




Automatic Design System With Generative Adversarial Network and Convolutional Neural Network for Optimization Design of Interior Permanent Magnet Synchronous Motor

Yuki Shimizu , *Member, IEEE*, Shigeo Morimoto , *Member, IEEE*, Masayuki Sanada , *Member, IEEE*, and Yukinori Inoue, *Member, IEEE*

Abstract—The optimal design of interior permanent magnet synchronous motors requires a long time because finite element analysis (FEA) is performed repeatedly. To solve this problem, many researchers have used artificial intelligence to construct a prediction model that can replace FEA. However, because the training data are generated by FEA, it takes a very long time to obtain a sufficient amount of data, making it impossible to train a large-scale prediction model. Here, we propose a method for generating a large amount of data from a small number of FEA results using machine learning. An automatic design system with a deep generative model and a convolutional neural network is then constructed. With its sufficient data, the proposed system can handle three topologies and three motor parameters in a wide range of current vector regions. The proposed system was applied to multi-objective optimization design, with the optimization completed in 13–15 seconds.

Index Terms—Convolutional neural network, design optimization, generative adversarial network, permanent magnet motor, semisupervised learning.

I. INTRODUCTION

INTERIOR permanent magnet synchronous motors (IPMSMs) are widely adopted in electric vehicles and industrial robots because of their high output, efficiency, and reliability [1], [2], [3]. However, a major problem with IPMSMs is their long optimization period, which is caused by the high degree of freedom in their design and the use of finite element analysis (FEA). Many researchers are trying to solve this problem using optimization methods; some proposed automatic design with efficient optimization methods against

a high degree of freedom [4], [5], [6], [7], [8] while others proposed optimization methods with surrogate models to replace time-consuming FEA [9], [10], [11], [12], [13].

Because an IPMSM rotor can be designed with a large number of geometries due to the embedded permanent magnet (PM), various design alternatives thus need to be considered during optimization. Bonthu et al. [4] minimized torque ripple and cogging torque by optimizing the notch shape of the rotor surface of the permanent magnet assisted synchronous reluctance motor. Islam et al. [5] optimized two rotor design parameters at multiple output points of an IPMSM using the response surface method. Zheng et al. [6] performed multi-objective optimization of an IPMSM with rare-earth PMs and ferrite PMs using the response surface method. These size optimizations with computer-aided design (CAD) were effective for optimizing the shape for a given topology. However, it is difficult to deal with multiple topologies because the geometries depend on the initial shape. To solve this problem, many studies have proposed rotor design based on topology optimization. Ishikawa et al. [7] minimized PM volume using multi-material topology optimization for an asymmetric IPMSM rotor. Sato et al. [8] applied multi-material topology optimization to an IPMSM rotor using a normalized Gaussian network. Although these methods produce completely new topologies, some topologies cannot be manufactured and a very large number of candidate solutions must be considered in the optimization due to the huge design space. There is thus a need for a method for generating manufacturable design alternatives with multiple topologies in a small design variable space. Therefore, this study uses a generative adversarial network (GAN) to solve this problem. A GAN is a deep generative model that uses two deep neural networks proposed by Goodfellow et al. [14]. The GAN has the advantage of dimensionality reduction from multidimensional images to a small latent variable space, and it can integrally represent design alternatives for various motor topologies.

FEA, a numerical modeling method, is generally used to calculate the characteristics of an IPMSM. FEA can be used to obtain very accurate operating characteristics of IPMSMs, but it is time-consuming. Many researchers have thus investigated the construction of prediction models using artificial

Manuscript received 5 July 2022; revised 23 August 2022; accepted 17 September 2022. Date of publication 20 September 2022; date of current version 21 February 2023. This work was supported by JST, ACT-X, Japan, under Grant JPMJAX20AE. Paper no. TEC-00691-2022. (*Corresponding author: Yuki Shimizu.*)

Yuki Shimizu is with the College of Science and Engineering, Ritsumeikan University, Kusatsu, Shiga 5258577, Japan (e-mail: yshimizu@fc.ritsumeikan.ac.jp).

Shigeo Morimoto, Masayuki Sanada, and Yukinori Inoue are with the Graduate School of Engineering, Osaka Metropolitan University, Sakai, Osaka 5998531, Japan (e-mail: morimoto@eis.osakafu-u.ac.jp; sanada@eis.osakafu-u.ac.jp; inoue@eis.osakafu-u.ac.jp).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TEC.2022.3208129>.

Digital Object Identifier 10.1109/TEC.2022.3208129

intelligence (AI) to reduce analysis time while maintaining the accuracy of FEA. Dhulipati et al. [9] used support vector regression (SVR) to train a prediction model for a six-phase IPMSM. Hao et al. [10] trained a model to learn the relationship between the design parameters and torque ripple of an IPMSM using radial basis function networks, and used the model for optimization. Pan et al. [11] used XGBoost to learn the relationship between the torque characteristics and the structural parameters of permanent magnet arc motors, and used the model for optimization. These conventional machine-learning-based approaches are mainly used for size optimization. On the other hand, some studies use deep learning for topology optimization. Barmada et al. [12] used a convolutional neural network (CNN) to learn the relationship between the shape and torque characteristics of synchronous reluctance motors, and used the CNN for optimization. Asanuma et al. [13] trained a model to learn the relationship between the topology near the rotor surface and the torque characteristics of an IPMSM using transfer learning with CNNs. The CNN has the advantage of achieving highly accurate predictions from complex topological information by feature extraction, making them suitable for predicting the characteristics of design alternatives generated by a GAN.

These conventional AI-based modeling methods achieve high accuracy but take into account only specific geometries or one current vector condition, making them unsuitable for various applications. The small scale of these modeling techniques is due to the difficulty in obtaining sufficient data. For example, when training a prediction model to replace FEA, the training data are generally generated by FEA. Assuming that the FEA for generating training data takes 12.4 minutes to analyze the speed-torque characteristics of an IPMSM [15], an FEA of 100000 datasets would take more than two years.

This research solves the problem of obtaining sufficient data and aims to construct a general-purpose IPMSM design system that applies deep learning models to design and modeling. Fig. 1 shows an overview of the present study. First, for data acquisition, we use semi-supervised learning where the training data are generated using machine learning. In [15], the authors proposed a method for training a prediction model that can accurately predict the speed-torque characteristics of double-layered IPMSMs from a small number of design parameters and FEA results using machine learning. This prediction model can be used to calculate the operating characteristics of various double-layered IPMSMs from their design parameters to generate a large dataset in a short time. Thus, by constructing a machine learning model that is limited to a certain typical rotor shape from a small dataset, and applying the data generation process to various shapes, we can quickly obtain sufficient data.

Using the generated data, an automatic rotor core design system based on the GAN and the CNN is constructed. The two trained deep learning models are then used for multi-objective optimization design. It is shown that the design time is significantly reduced. The contributions of this paper can be summarized as follows.

