Energy Consumption Prediction of Electrical Vehicles Through Transformation of Time Series Data

Xudong Hu
Electrical and Computer Engineering
National University of Singapore
Singapore
e0459193@u.nus.edu

Biplab Sikdar
Electrical and Computer Engineering
National University of Singapore
Singapore
elebisik@nus.edu.sg

***Abstract*—In recent years, global greenhouse gas emissions have become an important issue. Electrical Vehicles (EVs) are a promising solution to reduce greenhouse gas emissions. However, widespread adoption of EVs is hindered in part by "range anxiety" wherein the remaining drivable distance is unknown. Models that can predict the energy consumption of EVs can not only address this issue, but also assist in optimizing charging patterns. This paper proposes a model for EV energy consumption prediction that adopts relevant factors (temperature, road gradient, vehicle loading, etc.) to characterize a driver's real-life vehicle usage. By using a novel transformation of the input data, the proposed model can output the consumption prediction of multiple users simultaneously. Additionally, the model can process multiple types of car models that effectively use information specific to that group. The results show that the system can parallelize the prediction task with desired accuracy when compared to the single-output system and offers the ability to model different EV models at the same time.**

***Index Terms*—Electrical Vehicle, Convolutional Neural Network, Energy Consumption prediction, Parallel processing**

I. Introduction

Fossil fuel combustion is the major source of CO_2. As being highly reliant on fossil fuels for energy generating, the transportation sector is responsible for large quantities of CO_2 emissions, accounting for around one-third of energy-related CO_2 emissions from end-use sectors. Therefore, improving the energy efficiency of and reducing emissions from transportation systems play a key role in controlling general emissions and limiting global warming. According to the latest statistics published by International Energy Agency (IEA), road transportation accounts for three-quarters of CO_2 emissions from the current transport systems. With the highest percentage of total transport-related emissions, road transport emissions have received considerable critical attention, which has heightened the need for advanced vehicle and fuel technologies. Among such technologies, electrification has been widely considered a promising potential long-term solution to reduce emissions from road transport. Studies such as that conducted in [1] have shown that electric vehicles (EVs) using electricity generated by low-carbon sources contribute to a significant reduction in CO_2 emission from road transport.

Copyright: $979-8-3503-1997-2/23/\$31.00$ ©2023 IEEE

The potential environmental benefits are a strong motivator for the adoption of EVs. However, this adoption has not gone fast as desired. One of the barriers to widespread adoption is drivers' range anxiety (i.e., the driver's fear that the remaining energy of EVs is inadequate to reach their designated destination). The major causes for the anxiety are uncertainty in the remaining driving range and the sparse public charging infrastructure network. Accurate estimation of the remaining driving range has been heralded as a promising measure to alleviate drivers' anxiety. Estimating the remaining driving range requires both the evaluation of the remaining energy and the prediction of the possible energy consumption. Usually, it is easy to evaluate the remaining energy based on the state of charge presented on any vehicle dashboard. Energy consumption prediction, therefore, has become a central issue for the estimation of the remaining driving range. More recently, in addition to traditional machine learning (ML) models like Gaussian mixture models (GMM), Support Vector Machines (SVM), etc., and neural networks have applied to explore better approaches for the prediction of energy consumption.

There are different existing approaches that aim to obtain the energy consumption prediction of EVs. Analytical models utilize a set of physical equations to model the engine activities. However, it is a challenge to acquire all relevant parameters required by equations. Statistic models generate mathematical representations from sampling data. The models are heavily data-dependent and usually concentrate on linear correlations. Machine learning methods with neural network implementation improve the prediction by taking non-linear correlations into consideration. With different architectures and inputs, neural networks method can obtain the energy consumption prediction at different resolutions and improve the performance by processing the input in time-series format (per-second data). However, existing techniques have less considerations with regard to bigger scale of data to generate a possible consumption profile of the remaining energy at a higher frequency. Additionally, there is a lack of studies to distinguish the difference inside a training data group if multiple sub-group properties (multiple car models) are presented.

