# Electric Machine Two-dimensional Flux Map Prediction with Ensemble Learning

Talukder, Khaled; Wang, Bingnan; Sakamoto, Yusuke

TR2022-152 December 03, 2022

**Abstract**

Numerous finite-element simulations are required to evaluate the performance of an electric machine at different operating points, posing a great challenge to the design optimiza- tion of such electric machines. Surrogate modeling approaches have been investigated in recent years to speed up the analysis and optimization process, including machine learning and deep learning models. In particular, various convolutional neural network based deep learning models have been proposed and trained to predict motor performances for a given motor design. However, larger dataset and relatively long training time are required for such deep models. In this paper, we present a method for the rapid prediction of 2d flux maps using ensemble learning technique, with multiple relatively simple regression models. We show that the technique is much faster to train compared with deep convolutional networks, while achieving improved accuracy.

*International Conference on Electrical Machines and Systems (ICEMS) 2022*

# Electric Machine Two-dimensional Flux Map Prediction with Ensemble Learning

AKM Khaled Ahsan Talukder[1,2], Bingnan Wang[1*] and Yusuke Sakamoto[3]
[1]*Mitsubishi Electric Research Laboratories*, 201 Broadway, Cambridge, MA 02139, USA
[2]Department of Computer Science and Engineering, *Michigan State University*, East Lansing, MI, USA
[3]Advanced Technology R&D Center, *Mitsubishi Electric Corporation*, Amagasaki, Hyogo, 661-8661, Japan

***Abstract*—Numerous finite-element simulations are required to evaluate the performance of an electric machine at different operating points, posing a great challenge to the design optimization of such electric machines. Surrogate modeling approaches have been investigated in recent years to speed up the analysis and optimization process, including machine learning and deep learning models. In particular, various convolutional neural network based deep learning models have been proposed and trained to predict motor performances for a given motor design. However, larger dataset and relatively long training time are required for such deep models. In this paper, we present a method for the rapid prediction of 2d flux maps using ensemble learning technique, with multiple relatively simple regression models. We show that the technique is much faster to train compared with deep convolutional networks, while achieving improved accuracy.**

***Index Terms*—Electric machines, design optimization, machine learning.**

## I. Introduction

Electric motors are widely used in many sectors of the modern society, and becoming more important with the increasing demands from industries such as electric vehicles, factory automation, clean energy generation, etc. These new applications also have higher requirements to the performance of electric motors, such as higher power density and efficiency, and lower cost. While there are several ways for motor designer to tackle these challenges, one commonly used method is optimization during the design phase of the machines, including parameter based optimization [1] and more recently topology optimization [2]–[5] to best utilize the magnetic materials and improve motor performances.

The accurate analysis of electric machines often replies on finite-element method (FEM) based numerical simulations. Such simulations can be very time-consuming, especially when many different operating points are to be assessed for a design candidate. This poses a great challenge to the design optimization process of electric machines, when many design candidates need to be evaluated. It is therefore desirable to develop alternative method for the rapid prediction of motor performances.

Surrogate model based optimization has been investigated to speed up the process [6]. However, traditional surrogate modeling techniques do not work well for highly nonlinear problems, such as the current-dependent flux map prediction. Recently, data-driven approaches have been proposed to address the problem, by developing machine learning models for the estimation of output performances of an electric machine [7], [8]. In particular, deep convolutional neural network (CNN) based model architectures are commonly utilized, and are trained with prepared dataset to make predictions of multiple parameters with motor design candidates fed into the network as 2D images.

CNN model is capable of capturing the spatial and temporal dependencies, or features in an image, through the application of the often many filters in multiple convolution layers, and achieving superior fitting performance for the image based dataset over conventional methods. Many deep learning models have been proposed based on CNN for image classification tasks trained with ImageNet [9]. Very deep convolutional networks, such as VGGNet [10] and ResNet [11], have achieved superior classification accuracy. CNN based models have also been applied to facilitate electric machine optimization problems [7], [8], [12]–[14]. In recent study [14], CNN models have been developed to learn such current-dependent torque characteristics of motors. However, such deep networks have a huge number of parameters, often in the order of tens of millions, which require a large amount of data and resources to train. In addition, they are susceptible to noisy data and tend to overfit with training data, and can have difficulty in generalizing over unseen data.

In this work, we present an ensemble learning based technique, by utilizing multiple simple regression models, each trained to predict only a single output parameter. We show that this method is much faster to train compared with deep CNN based models, and better overall prediction accuracy can also be achieved.

## II. Proposed Ensemble Modeling Process

In this section, we first introduce the motor design optimization problem we are dealing with, and the dataset used for machine learning studies. Then we describe the proposed ensemble learning process with multiple regression models.