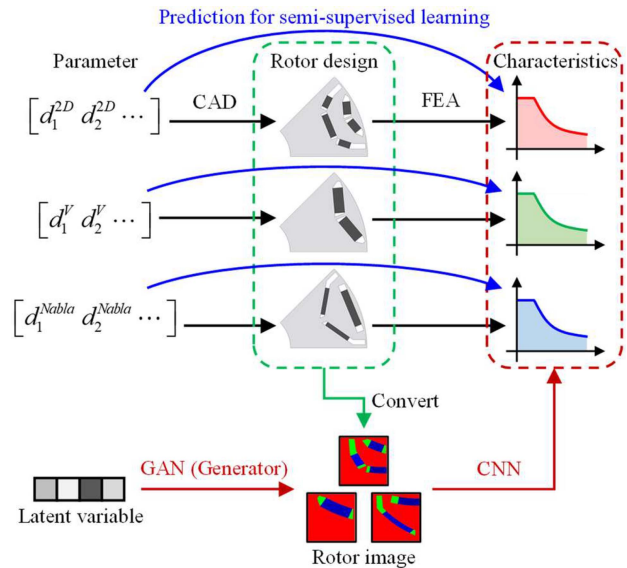


Fig. 1. Overview of this study. d_i^o is the design parameters for each rotor shape. Blue arrows represent the prediction flow used to generate the training data described in Section II. Red arrows represent the data flow of the proposed rotor core design system described in Section III.

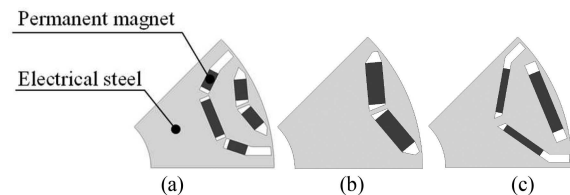


Fig. 2. Single-pole conventional rotor shapes. (a) 2D, (b) V, and (c) Nabla.

- A quick training data generation method for large-scale deep learning is proposed.
- A deep generative model that integrally represents different topologies in the latent space is applied.
- A prediction model of FEA that can be applied in a wide range of current vector regions is presented.
- An automatic design system for IPMSMs that enables quick design is proposed.

The rest of this paper is organized as follows. Section II describes the proposed method for quickly generating training data. Section III describes deep learning based on the generated dataset. Section IV shows the results of torque maximization and PM volume minimization using the proposed automatic design system. Section V summarizes the results.

II. OBTAINING TRAINING DATA

Semi-supervised learning with machine learning is used to quickly generate a sufficient amount of data for training the deep learning models. In this section, we describe the process of generating training data.

The target of this study is an IPMSM with 8 poles and 48 slots of distributed winding stators. Fig. 2 shows the rotor

topologies of 2D-, V-, and Nabla-structure IPMSMs [2], [16]. In this study, we generate data based on these three types of topologies. The permanent magnet is NMX-S49CH, and the iron core is 10JNEX900. See [2] for detailed information on the stator geometry, body size, and other parameters.

First, the relationship between rotor shape and speed-torque characteristics is learned using the method proposed in [15]. For the prediction model in [15], the input variables are the design parameters, such as the PM thickness, and current conditions. Although the objective is to predict the speed-torque characteristics, the prediction models do not directly predict the speed-torque characteristics; instead, we set three motor parameters (PM flux linkage Ψ_a and d - and q -axis inductances L_d, L_q) as the prediction targets to improve accuracy. The average torque T and the limit speed N_{lim} are obtained from the predicted motor parameters as follows.

$$T = P_n (\Psi_a i_q + (L_q - L_d) i_d i_q), \quad (1)$$

$$N_{lim} = \frac{V_{om}}{\sqrt{(\Psi_a + L_d i_d)^2 + (L_q i_q)^2}}, \quad (2)$$

where P_n is the number of pole pairs, i_d and i_q are the d - and q -axis current, respectively, and V_{om} is the maximum induced voltage.

The FEA conditions for the training data for the prediction models were generated according to (3).

$$\begin{cases} d_i \sim U(d_{lwr}, d_{upr}), \\ I_a \sim 140\sqrt{3}U(0, 1)(A), \\ \beta \sim U(0, 90)(^\circ), \end{cases} \quad (3)$$

where $U(a, b)$ is a random variable with a uniform distribution on an open interval (a, b) , d_i is the i -th design parameter, and d_{lwr} and d_{upr} are its lower and upper bound, respectively. The numbers of design parameters are 11, 5, and 8 for 2D, V, and Nabla, respectively. The relationship between the armature current I_a , the current phase angle β and i_d, i_q are as follows.

$$\begin{cases} i_d = -I_a \sin \beta \\ i_q = I_a \cos \beta \end{cases} \quad (4)$$

The shapes were generated by CAD based on the randomly generated design parameters, and FEA was performed under the randomly generated current conditions to generate the training data for prediction models. The objective of the prediction models is to accurately learn the function f in the following equation. See [15] for more details.

$$\Psi_a, L_d, L_q = f(i_d, i_q, d_1, d_2, \dots) \quad (5)$$

The prediction models described above are constructed for the three topologies. Table I shows the machine learning methods used to predict the characteristics for each topology, where GPR is Gaussian process regression. The values in parentheses are the coefficients of determination r^2 for the test data. For the training data, we used a total of 26209 randomly generated shapes (2D: 8256; V: 7927; Nabla: 10026), all of which were analyzed under random current vector conditions. The software JMAG-Designer 19.1 was used for the FEA. The prediction

TABLE I
MACHINE LEARNING METHODS AND COEFFICIENTS OF DETERMINATION FOR TEST DATA USED FOR DATA GENERATION

Topology \ Parameter	Ψ_a	L_d	L_q
2D	SVR ($r^2 = 0.997$)	SVR ($r^2 = 0.963$)	XGBoost ($r^2 = 0.998$)
V	SVR ($r^2 = 1.000$)	SVR ($r^2 = 0.979$)	SVR ($r^2 = 0.999$)
Nabla	GPR ($r^2 = 0.869$)	SVR ($r^2 = 0.725$)	XGBoost ($r^2 = 0.986$)

TABLE II
NUMBER OF DATASETS USED FOR TRAINING

Topology \ Purpose	For data generation	For training GAN and CNN		
		Training	Validation	Test
2D	8,256	40,000	10,000	5,000
V	7,927	× 3 topol.	× 3 topol.	× 3 topol.
Nabla	10,026	= 120,000	= 30,000	= 15,000
How to obtain	FEA	Prediction models		

accuracies for Nabla were the lowest among the three topologies, despite the largest number of training data. A detailed discussion of the differences in prediction accuracy between topologies is beyond the focus of this study and will be the subject of future research, but the effects of the imbalance in prediction accuracy for each topology and each parameter are described in Section IV.

The trained prediction models and the randomly generated conditions according to (3) were then substituted into (5) to generate shape and motor parameter pairs for 55000 shapes for each topology, for a total of 165000 shapes. Table II summarizes the number of datasets used for training in each phase. The reason for generating 55000 shapes is to stabilize the images generated by the GAN explained in Section III-B. Because the motor parameters change nonlinearly due to the effect of magnetic saturation, we predicted the change in motor parameters versus the current vector condition for each shape. Because the maximum armature current of the IPMSM used in this research is 232 A, the characteristic data generated by the prediction are the discrete PM flux linkage with an armature current ranging from 5 to 235 A in 5-A increments and the discrete d - and q -axis current versus d - and q -axis inductance characteristics with d - and q -axis currents ranging from 5 to 235 A in 10-A increments. Here, the PM flux linkage is assumed to be independent of the current phase [15].

The data augmentation method described above predicted the characteristics for 623 current vector conditions for each shape, meaning that the FEA results for 102795000 conditions (165000 shapes × 623 conditions/shape) were predicted from FEA results for only 24000 conditions. The prediction of 102795000 data points was completed in a total of 3.6 hours, thus concluding that sufficient data were obtained in a practical amount of time.