To address these concerns, this paper proposes a neural network architecture that aims to obtain the energy consumption profile of EVs. In contrast to existing work, the proposed model can also process multiple groups of data (corresponding to different car models) at the same time. Each group of the data generates a consumption prediction with consideration of its own sub-group property. By using the same size of input data, the model also proves the efficiency of information utility by achieving equal or better accuracy compared to the single-output architecture control model.

The overall structure of the study takes the form of five sections, including this "Introduction" section. Section II presents an overview of the current status of related works. Section III describes the proposed modeling approach, including data collection, preprocessing, and ML model elaboration, while model results are discussed in Section IV. Finally, the concluding remarks are presented in Section V.

II. Related Work

In recent years, numerous attempts have been made to predict the energy consumption of EVs. According to previous efforts, the major prediction approaches used to estimate energy consumption include analytical models, statistical models, and machine learning models. Analytical models describe the essential processes relevant to vehicle dynamics with a set of physical equations. According to the equations, the net force contributes to the motion of a vehicle, and/or the required power to generate the force is calculated. Subsequently, the energy consumption can be calculated based on its relationship with force [2], [3] ($F \times d$, where F is the net force that contributes to the motion of a vehicle and d is the distance traveled), or power [4], [5] ($P \times t$, where P is the required power and t is the operation time).

Statistical models use a set of statistical methods to generate a mathematical representation of the relationships between the energy consumption of a vehicle and plausible impact factors. Statistical models are data-dependent. Sampling data is required to deduce empirical relationships between the predictor variable and the response variable. Predefining the relationship between variables is the first step of statistical models [6]. For example, in regression analysis (a widely used approach to build a statistical model) the relationship is expressed as an equation with unknown coefficients which can be estimated based on sampling data. Previous studies have applied linear regression models to predict EV energy consumption [7], [8]. Liu et al. also conduct studies on the impacts of temperature [9] and road gradient [10] on energy consumption separately.

Machine learning models consist of traditional machine learning models and neural network based methods. The neural network based methods demonstrate better potential for the development of models for computation of power. Studies [11], [12] try to model driver's behavior with (ANN) architecture with vehicle speed, acceleration, jerk, and road information as input. The results show that neural networks are a promising method for predicting energy consumption. To focus on the EV model performance, another method is proposed in [13], where a convolutional neural network is built to predict the energy consumption for real-time output. However, the input parameter only contains 3 features that are considered. As mentioned in [14], other factors also have an impact on energy consumption and need to be considered. Furthermore, the model is trained based on one specific type of EV model. The result is not transferable and each additional EV model requires a separate model to obtain the desired output.

III. Methodology

A. Dataset

To obtain data that can represent different driving behaviors, emobpy [15] is adopted as a simulation model to generate training and testing data. emobpy is an open-source tool for battery electric vehicles to generate time series data using Python. It contains various settings to select driver behavior as well as vehicle models. To simulate the driving behavior profile, there are configurations to define the probability map of "departure and destination", "distance and duration", as well as "trips per day". The probabilities set can closely simulate the driving behavior in Germany by achieving cumulative distribution of trips and mileage as compared to underlying German mobility statistics [16]. On the other hand, there are three categories of drivers defined in the trip rule to select trips based on weekends and weekdays. The trip rules and category probability are listed in Table I. Full-time commuters and part-time commuters have more stable trips per day compared to non-commuters since they need to travel to workplaces. As for the vehicle, there are multiple built-in EV models with parameters of battery capacity, motor type, torque, etc. Instead of random selection, four EV models are selected evenly in the population to focus on the subgroup characteristic study. During the data generation phase, a driver category is selected based on the probabilities listed in Table I, and then one of the four EV models is assigned to that driver. By setting the time to one year and time resolution to 15 minutes, one driver will have a 35040 data-points with timestamps. Overall, the simulation generates records for 200 users that contain seven million data-points with timestamps using Algorithm 1. The records are split into 90% as the training dataset and 10% as the testing dataset.

