperature and comparison of actual measurements. (**a**) FTP-75 Phase 1,2; (**b**) WLTC; (**c**) HWFET; (**d**) On-road test.

The NN model predicts the coolant temperature accurately. However, error can occur in predicting the engine oil temperature for specific sections, such as the section where the vehicle stops. This error is presumed to be due to insufficient training data for vehicle stopping conditions and is expected to decrease when various situations, such as the moment the vehicle stops, are known.

5.2. Comparative Study

Table 6 compares the prediction results of the physics-based and TCN models in terms of accuracy and working days. Overall, the TCN-based NN method is more accurate than the physics-based modeling. The working days it takes to complete a model can make a big difference in developing time. The NN method can save a lot of time because it does not require experimental parameter studies and does not necessarily require an expert to perform the verification test and data acquisition. Comparing the results of the three models shows that TCN has sufficient potential to replace physics-based modeling.

Table 6. Comparison of the forecasting models.

Model	MAE	MSE	Working Days			
			Experimental Parameter Study	Validation Test	Model Tuning	Total
Physics-based	1.9697	6.6763	8 months	1 months	3 months	12 months
NN-based	0.2269	0.7441	N/A	2 months	1 month	3 months

Figure 12 shows the interquartile range of the predicted error values of the two modeling methods. Because the physics-based model is based on integration, errors are accumulated when incorrect calculations occur at some points. However, NN-based models are less susceptible to accumulation errors because they are based on nonlinear models and use correlations between data. The NN model predicts not only the value of the current step, but also the previous step as a weighted long sequence.

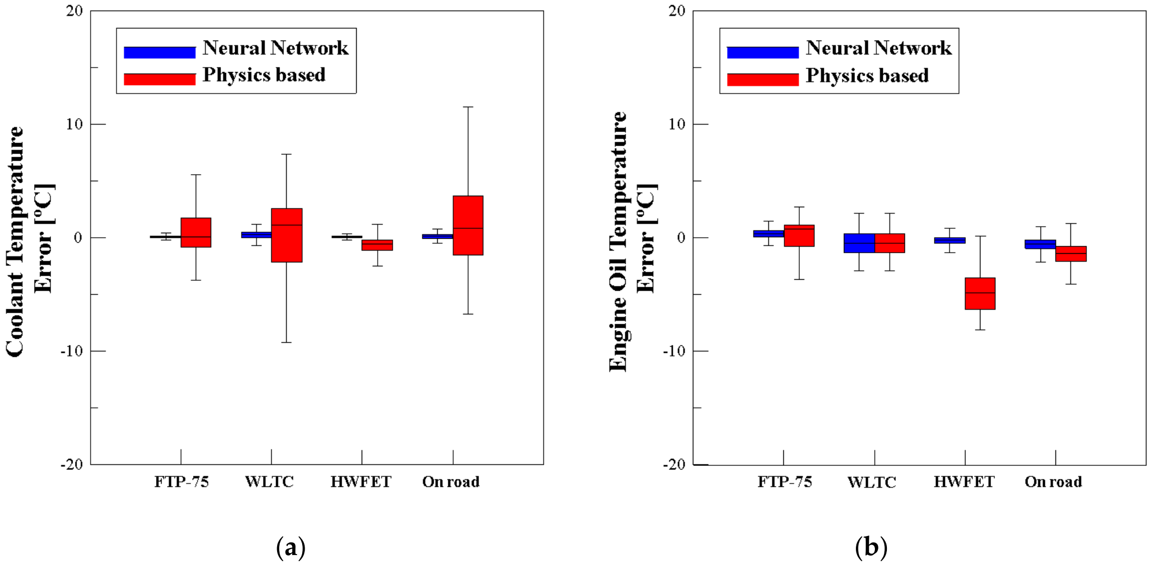


Figure 12. Prediction error expressed by the interquartile range method. The NN model and the physical-based model were compared under various driving conditions. (**a**) Coolant temperature; (**b**) Engine oil temperature.

The NN methods appear to be sufficient to replace the conventional physics-based models in terms of accuracy and speed under the following conditions. Physics-based modeling relies on assumptions to represent a large part of the physical phenomena in simplified models for computational speed. Extracting and simplifying features from these input data is one of the strong points of NN approaches. However, NN modeling shows errors due to lack of training data from regions with difficult driving conditions. The performance can be improved by collecting data from a few specific areas that are difficult to experiment and performing additional experiments using physical relationships as inputs.

6. Discussion

Modeling for predicting the coolant temperature in VTMS is important in design and control, but recently there has been a growing demand to reduce the software development costs. In this paper, we proposed a deep learning framework based on NNs in place of conventional physics-based modeling.