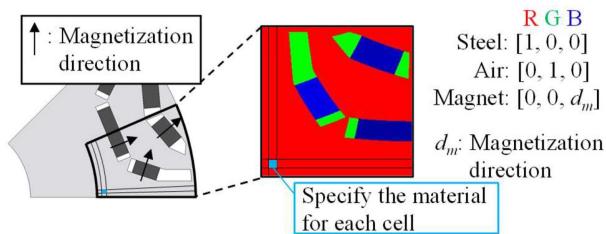


Fig. 3. Material representation of rotor shape.

III. AUTOMATIC DESIGN SYSTEM

Using the dataset created in Section II, we trained the deep learning models for the automatic rotor shape design system. The automatic design system consists of two types of deep learning model, one for design and the other for characteristic prediction. This section describes the training methods and the results for these two deep learning models.

A. Material Representation for Rotor Core

First, we describe the numerical representation of the motor shape. There are two types of numerical representation of motor shapes, namely that used in [15] (see Section II) that specifies the design parameters of the shape, denoted as the parameter representation, and that used in topology optimization that specifies the material at each coordinate, denoted as the material representation. The parameter representation can represent only one topology depending on the reference shape and is unsuitable for a system that handles multiple topologies in an integrated manner. For example, the design parameters used for the 2D, V, and Nabla structures are 11-, 5-, and 8-dimensional, respectively, making it difficult to handle different topologies with a given parameter representation. Therefore, in the proposed automatic design system, the motor shape is represented numerically by the material representation.

Fig. 3 shows the material representation method used in the proposed system. An electromagnetic steel sheet, a PM, or air is specified for each pole coordinate of the rotor, and the three materials are assigned to the RGB (red, green, blue) values of the image as one-hot vectors, respectively, to represent the rotor shape in the image, as shown in the right image in Fig. 3. Because the shape considered in this study is d -axis symmetric and there is no magnet or air layer near the shaft, only half of the geometry in the circumferential direction and 60% of the geometry in the radial direction are converted into images. The magnetization direction (angle) of the PM is represented by the difference in the brightness of the blue color by inputting the normalized value of the angle information d_{PM} , which is calculated as follows:

$$d_{PM} = \frac{\theta_{PM} + 90}{180}, \quad (6)$$

where $\theta_{PM} \in [-90^\circ, 90^\circ]$ is the angle of the magnetization direction of each PM. In the actual image generation process for the material representation, the material information at each coordinate was extracted and converted into an image for the rotor shape whose dimensions were parametrically generated

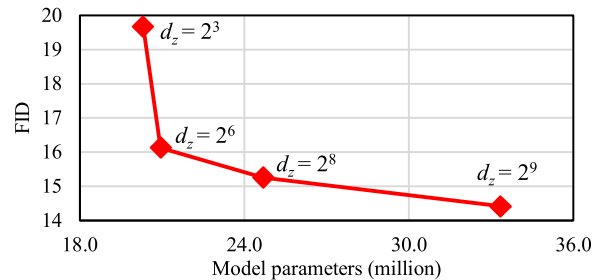
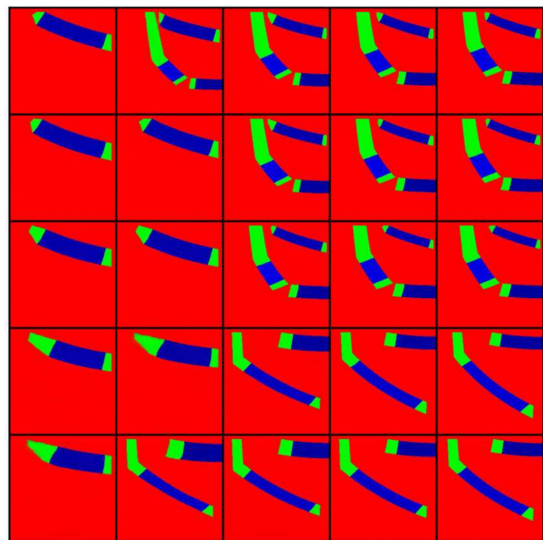
Fig. 4. Parameters-FID characteristics. d_z is the dimension of the latent variable. The batch size was 10, and the number of training steps was 200000.

Fig. 5. Example of results generated by lightweight GAN. The dimension of the latent variable was 256.

by CAD in JMAG-Designer. Note that due to the coordinate transformation, the straight part of the rotor shape becomes a gentle curve in the image.

B. Training of Generative Adversarial Network

This study uses lightweight GAN [17] to generate materially represented rotor images. The training data for the lightweight GAN is the converted 165000 images from the shapes parametrically generated in Section II. The image was a $3 \times 256 \times 256$ tensor.

To determine the parameters of GAN, this study uses Fréchet Inception Distance (FID) [18]. FID measures the overall semantic realism of the synthesized images. We let GAN randomly generate 50000 images and computed FID between the generated images and the whole training dataset. Fig. 4 shows the number of parameters of the models and the calculated FID in the different latent variable dimensions. To reduce the FID, the dimension of the latent variable needs to be increased, but the computational cost also increases with the number of parameters. Thus, the dimension of the latent variable was set to $2^8 = 256$.

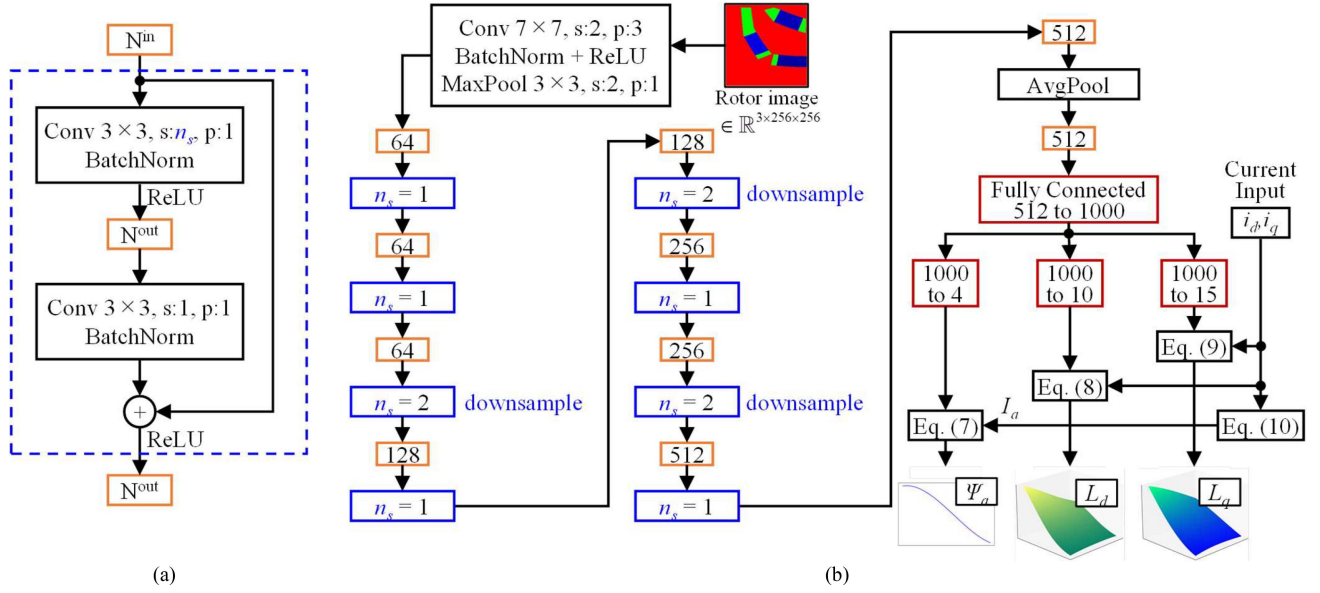


Fig. 6. Architecture of multi-task CNN. (a) Residual block and (b) overall network. Orange boxes represent feature maps, where the number of channels is shown. Blue boxes mean residual blocks shown in (a), and if "downsample" is written, a convolutional layer that has 1x1 filters and a stride of 2 is added to the shortcut connection. Red boxes mean the fully connected layers.