B. Features and their transformation and normalization

Instead of just focusing the engine efficiency like [13], we consider additional factors that have an impact on battery consumption in real life. Based on the analysis in [14], wind speed, temperature, inclination, and loading, etc. all have an impact on battery consumption. Considering the factors that are in the simulation tool, we select the available features that are listed in Table II. Wang et al. [17] proposed an encoding method to map the time series data as images

TABLE I: Driver category and rule setting

Condition	Full-time commuters		Part-time commuters		Non-commuters	
	Weekday	Weekend	Weekday	Weekend	Weekday	Weekend
Category probability	0.4		0.3		0.3	
Minimum Number of trips	1	1	1	1	1	1
Last trip destination	Home	Home	Home	Home	Home	Home
Minimum time at home	9	6	9	6	9	6
Trip to work	At least 1	Based on need	At least 1	Based on need	N.A.	N.A.
Minimum time at workplace	7	3	3.5	3	N.A.	N.A.
Maximum Time at workplace	8	4	4	4	N.A.	N.A.
Minimum state duration at workplace	3.5	3	3.5	3	N.A.	N.A.
Minimum state duration except for workplace	0.25	0.25	0.25	0.25	0.25	0.25

Algorithm 1: Dataset Generation Algorithm

$Initialize\ Categories,\ vehicle\ brand,\ and\ mean\ passenger\ numbers\ and\ then\ assign\ probability$

$//\ \ CFT:\ \ full-time\ \ commuter$
$//\ \ CPT:\ \ part-time\ \ commuter$
$//\ \ NFT:\ \ non-commuter$
$Cat_values \leftarrow [CFT: 0.4,\ CPT: 0.3,\ NFT: 0.3]$

$//\ \ Distribution: 0.4, 0.3, 0.2, 0.1$
$mean\_passenger\_number \leftarrow [1.5, 2, 2.5, 3]$

$//\ \ vehicle\_selection\ \ and\ \ distribution$
$Volkswagen\ \ \ \leftarrow ID.3:\ 0.25$
$BMW \leftarrow i3s\ Edition\ RoadStyle\ 42\ kWh:\ 0.25$
$Audi\ \leftarrow e-tron\ Sportback\ 55\ quattro:\ 0.25$
$Tesla\ \leftarrow Model\ X\ Long\ Range\ (SR):\ 0.25$

for $counter\ = 0,\ counter < 200,\ counter++$ **do**
  Pick driver category vehicle_model
  set rules of category based on Table I
  set total hours = 8760
  set time step = 0.25
  set Trips per day
  set Distance and duration
  Generate the travel summary profile

  set vehicle_model
  set mean_passenger_number
  set the rest of the parameter
  Generate the Consumption time_series file
  Combine data to form dataset
end

known as Gramian Angular Fields (GAF). The time series data $F = \{f_1, f_2, \cdots, f_n\}$ is normalized to an interval between $[-1, 1]$ by:

$$\widetilde{f}_i = \frac{(f_i - \max(F)) + (f_i - \min(F))}{\max(F) - \min(F)} \tag{1}$$

TABLE II: Selected features and description

Feature name	Description
Number of passengers	Number of passengers to obtain vehicle loading
Vehicle speed	Speed at different timestamps
Driving Cycle	Worldwide Harmonized Light Vehicles Test Cycle (WLTC) or Environmental Protection Agency (EPA)
Road gradient	Describes the slope in radians
Road type	Built in road type to obtain rolling resistance coefficient
Temperature	Ambient temperature in Kelvin
Wind speed	Local wind speed
Weekend	Weekend or weekday indicator
Category	Indicate if driver is full-time/part-time/non-commuter

After feature scaling, we can convert the features to polar coordinates with value and time stamps using:

$$\begin{cases} \phi = \arccos(\widetilde{f}_i), \quad -1 \leq \widetilde{f}_i \leq 1, \widetilde{f}_i \in \widetilde{F}_i \\ \\ r = \frac{t_i}{N}, \quad t_i \in N \end{cases} \tag{2}$$

where t_i represents the time stamp and N represents the distance span of the polar coordinate system. The GAF matrix is defined as the inner product of the time series feature:

$$\widetilde{F}' \cdot \widetilde{F} - \sqrt{I - \widetilde{F}^2}' \cdot \sqrt{I - \widetilde{F}^2} \tag{3}$$

where I is the unit vector. Thus, we can get a $N \times N$ square matrix where N is the number of timestamps that represents the temporal relations between timestamps.