### A. Problem & Dataset

In this study, we deal with the design optimization of an interior permanent magnet synchronous motor (IPMSM), with the nonlinear behavior at different operating points evaluated. Major parameters, such as the size and architecture of the motor are fixed, with 48 slots in the stator, 8 poles in the rotor composed of V-shaped permanent magnets. The whole rotor core is subject to shape optimization. During

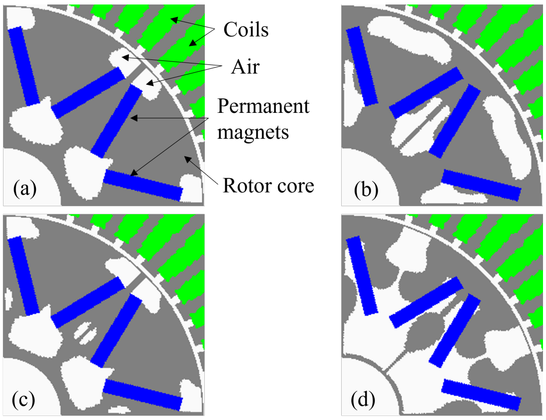


Figure 1. The cross-section image of an interior permanent magnet motor for the study.

the optimization process using Normalized Gaussian network (NGnet) method [2], a total of 27,949 design candidates are evaluated using FEM simulations. A quarter of the cross-section of these generated design candidates during the optimization process are stored in the form of RGB images, with size 224 $\times$ 224 $\times$ 3, and serve as input data of the dataset for machine learning purposes. Fig. 1 shows a few example images from the dataset. For each design, FEM simulations are conducted at each operating points($I_d, I_q$) to identify corresponding flux linkage ($\phi_d$, $\phi_q$). The 2D flux maps can be described using the below quadratic functions with least-square fitting:

$$\phi_q = \beta_{q0} i_d^2 + \beta_{q1} i_q^2 + \beta_{q2} i_d i_q + \beta_{q3} i_d + \beta_{q4} i_q + \beta_{q5}, \quad (1)$$

$$\phi_d = \beta_{d0} i_d^2 + \beta_{d1} i_q^2 + \beta_{d2} i_d i_q + \beta_{d3} i_d + \beta_{d4} i_q + \beta_{d5}. \quad (2)$$

The 12 coefficients, $\beta_{q0}$ through $\beta_{d5}$, are stored in the form a 12 $\times$ 1 vector and serve as the output data of the dataset. Our goal is to develop machine learning models to make prediction of the 12 flux map coefficients for a new design candidate, such that the normalized root-mean-square error (RMSE) between the prediction and the ground truth values is as small as possible.

### *B. Ensemble Learning Technique*

In this work, we need to predict the output of 12 current-dependent flux map model coefficients using a machine learning model for a given motor design, which is represented by a 2d image, which is essentially a high-dimensional input matrix. This problem is intrinsically a multi-output regression problem. Many machine learning algorithms have been developed and sophisticated for single-output regression problems, which deal with the prediction of a single numeric value. While some algorithms can in principle handle multi-output regression problems, such as linear regression model and decision tree model, they often do not perform well on highly nonlinear problems with high-dimensional input data. Deep learning models with more involved architectures are more commonly used to address such problems. In particular, for image recognition and classification problems using 2d image data as input, convolutional neural network (CNN) based models are among the most popular techniques. Previously, CNN based models have been applied to speed up the electric machine design optimization [7], [8], [12]–[14]. In a more recent study [14], CNN models have been developed to learn the highly nonlinear current-dependent torque output of motor design candidates. One main drawback of the CNN based deep learning models is that the huge number of model parameters often requires a large dataset and dedicated GPU for training. It is also challenging to generalize the trained model to new data, due to their tendency to over-fit with the training data, especially when the dataset is small.

Instead of tuning the hyper-parameters of deep learning models, in this work, we utilize ensemble learning technique to address the problem. In short, we use multiple relatively simple regression models, each trained to predict only one output parameter; these models are then ensembled to make prediction for all the required output parameters. Since each regression model is much simpler and easier to train, we can afford to have multiple such models working together and still be faster than deep learning models. In addition, the prediction accuracy can be improved, since regression models can work well for single-output problems. The process of the proposed method is illustrated in Fig. 2. With a total of 12 regression models trained, we can make a full prediction to both $\phi_d$ and $\phi_q$ flux maps.

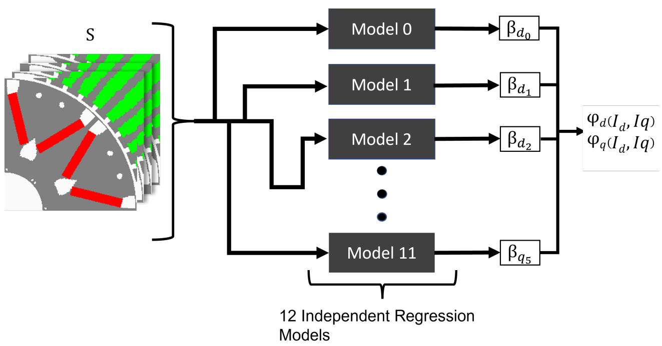


Figure 2. Ensemble learning process to predict flux map model parameters from input motor design images.