Fig. 5 shows an example of a shape generated by lightweight GAN. The output images of the GAN clearly show the three types of rotor shape. All of these images were sampled from the same latent variable space, indicating that the use of GAN allows us to handle a wide variety of shapes in a unified manner. In addition, the huge design space can be reduced to 256 dimensions of the latent variable space and undesignable shapes can be eliminated.

C. Training of Convolutional Neural Network

This study uses CNN to predict the motor characteristics from rotor images generated by the GAN. Fig. 6 shows the architecture of the CNN used in this study. The regression CNN is built based on ResNet-18 [19]. It is a multi-task learning architecture that simultaneously predicts three motor parameters for a single shape. Because each motor parameter is nonlinear with respect to the current vector, the characteristic data generated in Section II were approximated by polynomial equations using the least-squares method. The coefficients of the approximation equation were used as the prediction target. The approximation equations are shown below.

$$\Psi_a = w_0^{\Psi_a} + w_1^{\Psi_a} I_a + w_2^{\Psi_a} I_a^2 + w_3^{\Psi_a} I_a^3, \quad (7)$$

$$L_d = w_0^{L_d} + w_1^{L_d} i_d + w_2^{L_d} i_q + w_3^{L_d} i_d^2 + w_4^{L_d} i_d i_q + w_5^{L_d} i_q^2 + w_6^{L_d} i_d^3 + w_7^{L_d} i_d^2 i_q + w_8^{L_d} i_d i_q^2 + w_9^{L_d} i_q^3, \quad (8)$$

$$L_q = w_0^{L_q} + w_1^{L_q} i_d + w_2^{L_q} i_q + w_3^{L_q} i_d^2 + w_4^{L_q} i_d i_q + w_5^{L_q} i_q^2 + w_6^{L_q} i_d^3 + w_7^{L_q} i_d^2 i_q + w_8^{L_q} i_d i_q^2 + w_9^{L_q} i_q^3 + w_{10}^{L_q} i_d^4 + w_{11}^{L_q} i_d^3 i_q + w_{12}^{L_q} i_d^2 i_q^2 + w_{13}^{L_q} i_d i_q^3 + w_{14}^{L_q} i_q^4, \quad (9)$$

where w_i^p ($p \in \{\Psi_a, L_d, L_q\}$) is the coefficient, and I_a is calculated as follows:

$$I_a = \sqrt{i_d^2 + i_q^2}. \quad (10)$$

As shown in Fig. 6, the number of output nodes in fully connected layers was determined according to the number of coefficients in (4)–(6).

The true values of the coefficients of each motor parameter for 16500 shapes were calculated to minimize the squared errors from the motor parameters in each of the 623 conditions for each shape generated in Section II. From the normalized 165000 datasets (combinations of shapes and approximation equation coefficients), 120000 were used as training data, 30000 were used as validation data, and 15000 were used as test data for training. Adam was used for optimization, the weight decay was 0.0001, the learning rate was 0.001, and the mini-batch size was 16. The loss function \mathcal{L}_{CNN} is defined as follows.

$$\mathcal{L}_{CNN} = \sum_{p \in \{\Psi_a, L_d, L_q\}} k^p MSE(\hat{\mathbf{w}}^p, \mathbf{w}^p), \quad (11)$$

where k^p is the coefficient for balancing losses, MSE is a function that returns the mean squared error, \mathbf{w}^p is the true value of the weight vector for each parameter, and $\hat{\mathbf{w}}^p$ is the CNN prediction of \mathbf{w}^p . In this study, we set the coefficient ($k^{\Psi_a}, k^{L_d}, k^{L_q}$) = (3, 2, 1).

Fig. 7 shows the prediction accuracy of the trained multi-task CNN on the test data, and Fig. 8 shows the training and validation errors. Note that this prediction accuracy is not for the FEA results, but for the data generated by machine learning. First, the validation error was almost the same as the training error, concluding in no tendency to overfit. A comparison of the prediction results indicates that the accuracy was high except for

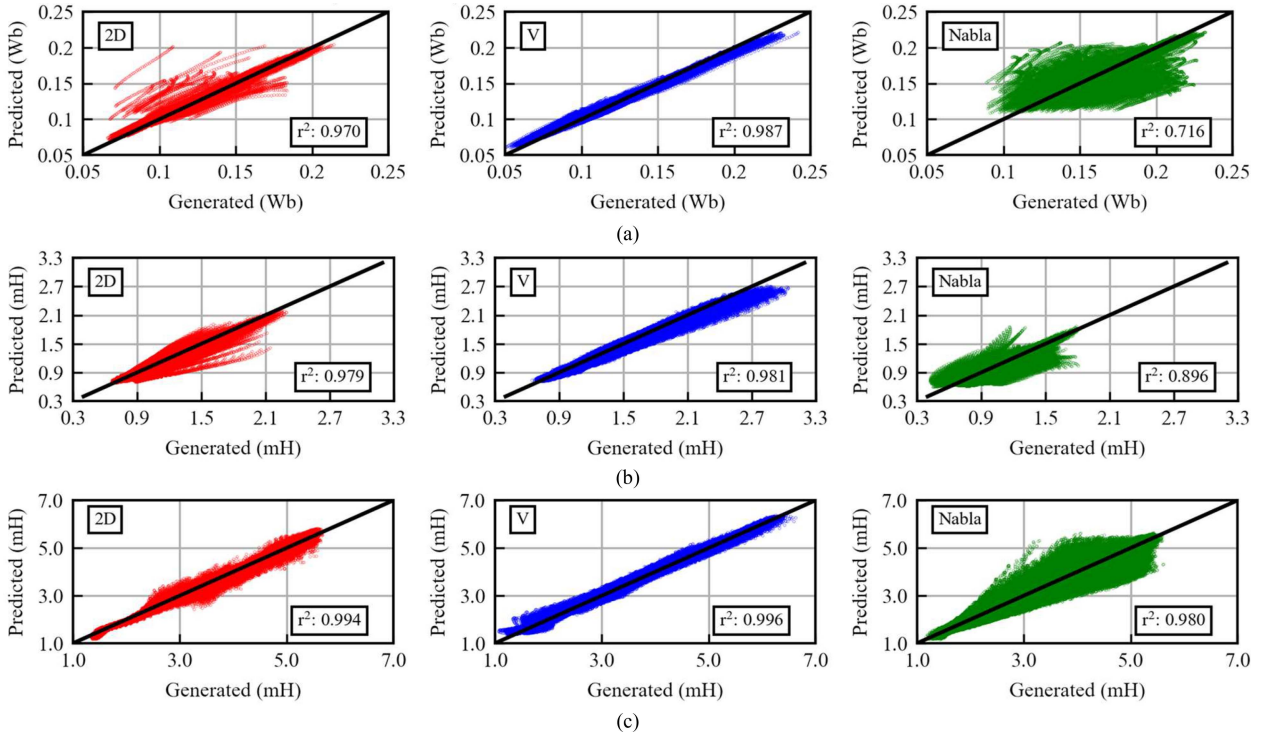


Fig. 7. Prediction results for test data, where r^2 is the coefficient of determination. Results for (a) PM flux linkage, (b) d -axis inductance, and (c) q -axis inductance.