C. Transformation

Inspired by the fact that linear transformation retains the data properties with shifting and scaling, we can take its advantage to achieve parallel processing as well as study subgroup properties by searching for an appropriate linear transformation. 2D-Discreet Cosine Transformation (2D-DCT) [18] is selected as the transformation method to retain the

Algorithm 2: Data Preprocessing

input : *Individual driving record*
output: *Transformed input*

Load dataset from the path
Drop unwanted columns
Create dictionary ranged $(0, N-1)$ *based on vehicle_selection*

$Rx\_i \leftarrow 9$ *features from Table* II
$Ry\_i \leftarrow$ *Time series consumption*
// $i \leftarrow dict\{vehicle\_model\}$

// *General process*
$Rx\_i \leftarrow Rx\_i.reshape(-1, 48, 9)$
$Rx\_i \leftarrow$ *add* 2 *pad at bottom*
$Rx\_i \leftarrow transpose(0, 2, 1)$
$Rx \leftarrow$ *concatenate* Rx_i *on axis* 2

$Ry\_i \leftarrow Ry\_i.reshape(-1, 48, 1)$
$Ry\_i \leftarrow$ *add* 2 *pad at bottom*
$Ry\_i \leftarrow transpose(0, 2, 1)$
$Ry \leftarrow$ *concatenate* Ry_i *on axis* 2

// *transform and create dataset*
for *rows in Rx* **do**
  Create input matrix S // 100×100
  for *model in models* **do**
    Get N based on model dictionary
    $i \leftarrow N * 50,\ j \leftarrow 50 * (N+1)$
    $r \leftarrow int(N/2) * 50,\ l \leftarrow 50 * (N\%2 + 1)$
    $S\_temp \leftarrow Rx[:, i:j]$
    $S\_temp \leftarrow$ *normalize from Eq.*1
    $S\_temp \leftarrow$ *GAF representation from Eq.*3
    $S\_temp \leftarrow 2D - DCT$ *from Eq.*4
    $S[:, r : r+50, l : l+50] \leftarrow S\_temp$
  end
  $S \leftarrow Invers\ 2D - DCT$
end

feature independence when parallel processing multiple inputs simultaneously. 2D-DCT is defined as:

$$X_{k1,k2} = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x_{n1,n2} \cos\left[\frac{\pi}{N_1}\left(n_1 + \frac{1}{2}\right)k_1\right] \cos\left[\frac{\pi}{N_2}\left(n_2 + \frac{1}{2}\right)k_2\right] \quad (4)$$

$$k = 0, \cdots, N-1$$

The transformation also has the property that convolution before transformation equals multiplication after transformation:

$$\{g * h\}(X) = T^{-1}\{G \cdot H\} \quad (5)$$

where $*$ denotes the convolution operation and $\cdot$ denotes point-wise multiplication. During the data preprocessing stage, the input is generated through Algorithm 2. The input data (100×100) is split into four 50×50 regions. The mixing output from each region is a convolutional result from the transformed message and position shifting matrix (the matrix is 50×50 in size with a value of 1 at the center of the matrix). Figure 1 demonstrates the mixing process of a data input that contains normalization, GAF matrix conversion, as well as the 2D-DCT transformation before entering the convolutional neural network.

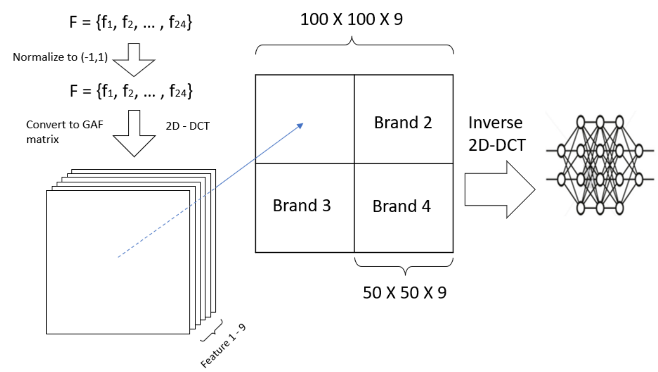


Fig. 1: Illustration of data preprocessing for one subgroup.