### *C. Regression Models*

Two types of regression models are implemented and tested in this paper: support vector regression (SVR) model, and multi-layer perceptron (MLP) model.

Support vector regression (SVR) models are built on support vector machines (SVMs), which are more well-known for classification problems. SVM is an established machine learning model built on statistical learning framework, and offers robust prediction to data samples. The classical SVM algorithm is a non-probabilistic binary linear classifier. It takes data samples from two categories and maps these samples into space, and finds a line, or hyperplane in higher-dimensional space, that distinctly classifies the data points, so that the width of the gap between the two categories is maximized. When a new data sample is fed into the SVM, it is mapped into a point in the data space in the same manner, and its category is easily predicted by which side of the gap the point falls into. SVMs can not only perform linear classification problems, but also work effectively for nonlinear problems using kernel method, which implicitly maps low-dimensional input data into high-dimensional space with the help of kernel functions. Commonly used nonlinear kernel functions for SVMs include polynomial functions, radial basis functions (RBFs), sigmoid functions, etc.

SVR is a supervised learning algorithm that uses the same principle of SVM for regression problems. For a given threshold value $\epsilon$ , the algorithms finds the best fit line, or hyperplane, for given training data samples, so that it has the maximum number of points with a distance smaller than the threshold value $\epsilon$. Unlike other regression models which typically minimize the mean-squared error between prediction and ground truth, SVR tries to find the best fit within an error threshold $\epsilon$, making it more robust to outliers and generalizes better to new data.

Multi-layer perceptron (MLP) is a supervised learning algorithm that learns a nonlinear function between the input data and corresponding output data, by training on a dataset. MLP utilizes a feed-forward artificial neural network (ANN) model, which consists of multiple fully-connected layers. There may be one or more nonlinear hidden layers in between the input and output layers, and nonlinear activation functions such as *logistic*, *tanh*, and *ReLU* are applied to the nodes of the hidden layers to provide nonlinear mapping between input and output data. It is a classical machine learning algorithm for both classification and regression problems.

We use scikit-learn python library for the implementation of both SVR and MLP models [15].

## III. Numerical Test Results

Th performance of a regression model largely depend on its hyper-parameters. For SVR, the kernel function can be polynomial, RBF, or sigmoid; the optimal value of kernel coefficient $\gamma$ can also be different depending on the dataset and the choice of kernel function; the regularization parameter $C$ can also be adjusted to tune the strength of regularization; different $\epsilon$ values can also be specified. By default, for each SVR model, we use RBF as kernel function, with $\gamma$ inversely proportional to the variance of the training data, regularization parameter $C = 1.0$, and $\epsilon = 0.1$. We also individually tune the hyper-parameters for each model using a grid search approach.

For MLP model, we can tune the number of hidden layers, the size and the activation function of each hidden layer, the choice of weight optimization solver, the strength of the $L_2$ regularization term $\alpha$, the learning rate , etc. By default, we use one hidden layer of size 100, with $ReLU$ as activation function, $adam$ as optimization solver, and $\alpha = 0.0001$. Grid search can also be conducted to tune these parameters for each MLP model.

For comparison purpose, two deep learning models based on CNN, namely VGG16 and ResNet50 [16], are built and trained on the same dataset. They are the high-performing network architectures image classification tasks on ImageNet.

For VGG network, the basic building block includes a stack of multiple convolution layers (filter size $3 \times 3$, stride length 1, padding 1) followed by a MaxPooling layer ($2 \times 2$). Such building blocks can be repeated multiple times in the model to reach different depths as desired. For VGG16, 16 layers with weight parameters are included in the model. Original VGG16 model also includes three fully-connected layers following the convolution blocks, with the last one of size 1,000 with *Softmax* activation, representing the classes in the ImageNet. In our case, since we use it to make predictions for the 12 flux map parameters, the fully-connected layers are modified to make the last layer of size 12 without activation for our regression task.

ResNet, which stands for residual network, allows for effective training of very deep neural networks that were previously very challenging or even impossible to train due to vanishing gradient problem [16]. The innovative feature of the ResNet is the “skip connections” in convolution blocks and identity blocks, which adds the input $x$ itself to the output $\mathcal{F}(x)$ of each block including several convolution layers, so that the output becomes $\mathcal{F}(x) + x$. Skip connections set up a shortcut for gradients to pass through, which mitigates the vanishing gradient problem; they also allow the model to learn an identity function to ensure the higher layers perform at least as good as the lower layers. As a result, ResNet improves the efficiency of deeper neural networks while minimizing the error propagation.