TABLE III

COMPARISON OF PREDICTION ERRORS AND MODEL PARAMETERS FOR DIFFERENT CNNs (MEAN \pm STD FOR 10 TRAINING RUNS; STD: STANDARD DEVIATION)

Model	This study (ResNet-18)	ResNet-34 [19]	ResNet-50 [19]	VGG-11 [20] (w/ batch normalization)	AlexNet [21]
Validation error of Ψ_a	0.0129 \pm 0.0017	0.0124 \pm 0.0022	0.0153 \pm 0.0022	0.0301 \pm 0.0021	0.2226 \pm 0.3864
Validation error of L_d	0.0278 \pm 0.0032	0.0260 \pm 0.0031	0.0370 \pm 0.0043	0.0687 \pm 0.0048	0.2556 \pm 0.3663
Validation error of L_q	0.0557 \pm 0.0027	0.0543 \pm 0.0041	0.0621 \pm 0.0044	0.1037 \pm 0.0042	0.2777 \pm 0.3592
Model parameters (million)	11.7	21.8	25.6	132.9	61.1

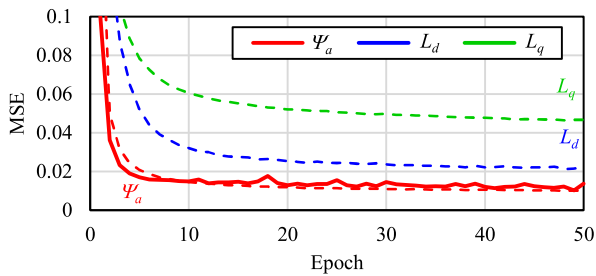


Fig. 8. Training and validation errors. Dashed curves denote training error, and bold curves denote validation error.

the PM flux linkage and d -axis inductance in the Nabla structure. The low prediction accuracy of these two characteristics is due to the prediction error generated during data generation because their prediction accuracy in Table I is also low. Because a perfect prediction of the data with errors implies a deviation from the true data by FEA, this result in Fig. 7 is reasonable. Prediction accuracy for FEA is discussed in Section IV.

To demonstrate the high performance of the proposed CNN, we compare its performance with other conventional CNNs, ResNet-34 [19], ResNet-50 [19], VGG-11 [20] with batch normalization, and AlexNet [21]. Table III compares the prediction results and the model parameters for 10 training runs, where only the networks on the input side from the 1000-dimensional fully connected layer were changed while keeping the multi-task architecture in Fig. 6 fixed. All the training parameters were the same. The proposed CNN based on ResNet-18 and the CNN based on ResNet-34 had nearly equal validation errors for all motor parameters and had lower prediction errors than the other CNNs. In addition, the proposed model has fewer parameters and then requires less training time. Therefore, the proposed architecture is superior in both performance and computational cost.

Then, to verify the effectiveness of the proposed CNN's multi-task architecture, we compare the prediction accuracy of the CNNs when trained with different loss coefficients. Table IV compares their prediction results for 10 training runs, where one-hot vectors mean that the learning is not multitasking. Comparison results show that the multi-task architecture reduces

TABLE IV
COMPARISON OF PREDICTION ERRORS AND MODEL PARAMETERS FOR DIFFERENT LOSS COEFFICIENTS (MEAN \pm STD FOR 10 TRAINING RUNS)

$(k^{\Psi_a}, k^{L_d}, k^{L_q})$	This study (3,2,1)	(1,1,1)	(1,0,0); (Ψ_a only)	(0,1,0); (L_d only)	(0,0,1); (L_q only)
Validation error of Ψ_a	0.0129 \pm 0.0017	0.0163 \pm 0.0023	0.0178 \pm 0.0035	-	-
Validation error of L_d	0.0278 \pm 0.0032	0.0337 \pm 0.0060	-	0.0420 \pm 0.0122	-
Validation error of L_q	0.0557 \pm 0.0027	0.0548 \pm 0.0046	-	-	0.0635 \pm 0.0071

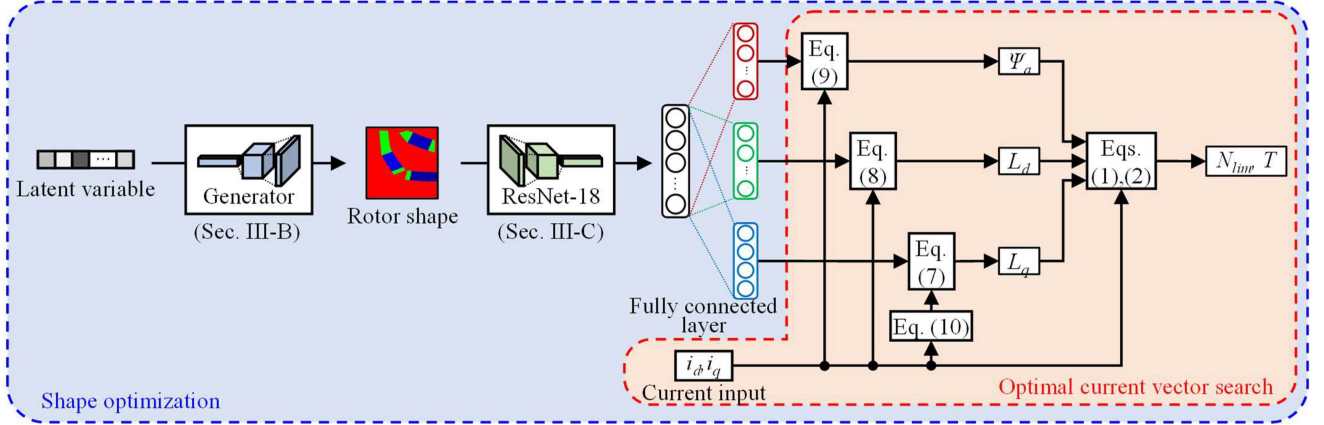


Fig. 9. Overall configuration of automatic design system. The blue part is used for shape optimization, and the red part is used for optimal current vector search for a given shape.

the verification errors for all motor parameters due to the suppression effect of overfitting. Furthermore, the setting with a larger weight of errors in Ψ_a and L_d efficiently reduced the verification error, although a slightly larger verification error in L_q .

IV. OPTIMIZATION DESIGN

The combination of the deep generative model and the characteristic prediction model in Section III leads to an automatic design system, as shown in Fig. 9. This section demonstrates the usefulness of the system by performing a multi-objective optimization design in the 256-dimensional latent variable space of the generative model.