D. Network

Convolutional neural networks are good at learning the characteristics of images [19], [20] such as Alex, Image-net, etc. The image encoding input is designed to go through a convolutional neural network. The network typically consists of convolutional layers, Rectified Linear Units (ReLUs), max-pooling layers, as well as fully connected layers. There are changes at the fully connected layer with the change of the output format as shown in Figure 2. The last fully connected layer X_{out} is defined as:

- *Single-output model:* $X_{out} = 1$ for period consumption prediction; $X_{out} = 100$ for timestamp consumption prediction
- *Multiple-output model:* $X_{out} = 4$ for period consumption prediction; $X_{out} = 200$ for timestamp consumption prediction

To prevent the dead neuron challenge, Leaky ReLU is adopted to replace normal ReLU function.

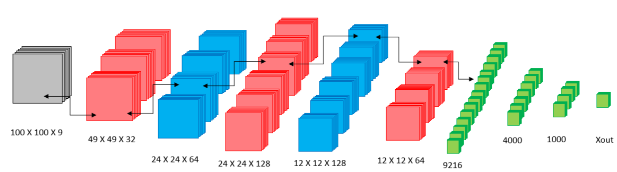


Fig. 2: Convolutional neural network architecture.

TABLE III: Comparison of time-series output to total sum output

Test Name	Xout = 1		Xout = 100	
	MSE	MAE	MSE	MAE
Mean test	327	6.62	1.39	0.45
Std test	3001	16.8	11.55	1.09
Median test	10.4	3.23	0.03	0.17

E. Objective functions and optimizer

Mean average error (MAE: defined in Eqn. (6)) and mean squared error (MSE: defined in Eqn. (7)) are commonly adopted for regression task evaluation:

$$MAE\ Loss = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right| \tag{6}$$

$$MSE\ Loss = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \tag{7}$$

where $\hat{Y}_i$ is the predicted output from the neural network for the i-th time interval and Y_i is the actual consumption. We adopt both functions in the different training sessions to find the optimum decision.

Adam optimizer [21] is selected during the training back-propagation process to update the weight of the neurons. We also adopt the adaptive learning rate method with starting learning set to 0.001.

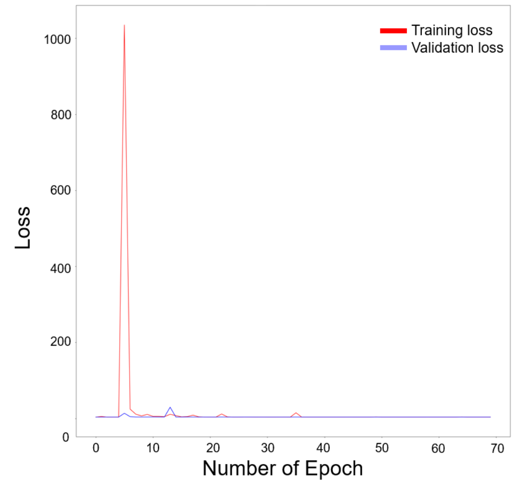
IV. Results

A. Time series output

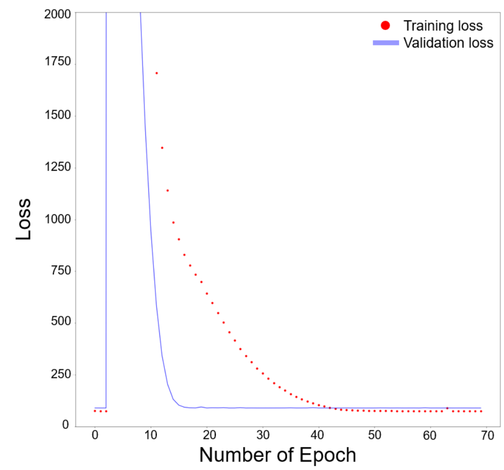
To obtain the energy consumption during each prediction interval, we implemented two networks with different approaches. The first one is to directly output the total energy consumption prediction as a single value. Thus, we set X_{out} to 1 at the last fully connected layer. The other one is to output the energy consummation per time interval and sum the result to obtain the total consumption prediction. This setup requires the last fully connected layer X_{out} to be set to 100. Comparing the output from different set ups of the network, we can see the time-series output setup is better in terms of the mean value as well as the standard deviation as compared to the total consumption prediction output, as shown in Table III. The two models were trained under MAE as the objective function. The performance difference is observed because of the additional penalty from the MSE function.