For evaluating the prediction results, the experimental data of the physics- and NN-based models (ConvLSTM, TCN) were compared. Our study shows that, among NN methods, TCN can be an effective tool for predictive modeling in VTMS. The comparison results agree with those of recent studies [48,49]. The optimized structure of TCN was superior to ConvLSTM in terms of performance and cost and is sufficient to replace physics-based modeling under this study conditions.

Rather than formulating all the systems as in physics-based modeling, the NN method learns the causal relationship between input and output. The TCN-based modeling, which has dilated causal convolutions and residual blocks, is less sensitive to parameter selection and can reduce the system cost by replacing the multiple experiments needed for coefficient selection in physics-based modeling. The TCN approach has the potential to further improve performance as better NN architectures are developed.

Based on our research, NN and physical-based models are found to have their own advantages. The NN modeling is error-resistant and useful for finding correlations and features between inputs and outputs. Physics-based modeling is intuitive and computes faster than NN. Intuitive characteristics can be beneficial for engineers to infer the changes in the system and understand the physical impact of each change of element, unlike NNs that only grasp the relationship between inputs and outputs. In addition, it will have some durability against the worst anomalous conditions in which data could not be acquired by the NN method. These two qualities can complement each other. Physics-based modeling can be used to refine some of the inputs, or hybrid models that use both NN and physical-based models in parallel can offer robustness against unpredictable errors in NN model. The future study will consider integrating both the models. In addition, predictive control can be attempted by applying NN-based cooling system modeling to physical coolant flow control.

7. Conclusions

This paper proposed an NN model for a conventional cooling system modeling technique for VTMS. The proposed model was compared with physics-based modeling and verified through experimental results. The model conditions FTP-75, HWFET, WLTC, which express various driving conditions, and the results of real road driving in difficult-to-predict random conditions show that NN technology can be used successfully to predict temperature changes. Recent NN technologies such as ConvLSTM and TCN were compared to physics-based modeling. The TCN's performance was the best of the models tested, and it was easier to build than physics-based models. Unlike physics-based modeling, the NN method only learns the correlation between inputs and outputs, excluding experimental parameter studies, so it can reduce the cost of the experimental equipment required for experimental parameter research and the long time required for the study. The proposed method is expected to reduce research cost and time in the field of predictive control and VTMS design.

Author Contributions: Conceptualization, D.C. and Y.A.; data curation, Y.A.; formal analysis, D.C. and Y.A.; funding acquisition, J.P.; investigation, D.C. and Y.A.; methodology, D.C., N.L. and J.P.; project administration, J.L.; software, D.C., Y.A. and N.L.; supervision, J.P. and J.L.; validation, J.P.; writing—original draft, D.C.; writing—review and editing, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Page, R.W.; Hnatczuk, W. *Thermal Management for the 21st Century—Improved Thermal Control & Fuel Economy in an Army Medium Tactical Vehicle*; SAE Technical Paper 2005-01-2068; SAE International: Washington, DC, USA, 2005.
2. Mahmoud, K.G.; Loibner, E.; Wiesler, B.; Samhaber, C.; Kußmann, C. Simulation-based vehicle thermal management system—Concept and methodology. In Proceedings of the SAE World Congress, Detroit, MI, USA, 3–6 March 2003.
3. Osborne, S.; Kopinsky, J.; Norton, S.; Sutherland, A.; Lancaster, D.; Nielsen, E.; Isenstadt, A.; German, J. *Automotive Thermal Management Technology*; The International Council on Clean Transportation (ICCT): Washington, DC, USA, 2016.
4. Allen, D.; Lasecki, M. *Thermal Management Evolution and Controlled Coolant Flow*; SAE Technical Paper 2001-01-1732; SAE International: Washington, DC, USA, 2001.