A. Problem

In this study, the design goals are to minimize the volume of the PMs and maximize the maximum torque under the torque constraint. The problem setup is as follows:

$$\begin{aligned} \min \quad & w_1 \frac{V^{PM}}{V_{init}^{PM}} - w_2 \frac{T_{pred}^{Max}}{T_{init}^{Max}}, \\ \text{s.t.} \quad & g_i : T_{pred}^{(i)} \geq \alpha T_{req}^{(i)} \quad (i = 1, 2, \dots, n), \end{aligned} \quad (12)$$

where V^{PM} is the volume of a PM for each candidate solution and T_{pred}^{Max} is the predicted maximum torque for each candidate solution. These parameters are normalized by the initial values V_{init}^{PM} and T_{init}^{Max} , respectively. The volume of the PM was defined as a percentage of the image area. w_1 and w_2 are weight coefficients. $(w_1, w_2) = (1, 1)$ in this setup. The constraint

condition g_i is a torque constraint for n required operating points $\{(N_{req}^{(i)}, T_{req}^{(i)})\}_{i=1}^n$, which is multiplied by a coefficient $\alpha = 1.03$ to consider the prediction error. The torque prediction results are given as the results of maximum power control at the required speed as follows:

$$\begin{aligned} T_{pred}^{(i)} = \max_{I_a \in (0, I_{am}], \beta \in (0, 90)} T_{CNN}(I_a, \beta), \\ \text{s.t.} \quad N_{CNN}(I_a, \beta) \geq N_{req}^{(i)}, \end{aligned} \quad (13)$$

where I_{am} is the maximum armature current, and T_{CNN} and N_{CNN} are the torque and limit speed calculated by substituting the motor parameters predicted by the CNN into (1) and (2), respectively. The solution for maximum power control was obtained by a brute-force search.

NSGA-II [22] was used as the optimization algorithm, and the framework pymoo was used for the implementation [23]. The population size was set to 100 and the number of offspring was set to 10. Latin hypercube sampling was used for sampling the initial population, the tournament method was used for selection, simulated binary crossover was used for crossover, and polynomial mutation was used for mutation. The termination condition was set to 100 generations.

B. Optimization Results

To verify the robustness of the proposed system, this study performed optimization under three conditions. Table V shows the settings for the three conditions and Figs. 10 and 11 show

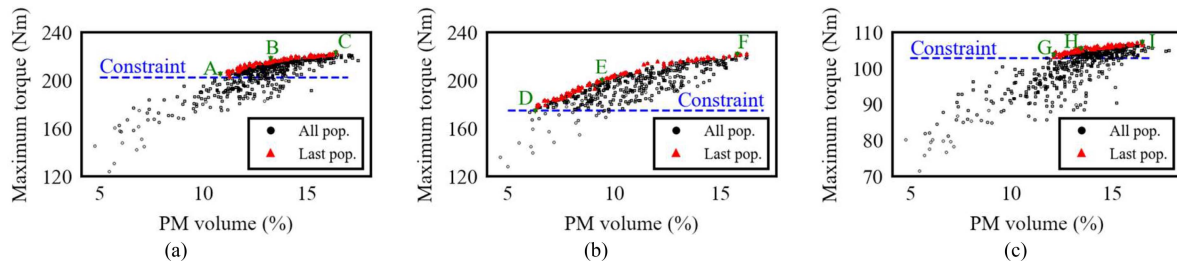


Fig. 10. Characteristics of all populations and the last population for optimization design under (a) Condition 1, (b) Condition 2, and (c) Condition 3.

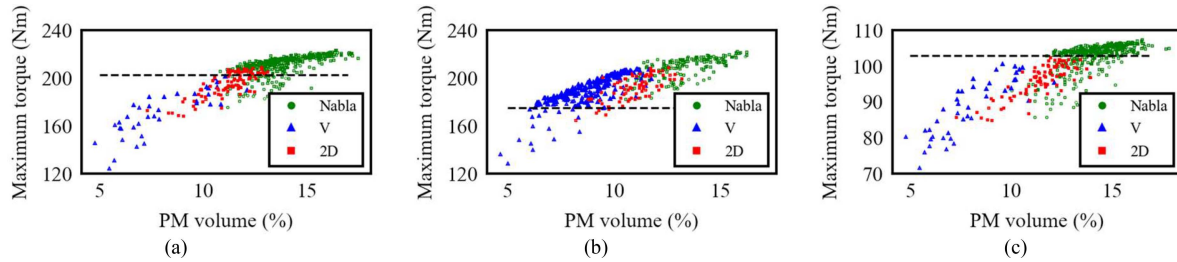


Fig. 11. Characteristics of all populations with topology information for optimization design under (a) Condition 1, (b) Condition 2, and (c) Condition 3.

TABLE V
OPTIMIZATION CONDITIONS

Condition No.	Required operating points	Maximum armature current I_{am}
Condition 1	(3,000 r/min, 197 Nm)	232 A
	(11,000 r/min, 40 Nm)	
Condition 2	(1,000 r/min, 170 Nm)	232 A
	(11,000 r/min, 40 Nm)	
Condition 3	(1,000 r/min, 100 Nm)	104 A
	(9,000 r/min, 30 Nm)	

the characteristics of all the populations generated in the optimization process under each condition. The maximum armature voltage was set to 507 V.

In Condition 1, the maximum armature current was set to 232 A, and the optimization was performed at two required operating points, (3000 min^{-1} , 197 Nm) and (11000 min^{-1} , 40 Nm), which were determined based on the reference motor in [2]. The solution population transitioned to satisfy the constraints; all the individuals in the final population satisfied the constraints. The Nabla structure most easily produced the maximum torque. Almost all of the Pareto solutions are the Nabla structure under severe torque requirements.

In Condition 2, the maximum armature current was fixed at 232 A and the torque constraint was relaxed from that in Condition 1. Optimization was performed at two required operating points, (3000 min^{-1} , 170 Nm) and (11000 min^{-1} , 40 Nm). In Condition 2, many candidate solutions for all three topologies satisfy the relaxed torque constraint. The V structure reduced the volume of PMs the most while maintaining high torque. The 2D structure, which was designed for high efficiency [2], does not appear in the Pareto solution for Condition 2.

TABLE VI
COMPARISON OF OPTIMIZATION ALGORITHM (MEAN \pm STD FOR 10 RUNS)

Algorithm	Minimum fitness of last population
This study (NSGA-II)	-0.343 \pm 0.025
RNSGA-II [24]	-0.339 \pm 0.017
NSGA-III [25]	-0.333 \pm 0.019
UNSGA-III [26]	-0.339 \pm 0.026
RNSGA-III [27]	-0.336 \pm 0.022

In Condition 3, the armature current limit was reduced from 232 to 104 A and the two required operating points were set to (1000 min^{-1} , 100 Nm) and (9000 min^{-1} , 30 Nm). In this condition, the maximum torque requirement was reduced, but the current limit was also reduced, resulting in a tighter torque requirement. In addition, as in Condition 1, many Nabla structures were selected to easily obtain the torque. Thus, the optimization design of the proposed system can be performed under various current limits.

To verify the optimality of the obtained solutions, we performed comparative experiments with different multi-objective optimization algorithms, RNSGA-II [24], NSGA-III [25], UNSGA-III [26], and RNSGA-III [27]. Table VI compares the minimum fitness of the last population for each optimization, where the reference points for RNSGA-II and RNSGA-III were set to (0.82, -1.17) and (0.68, -1.07), and the reference directions for NSGA-III and UNSGA-III were set in 8 segments based on Das-Dennis method. The other basic optimization parameters were common. The comparison results showed that NSGA-II used is superior to the other multi-objective optimization methods in this problem set.