B. MSE versus MAE

During the training session, the MAE objective function has better performance compared to the MSE function in two aspects: convergence and speed of convergence. Since the target consumption contains lots of 0 values within the time series representation, the MSE function may fail to converge during the training (observed during the training for the time-series representational single-output model). As for the convergence speed, it is also observed that adopting MSE as the objective function (Figure 3b) makes the training process slower to converge compared to the MAE objective function (Figure 3a) as shown in Figure 3.



(a) Loss during training for MAE objective function.



(b) Loss during training for MSE objective function.

Fig. 3: Convergence speed comparison between MAE and MSE as objective function.

As mentioned in Section IV-A, training fails to converge quickly when MSE is adopted as the objective function. To achieve quicker convergence, we implement a cumulative sum operation for the timestamp representation output:

$$\widetilde{Y}_{i+1} = Y_{i+1} + Y_i \tag{8}$$

This implementation reduces the number of 0 values in the target time series array as shown in Figure 4 and successfully makes the training process converge. Table IV shows the performance change before and after cumulative sum implementation.

C. Parallel processing performance

With the 2D-DCT implementation, the model can process four groups (corresponding to different car models) of data in

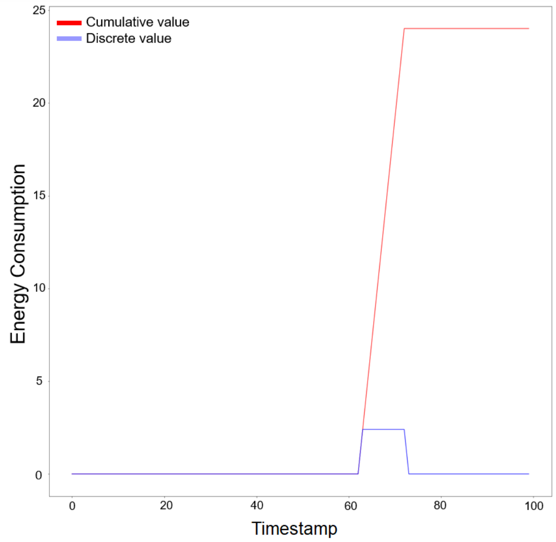


Fig. 4: Illustration of the 0 value reduction from cumulative sum.

TABLE IV: Model performance comparison before and after convergence

Test Name	No cumulative sum		With cumulative sum	
	MSE	MAE	MSE	MAE
Mean test	1.5e10	31570	121	4.07
Std test	2e8	2166	1143	6.94
Median test	1.5e10	31931	21.36	3.55

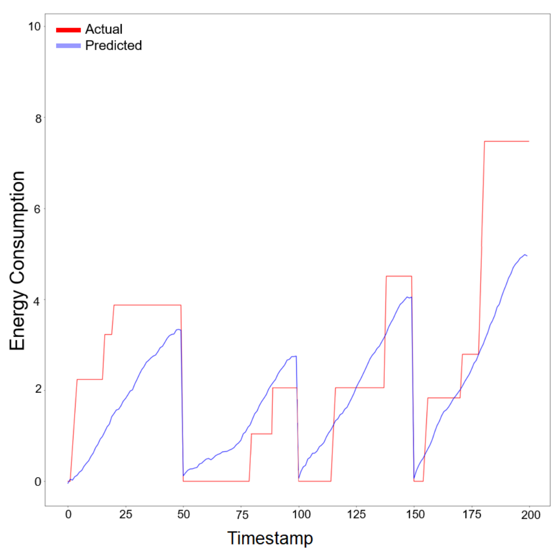


Fig. 5: Illustration of predicted value failing to catch up with the speed of increment

parallel. Each of the subgroups contains 50×50 data to form the 100×100 input as network input. Table V lists the models with setting variations on output mode, objective function, as well as cumulative sum implementation. It contains two evaluation metrics:

- *Per timestamp accuracy:* The accuracy is calculated based on each timestamp. The result represents the average performance of the output neurons.
- *Consumption over period accuracy:* The accuracy is calculated based on the overall consumption over the timestamp. The result represents the capability of predicting driver behavior at a larger scale of time.