5. Curran, A.R.; Johnson, K.R.; Marttila, E.A.; Dudley, S.P. *Automated Radiation Modeling for Vehicle Thermal Management*; SAE Technical Paper 950615; SAE International: Washington, DC, USA, 1995.
6. Kumar, V.; Shendge, S.A.; Baskar, S. *Underhood Thermal Simulation of a Small Passenger Vehicle with Rear Engine Compartment to Evaluate and Enhance Radiator Performance*; SAE Technical Paper 2010-01-0801; SAE International: Washington, DC, USA, 2010.
7. Cipollone, R.; Villante, C. Vehicle thermal management: A model-based approach. In Proceedings of the ASME Internal Combustion Engine Division Fall Technical Conference, Long Beach, CA, USA, 24–27 October 2004.
8. Berry, A.; Blissett, M.; Steiber, J.; Tobin, A.T.; McBroom, S.T. *A New Approach to Improving Fuel Economy and Performance Prediction through Coupled Thermal Systems Simulation*; SAE Technical Paper 2002-01-1208; SAE International: Washington, DC, USA, 2002.
9. Zanini, F.; Atienza, D.; Jones, C.N.; De Micheli, G. Temperature sensor placement in thermal management systems for MPSoCs. In Proceedings of the 2010 IEEE International Symposium on Circuits and Systems, Paris, France, 30 May–2 June 2010; p. 11462528.
10. Zerfowski, D.; Buttle, D. Paradigmenwechsel im automotive-software-markt. *ATZ Automob. Z.* **2019**, *121*, 28–35. [CrossRef]
11. Samad, T.; Stewart, G. *Systems Engineering and Innovation in Control—An Industry Perspective and an Application to Automotive Powertrains*; University of Maryland Model-Based Systems Engineering Colloquia Series: Washington, DC, USA, 2013.
12. Müller, M.T. Interview by Alfred Vollmer, Die neue Architektur der Fahrzeuge. *Automob. Elektron.* **2019**, *3*, 16ff.
13. Proff, H.; Pottebaum, T.; Wolf, P. Software is transforming the automotive world—Four strategic options for pure-play software companies merging into the automotive lane. *Deloitte Insights* **2020**, *1*, 10–11.
14. Patrini, G.; Rozza, A.; Menon, A.K.; Nock, R.; Qu, L. Making deep neural networks robust to label noise: A loss correction approach. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1944–1952.
15. Yang, X.; Ho, D.W.C. Synchronization of delayed memristive neural networks: Robust analysis approach. *IEEE Trans. Cybern.* **2016**, *46*, 3377–3387. [CrossRef]
16. Maas, A.; Le, Q.V.; O'Neil, T.M.; Vinyals, O.; Nguyen, P.; Ng, A.Y. Recurrent neural networks for noise reduction in robust ASR. *Interspeech* **2012**. Available online: http://ai.stanford.edu/~{}amaas/papers/drnn_intrspch2012_final.pdf (accessed on 2 September 2020).
17. Seltzer, M.L.; Yu, D.; Wang, Y. An Investigation of Deep Neural Networks for Noise Robust Speech Recognition. *Proc. ICASSP* **2013**, 7398–7402. [CrossRef]
18. Parveen, S.; Green, P. Speech recognition with missing data using recurrent neural nets. *Adv. Neural Inf. Process. Syst.* **2001**, *14*, 1189–1195.
19. Deng, L.; Yu, D. Deep learning: Methods and applications. *Found. Trends Signal Process.* **2014**, *7*, 197–387. [CrossRef]
20. Gregor, K.; Danihelka, I.; Mnih, A.; Blundell, C. Deep AutoRegressive Networks. In Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 21–26 June 2014.
21. Cao, J.; Li, P.; Wang, W. Synchronization in arrays of delayed neural networks with constant and delayed coupling. *Elsevier Phys. Lett. A* **2006**, *353*, 318–325. [CrossRef]
22. Gooijer, J.G.D.; Hyndman, R.J. 25 years of time series forecasting. *Int. J. Forecast.* **2006**, *22*, 443–473. [CrossRef]
23. Gamboa, J.C.B. Deep learning for time-series analysis. *arXiv* **2017**, arXiv:1701.01887.
24. Dorffner, G. Neural networks for time series processing. *Neural Network World* **1996**, *6*, 447–466.
25. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–445. [CrossRef] [PubMed]
26. LeCun, Y.; Bengio, Y. Convolutional networks for images, speech, and time-series. In *The Handbook of Brain Theory and Neural Networks*; Arbib, M.A., Ed.; MIT Press: Cambridge, UK, 1995.
27. Brownlee, J. Promise of deep learning for time series forecasting. In *Deep Learning for Time Series Forecasting*, 1st ed.; Machine Learning Mastery: Vermont Victoria, Australia, 2018; pp. 3–7.
28. Chollet, F. Deep learning for text and sequence. In *Deep Learning with Python*, 1st ed.; Manning Publications Company: Shelter Island, NY, USA, 2017; pp. 305–309.
29. Aggarwal, C.C. Recurrent neural networks. In *Neural Networks and Deep Learning*, 1st ed.; Springer: Berlin, Germany, 2018; pp. 271–313.