FEA was conducted by selecting three candidate solutions for each condition from the Pareto solutions. Fig. 12 shows the

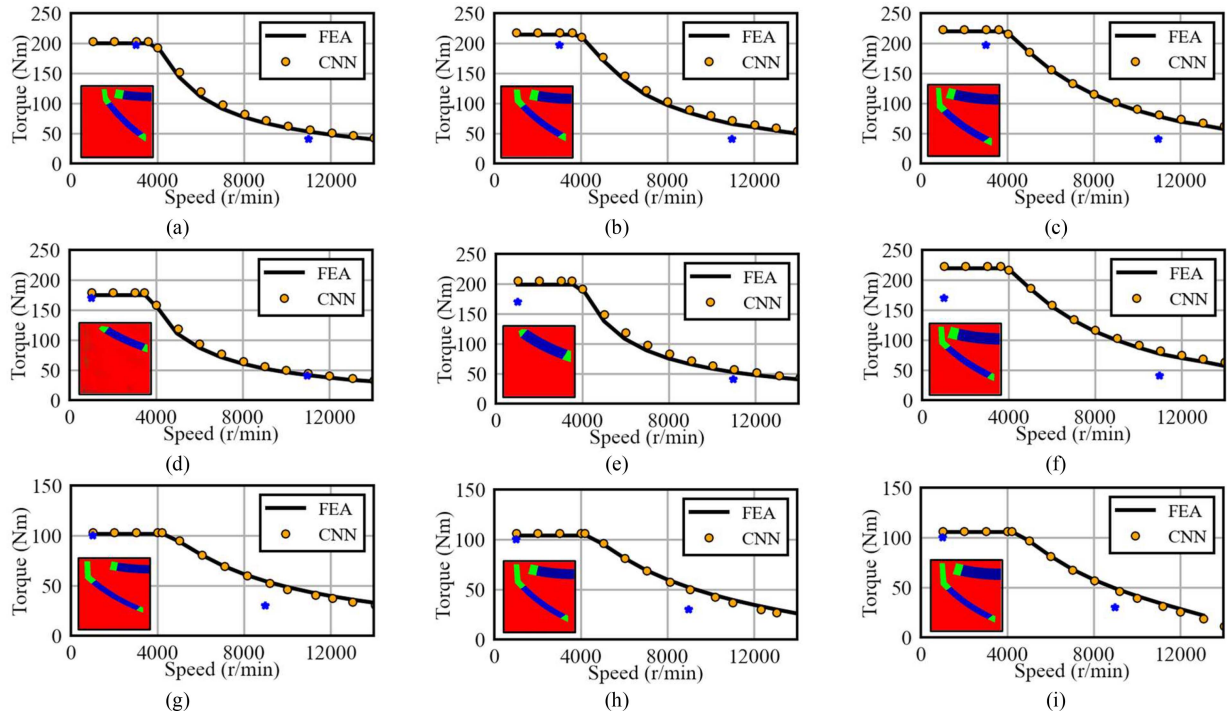


Fig. 12. Prediction by proposed method and FEA results for speed-torque characteristics of Pareto solutions, where blue points are required operating points. (a)–(i) Solutions A–I defined in Fig. 9.

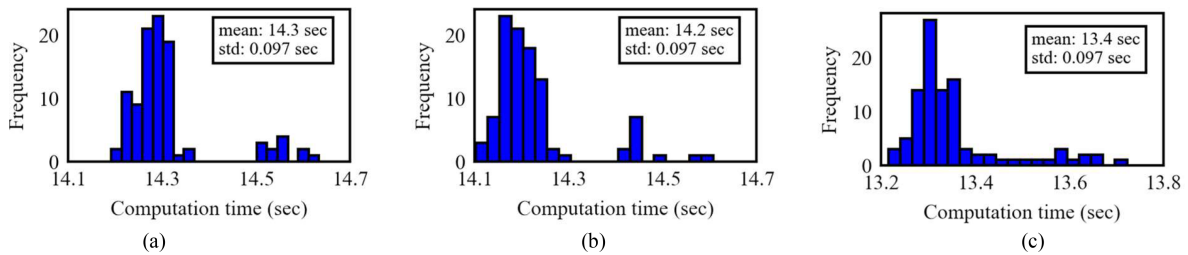


Fig. 13. Optimization time for (a) Condition 1, (b) Condition 2, and (c) Condition 3.

prediction results and the FEA results for the speed-torque characteristics of the selected candidate solutions, where solutions A–I correspond to the solutions in Fig. 10 and the blue points represent the required operating points for each condition.

The shape images of the Pareto solutions are all clear and designable, confirming the effectiveness of the trained GAN. A comparison of the FEA results with the CNN prediction results indicates that the prediction accuracy of the speed-torque characteristics is very high for all candidate solutions. The FEA results for all Pareto solutions satisfy the required operating point, which means that the prediction error is less than 3%. These results imply that the poor prediction accuracy for the Nabla structure in Table I was improved by the CNN. In other words, the CNN recognized and eliminated as noise the insensitive error and standard error generated in the process of data generation by SVR and GPR, respectively. This shows that the proposed data generation method can tolerate a certain degree of inaccuracy in a machine learning method.

Finally, we discuss the optimization time. Fig. 13 shows a histogram of the design time for 100 optimization designs under each condition. For the calculations, a computer with an Intel Core i7-9700K CPU, 32.0 GB of RAM, and an NVIDIA GeForce RTX 2070 SUPER (8 GB) GPU was used. The proposed system can design a shape that satisfies the requirements in 13–15 seconds, effectively reducing the optimization time compared to that for the conventional optimization calculation for the same scale (generally several days to several weeks).

V. CONCLUSION

This paper proposed a deep learning technique for optimizing IPMSM rotors. The results can be summarized as follows.

- a) We proposed a method for quickly generating a large amount of FEA data for training large-scale deep learning models using machine learning specific to each topology. This method generated 102795000 training data from

26209 FEA results. Also, the proposed data generation method could tolerate a certain degree of inaccuracy in a machine learning method.

- b) The proposed generative model can be used to design rotor topologies for three IPMSMs with high precision and can represent different topologies in a unified 256-dimensional latent variable space.
- c) We proposed a prediction model that can quickly and accurately predict motor parameters in a wide range of current vectors with various geometries.
- d) We proposed an automatic rotor design system for IPMSMs using two deep learning models. The proposed system can be used for various required operating points and current ranges. The time required for design optimization was only 13-15 seconds.

Once the proposed automatic design system is trained, the design optimization under various conditions can be performed many times in a short time, leading to a significant reduction in the design and development time of IPMSMs. In the future, we will study the characteristics not covered in this paper, such as loss and vibration. Additionally, other magnetic materials and rotor topologies will be examined to expand the scope of the proposed automatic design system.

The dataset generated in Section II is available at IEEE DataPort [28] and the Python implementation of the characteristics prediction models is available at GitHub [29].