We observe that the multiple-output models offer equal or better results regardless of the chosen objective function for both metrics. This can be explained based on the fact that the target signal is sparse, and multiple drivers' data contribute more non-zero values at the output which makes it easier for the model to learn the pattern during training. Besides, the multiple-output model also provides visibility on per EV model accuracy (TableVI); the model also demonstrates that fewer timestamps or smaller input matrix can also achieve the same accuracy. On the other hand, although cumulative sum implementation can help in model convergence during training, it still under-performs when compared to the direct output model when the model can converge itself. It can also lead to results where the cumulative output can increase very fast if the output target is large in magnitude. The network tends to have a smooth output and thus fails to catch-up with the increment, as shown in Figure 5.

V. Conclusion

This paper addressed the problem of energy consumption forecasting for EVs by adopting a machine learning framework that leverages data transformation to facilitate the simultaneous prediction of users from different groups. During the data pre-processing stage, besides normalization and image encoding, this paper proposes a 2D-DCT-based transformation system that can mix four groups of drivers' records as one input. The transformation allows the network to study the behavior of different EV models and provide energy consumption prediction simultaneously. There is a challenge encountered during the training stage for some networks due to the sparseness of the output data. A cumulative summation operation is implemented to reduce the number of 0 values, thereby helping the network to converge during training. The implementation proves to be effective in network convergence but does not perform as well as the normal architecture. It is a method that can solve a specific problem and it worth further exploration to enable it to work in more scenarios. The test results show that the multiple-output architecture can provide better results while focusing on subgroup behavior at the same time.

VI. Acknowledgment

This research was supported in part by the Ministry of Education, Singapore under grants R-263-000-E78-114 and R-263-001-E78-114).

References

[1] L. A.-W. Ellingsen, B. Singh, and A. H. Strømman, "The size and range effect: lifecycle greenhouse gas emissions of electric vehicles," *Environmental Research Letters*, vol. 11, no. 5, p. 054010, 2016.

TABLE V: Model performance summary

Output Mode		Single-output Model [22]				Multiple-output Model			
Objective Function		MSE		MAE		MSE		MAE	
With Cumulative sum		Yes	No	Yes	No	Yes	No	Yes	No
Per Timestamp	RMSE Mean	11	38449	11.2	0.04	8.41	6.02	8.63	0.04
	RMSE Std	33	14329	34.8	0.1	22.3	1.74	22.6	0.08
	MAE Mean	4.07	31570	2.86	0.005	2.53	5	1.8	0.005
	MAE Std	6.94	2166	7.59	0.01	3.3	0.21	3.61	0.008
Sum over the period	RMSE Mean	796	583284	810	1.17	350	105	360	0.84
	RMSE Std	2596	202148	2672	3.4	993	30	319	2
	MAE Mean	384.8	582213	280	0.5	120	95.8	90	0.24
	MAE Std	697.3	35335	760	1.1	166	4	181	0.41

TABLE VI: Performance decomposition for different EV models

	Overall	Model 1	Model 2	Model 3	Model 4
RMSE	0.84	0.84	0.85	0.76	0.91

[2] O. Travesset-Baro, M. Rosas-Casals, and E. Jover, "Transport energy consumption in mountainous roads. a comparative case study for internal combustion engines and electric vehicles in andorra," *Transportation Research Part D: Transport and Environment*, vol. 34, pp. 16–26, 2015.

[3] A. I. Croce, G. Musolino, C. Rindone, and A. Vitetta, "Traffic and energy consumption modelling of electric vehicles: Parameter updating from floating and probe vehicle data," *Energies*, vol. 15, no. 1, p. 82, 2022.

[4] X. Wu, D. Freese, A. Cabrera, and W. A. Kitch, "Electric vehicles' energy consumption measurement and estimation," *Transportation Research Part D: Transport and Environment*, vol. 34, pp. 52–67, 2015.