30. Bai, S.; Kolter, J.Z.; Koltun, V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv* **2018**, arXiv:1803.01271.
31. Zhao, W.; Gao, Y.; Ji, T.; Wan, X.; Ye, F.; Bai, G. Deep temporal convolutional networks for short-term traffic flow forecasting. *IEEE Access* **2019**, *7*, 114496–114507. [CrossRef]
32. Lea, C.; Flynn, M.D.; Vidal, R.; Reiter, A. Temporal Convolutional Networks for Action Segmentation and Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
33. Oord, A.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; Kavukcuoglu, K. WaveNet: A generative model for raw audio. *arXiv* **2016**, arXiv:1609.03499.
34. Briggs, I.; Murtagh, M.; Kee, R.; McCulloug, G.; Douglas, R. Sustainable non-automotive vehicles: The simulation challenges. *Energy Rev.* **2017**, *68*, 840–851. [CrossRef]
35. Hussein, A.A. Capacity fade estimation in electric vehicle li-ion batteries using artificial neural networks. *IEEE Trans. Ind. App.* **2015**, *51*, 2321–2330. [CrossRef]
36. Zahid, T.; Xu, K.; Li, W.; Li, C.; Li, H. State of charge estimation for electric vehicle power battery using advanced machine learning algorithm under diversified drive cycles. *Energy* **2018**, *162*, 871–882. [CrossRef]
37. Park, J.; Kim, Y. Supervised-learning-based optimal thermal management in an electric vehicle. *IEEE Access* **2020**, *8*, 1290–1302. [CrossRef]
38. Heywood, J.B. Engine heat transfer. In *Internal Combustion Engine Fundamentals*, 1st ed.; McGraw-Hill Book Company: New York, NY, USA, 1988; pp. 673–674.
39. Ryu, T.Y.; Shin, S.Y.; Lee, E.H.; Choi, J.K. A study on the heat rejection to coolant in a gasoline engine. *Trans. Korean Soc. Auto. Eng.* **1997**, *5*, 77–88.
40. Kuo, P.-H.; Huang, C.-J. A high precision artificial neural networks model for short-term energy load forecasting. *Energies* **2018**, *11*, 213. [CrossRef]
41. Pascanu, R.; Mikolov, T.; Bengio, Y. On the difficulty of training recurrent neural networks. In Proceedings of the 30st International Conference on Machine Learning (ICML), Atlanta, GA, USA, 16–21 June 2013; Volume 28, pp. III-1310–III-1318.
42. Jozefowicz, R.; Zaremba, W.; Sutskever, I. An empirical exploration of recurrent network architectures. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 6–11 July 2015; Volume 37, pp. 2342–2350.
43. Culurciello, E. The Fall of RNN/LSTM. Available online: https://towardsdatascience.com/the-fall-of-rnn-lstm-2d1594c74ce0 (accessed on 2 September 2020).
44. Sherstinsky, A. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. *Elsevier Phys. D Nonlinear Phenom.* **2020**, *404*, 132306. [CrossRef]
45. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.-Y. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 3–4.
46. Hinton, G.; Deng, L.; Yu, D.; Dahl, G.; Mohamed, A.; Jaitly, N.; Senior, A.; Vanhoucke, V.; Nguyen, P.; Sainath, T.N.; et al. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Process. Mag.* **2012**, *29*, 82–97. [CrossRef]
47. Zhang, Y.; Chan, W.; Jaitly, N. Very deep convolutional networks for end-to-end speech recognition. In Proceedings of the 42nd International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 4845–4849.
48. Lara-Benítez, P.; Carranza-García, M.; Luna-Romera, J.M.; Riquelme, J.C. Temporal convolutional networks applied to energy-related time series forecasting. *Appl. Sci.* **2020**, *10*, 2322. [CrossRef]
49. Lea, C.; Vidal, R.; Reiter, A.; Hager, G.D. Temporal convolutional networks: A unified approach to action segmentation. In Proceedings of the Computer 14th European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–10 and 15–16 October 2016; pp. 47–54.
50. Lara-Benítez, P.; Carranza-García, M.; García-Gutiérrez, J.; Riquelme, J. Asynchronous dual-pipeline deep learning framework for online data stream classification. *Integr. Comput. Aided Eng.* **2020**, *27*, 1–19. [CrossRef]
51. Keskar, N.S.; Mudigere, D.; Nocedal, J.; Smelyanskiy, M.; Tang, P.T.P. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv* **2016**, arXiv:1609.04836.
52. Elsayed, N.; Maida, A.S.; Bayoumi, M. Empirical activation function effects on unsupervised convolutional lstm learning. In Proceedings of the 30th International Conference on Tools with Artificial Intelligence (ICTAI), Volos, Greece, 5–7 November 2018; Volume 10, p. 1109.

53. US Environmental Protection Agency. Fuel economy labeling of motor vehicles: Revisions to improve calculation of Fuel Economy Estimates; 40 CFR Part 86 and 600. *Fed. Regist.* **2006**, *71*, 77872–77969.
54. United Nations. Global technical regulation No.15 worldwide harmonized Light vehicles Test procedure. *UN ECE Trans.* **2014**, *180*, Add.15.