REFERENCES

- [1] A. Brosch, O. Wallscheid, and J. Böcker, "Torque and inductances estimation for finite model predictive control of highly utilized permanent magnet synchronous motors," *IEEE Trans. Ind. Inform.*, vol. 17, no. 12, pp. 8080–8091, Dec. 2021.
- [2] Y. Shimizu, S. Morimoto, M. Sanada, and Y. Inoue, "Influence of permanent magnet properties and arrangement on performance of IPMSMs for automotive applications," *IEEJ J. Ind. Appl.*, vol. 6, no. 6, pp. 401–408, Nov. 2017.
- [3] Y.-H. Jung, M.-R. Park, K.-O. Kim, J.-W. Chin, J.-P. Hong, and M.-S. Lim, "Design of high-speed multilayer IPMSM using ferrite PM for EV traction considering mechanical and electrical characteristics," *IEEE Trans. Ind. Appl.*, vol. 57, no. 1, pp. 327–339, Jan. 2021.
- [4] S. S. R. Bonthu, M. T. B. Tarek, and S. Choi, "Optimal torque ripple reduction technique for outer rotor permanent magnet synchronous reluctance motors," *IEEE Trans. Energy Convers.*, vol. 33, no. 3, pp. 1184–1192, Sep. 2018.
- [5] M. S. Islam, M. Chowdhury, A. Shrestha, M. Islam, and I. Husain, "Multiload point optimization of interior permanent magnet synchronous machines for high-performance variable-speed drives," *IEEE Trans. Ind. Appl.*, vol. 57, no. 1, pp. 427–436, Jan. 2021.
- [6] S. Zheng, X. Zhu, L. Xu, Z. Xiang, L. Quan, and B. Yu, "Multi-objective optimization design of a multi-permanent-magnet motor considering magnet characteristic variation effects," *IEEE Trans. Ind. Electron.*, vol. 69, no. 4, pp. 3428–3438, Apr. 2022.
- [7] T. Ishikawa, S. Mizuno, and N. Krita, "Topology optimization method for asymmetrical rotor using cluster and cleaning procedure," *IEEE Trans. Mag.*, vol. 53, no. 6, Jun. 2017, Art. no. 7001504.
- [8] T. Sato, K. Watanabe, and H. Igarashi, "Multimaterial topology optimization of electric machines based on normalized Gaussian network," *IEEE Trans. Mag.*, vol. 51, no. 3, Mar. 2015, Art. no. 7202604.
- [9] H. Dhulipati, E. Ghosh, S. Mukundan, P. Korta, J. Tjong, and N. C. Kar, "Advanced design optimization technique for torque profile improvement in six-phase PMSM using supervised machine learning for direct-drive eV," *IEEE Trans. Energy Convers.*, vol. 34, no. 4, pp. 2041–2051, Dec. 2019.
- [10] J. Hao, S. Suo, Y. Yang, Y. Wang, W. Wang, and X. Chen, "Optimization of torque ripples in an interior permanent magnet synchronous motor based on the orthogonal experimental method and MIGA and RBF neural networks," *IEEE Access*, vol. 8, pp. 27202–27209, 2020.
- [11] Z. Pan and S. Fang, "Torque performance improvement of permanent magnet arc motor based on two-step strategy," *IEEE Trans. Ind. Inform.*, vol. 17, no. 11, pp. 7523–7534, Nov. 2021.
- [12] S. Barmada, N. Fontana, L. Sani, D. Thomopoulos, and M. Tucci, "Deep learning and reduced models for fast optimization in electromagnetics," *IEEE Trans. Mag.*, vol. 56, no. 3, Mar. 2020, Art. no. 7513604.
- [13] J. Asanuma, S. Doi, and H. Igarashi, "Transfer learning through deep learning: Application to topology optimization of electric motor," *IEEE Trans. Mag.*, vol. 56, no. 3, Mar. 2020, Art. no. 7513604.
- [14] I. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
- [15] Y. Shimizu, S. Morimoto, M. Sanada, and Y. Inoue, "Using machine learning to reduce design time for permanent magnet volume minimization in IPMSMs for automotive applications," *IEEJ J. Ind. Appl.*, vol. 10, no. 5, pp. 554–563, 2021.
- [16] S. Suzuki, S. Morimoto, M. Sanada, and Y. Inoue, "Performance comparison of IPMSMs using a low iron loss material for automotive application," in *Proc. IEEE 19th Int. Conf. Elect. Machines Syst.*, 2016, pp. 839–844.
- [17] B. Liu, Y. Zhu, K. Song, and A. Elgammal, "Towards faster and stabilized GAN training for high-fidelity few-shot image synthesis," in *Proc. Int. Conf. Learn. Representations*, 2021. [Online]. Available: <https://openreview.net/forum?id=1Fqg133qRal>
- [18] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local nash equilibrium," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6626–6637.
- [19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Int. Conf. Learn. Representations*, 2016, pp. 770–778.
- [20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 409–420.
- [21] A. Krizhevsky, "One weird trick for parallelizing convolutional neural networks," 2014, *arXiv:1404.5997*.
- [22] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [23] J. Blank and K. Deb, "Pymoo: Multi-objective optimization in python," *IEEE Access*, vol. 8, pp. 89497–89509, 2020.
- [24] K. Deb and J. Sundar, "Reference point based multi-objective optimization using evolutionary algorithms," in *Proc. 8th Annu. Conf. Genet. Evol. Computation*, Seattle, WA, USA, 2006, pp. 635–642.
- [25] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014.
- [26] H. Seada and K. Deb, "A unified evolutionary optimization procedure for single, multiple, and many objectives," *IEEE Trans. Evol. Comput.*, vol. 20, no. 3, pp. 358–369, Jun. 2016.
- [27] Y. Vesikar, K. Deb, and J. Blank, "Reference point based NSGA-III for preferred solutions," in *Proc. IEEE Symp. Ser. Comput. Intell.*, 2018, pp. 1587–1594.
- [28] Y. Shimizu, "Dataset for motor parameters of IPMSM" IEEE dataport," Oct. 2021. [Online]. Available: <https://iee-dataport.org/documents/dataset-motor-parameters-ipmsm>
- [29] Y. Shimizu, "IPMSM-automatic-design," Oct. 12, 2021. [Online]. Available: <https://github.com/yshimizu12/IPMSM-automatic-design>



Yuki Shimizu (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Osaka Prefecture University, Sakai, Japan, in 2016, 2018, and 2022, respectively. In 2018, he joined Toyota Motor Corporation, Aichi, Japan. Since 2022, he has been with the Graduate School of Science and Engineering, Ritsumeikan University, Kyoto, Japan, where he is currently an Assistant Professor. His main research interest include design and control of permanent magnet synchronous motors using machine learning and deep learning. Dr. Shimizu is a member of the Institute of Electrical Engineers of Japan, Japan Institute of Power Electronics, and Society of Automotive Engineers of Japan.



Shigeo Morimoto (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Osaka Prefecture University, Sakai, Japan, in 1982, 1984, and 1990, respectively. In 1984, he joined Mitsubishi Electric Corporation, Tokyo, Japan. In 1988, he joined the Graduate School of Engineering, Osaka Prefecture University. Since 2022, he has been with the Graduate School of Engineering, Osaka Metropolitan University, Sakai, where he is currently a Professor. His main research interests include permanent magnet synchronous machines, reluctance machines and their control systems. Dr. Morimoto is a member of the Institute of Electrical Engineers of Japan, Society of Instrument and Control Engineers of Japan, Institute of Systems, Control and Information Engineers, Japan Institute of Power Electronics, and Society of Automotive Engineers of Japan.



Yukinori Inoue (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Osaka Prefecture University, Sakai, Japan, in 2005, 2007, and 2010, respectively. In 2010, he joined the Graduate School of Engineering, Osaka Prefecture University. Since 2022, he has been with the Graduate School of Engineering, Osaka Metropolitan University, Sakai, where he is currently an Associate Professor. His research interests include control of electrical drives, in particular, the direct torque control of permanent magnet synchronous motors and position sensorless control of these motors. Dr. Inoue is a member of the Institute of Electrical Engineers of Japan and the Japan Institute of Power Electronics.



Masayuki Sanada (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Osaka Prefecture University, Sakai, Japan, in 1989, 1991, and 1994, respectively. In 1994, he joined the Graduate School of Engineering, Osaka Prefecture University. Since 2022, he has been with the Graduate School of Engineering, Osaka Metropolitan University, Sakai, where he is currently an Associate Professor. His main research interest include permanent-magnet motors for direct-drive applications, their control systems, and magnetic field analysis. Dr. Sanada is a member of the Institute of Electrical Engineers of Japan, Japan Institute of Power Electronics, and Japan Society of Applied Electromagnetics and Mechanics.