[5] R. Zhang and E. Yao, "Electric vehicles' energy consumption estimation with real driving condition data," *Transportation Research Part D: Transport and Environment*, vol. 41, pp. 177–187, 2015.

[6] M. Koengkan, J. A. Fuinhas, M. Belucio, N. K. Alavijeh, N. Salehnia, D. Machado, V. Silva, and F. Dehdar, "The impact of battery-electric vehicles on energy consumption: A macroeconomic evidence from 29 european countries," *World Electric Vehicle Journal*, vol. 13, no. 2, 2022.

[7] X. Qi, G. Wu, K. Boriboonsomsin, and M. J. Barth, "Data-driven decomposition analysis and estimation of link-level electric vehicle energy consumption under real-world traffic conditions," *Transportation Research Part D: Transport and Environment*, vol. 64, pp. 36–52, 2018, the contribution of electric vehicles to environmental challenges in transport. WCTRS conference in summer.

[8] C. De Cauwer, W. Verbeke, T. Coosemans, S. Faid, and J. Van Mierlo, "A data-driven method for energy consumption prediction and energy-efficient routing of electric vehicles in real-world conditions," *Energies*, vol. 10, no. 5, 2017.

[9] K. Liu, J. Wang, T. Yamamoto, and T. Morikawa, "Exploring the interactive effects of ambient temperature and vehicle auxiliary loads on electric vehicle energy consumption," *Applied Energy*, vol. 227, pp. 324–331, 2018, transformative Innovations for a Sustainable Future – Part III.

[10] K. Liu, T. Yamamoto, and T. Morikawa, "Impact of road gradient on energy consumption of electric vehicles," *Transportation Research Part D: Transport and Environment*, vol. 54, pp. 74–81, 2017.

[11] A. D. Alvarez, F. S. Garcia, J. E. Naranjo, J. J. Anaya, and F. Jimenez, "Modeling the driving behavior of electric vehicles using smartphones and neural networks," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 3, pp. 44–53, 2014.

[12] J. Felipe, J. C. Amarillo, J. E. Naranjo, F. Serradilla, and A. Díaz, "Energy consumption estimation in electric vehicles considering driving style," in *2015 IEEE 18th international conference on intelligent transportation systems*. IEEE, 2015, pp. 101–106.

[13] S. Modi, J. Bhattacharya, and P. Basak, "Estimation of energy consumption of electric vehicles using deep convolutional neural network to reduce driver's range anxiety," *ISA transactions*, vol. 98, pp. 454–470, 2020.

[14] A. Skuza and R. Jurecki, "Analysis of factors affecting the energy consumption of an ev vehicle-a literature study," in *IOP Conference Series: Materials Science and Engineering*, vol. 1247, no. 1. IOP Publishing, 2022, p. 012001.

[15] C. Gaete-Morales, H. Kramer, W.-P. Schill, and A. Zerrahn, "An open tool for creating battery-electric vehicle time series from empirical data, emobpy," *Scientific Data*, vol. 8, no. 1, p. 152, 2021.

[16] "Kuhnimhof, t. & nobis, c. mobilität in deutschland – mid: Ergebnisbericht." https://elib.dlr.de/125879/, accessed: 2023-3-5.

[17] Z. Wang, T. Oates *et al.*, "Encoding time series as images for visual inspection and classification using tiled convolutional neural networks," in *Workshops at the twenty-ninth AAAI conference on artificial intelligence*, vol. 1. AAAI Menlo Park, CA, USA, 2015.

[18] X. Hu and B. Sikdar, "Sub-group based machine learning for gas consumption prediction," in *Proc. IEEE CSDE*, 2021, pp. 1–6.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.

[20] E. A. Smirnov, D. M. Timoshenko, and S. N. Andrianov, "Comparison of regularization methods for imagenet classification with deep convolutional neural networks," *AASRI Procedia*, vol. 6, pp. 89–94, 2014, 2nd AASRI Conference on Computational Intelligence and Bioinformatics.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[22] S. Modi, J. Bhattacharya, and P. Basak, "Estimation of energy consumption of electric vehicles using deep convolutional neural network to reduce driver's range anxiety," *ISA transactions*, vol. 98, pp. 454–470, 2020.