The input RGB image of size $224 \times 224 \times 3$ is fed to the first convolution layer after zero padding, followed by batch normalization, *ReLU* activation, and MaxPooling; four consecutive stages follow, each composed of a convolution block and multiple identity blocks. After average pooling, the output is flattened and connected to a dense layer.

For all the numerical tests, models are trained on 20,124 training data, and test on 7,825 unseen data, corresponding to a 0.72/0.28 training/test split ratio. The RMS error is evaluated using the test data for each of the trained model, and the results are shown in Table I. There are two numbers shown for SVR and MLP, with the first number showing the test result for models built with the default settings, the later number showing the test result for models using parameters tuned with grid search method. Even with the default model settings, the prediction performance for the ensemble learning with regression models already outperform the deep learning models built with VGG16 and ResNet50. After the fine tuning, the RMS error is further reduced for both SVR and MLP models, making the ensemble learning method more accurate compared with the deep CNN models. The accuracy of the predicted model parameters is also reflected in the correlation plot in Fig. 3 using the ensembled MLP models as an example. For all the $\beta_q$ parameters, the $R$ value is higher than 0.97. Similar results for $\beta_d$ parameters can also be

Table I
COMPARISON OF DIFFERENT MODELS

| Model | Prediction RMS Error |
|---|---|
| VGG16 | 0.205 |
| ResNet50 | 0.172 |
| SVR | 0.162/0.154 |
| MLP | 0.165/0.147 |

obtained. It is worth to note that, in addition to the improved accuracy, the training process of ensemble models is much faster and does not require a dedicated GPU, in contrast to the deep learning models built on CNNs.

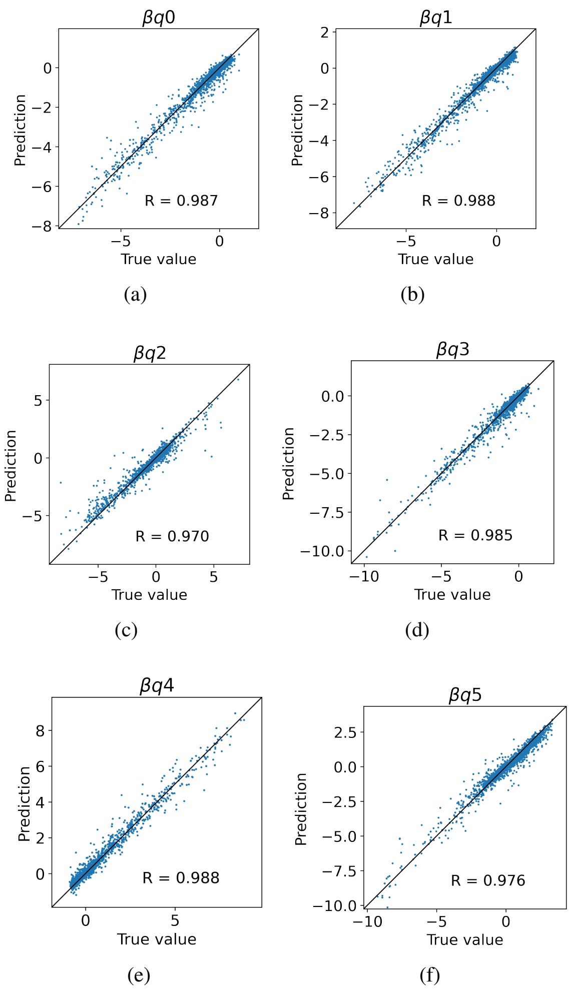


Figure 3. Correlation plot of the predicted flux map coefficients using ensembled MLP models from $\beta_{q0}$ through $\beta_{q5}$, with correlation coefficients between predicted value and ground truth listed for each plot.

## IV. CONCLUSIONS

In this paper, we investigated the surrogate modeling method for electric motor design optimization. In particular, a machine learning based surrogate model was developed to rapidly predict the current-dependent flux map, which would require a large number of finite-element simulations for each motor design. We showed that ensemble learning method with multiple relatively simple regression models can achieve very good accuracy as compared with much deeper models built on CNNs, while much lighter to train and deploy. The proposed method can be easily adopted to other motor design problems with different output metrics.

## V. ACKNOLEDGEMENT

The authors sincerely thank Prof. Hajime Igarashi and Mr. Taiga Aoyagi from Graduate School of Information Science and Technology of Hokkaido University for the support of this work and generously providing the dataset.

## REFERENCES

[1] G. Bramerdorfer, J. A. Tapia, J. J. Pyrhönen, and A. Cavagnino, “Modern electrical machine design optimization: Techniques, trends, and best practices,” *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 7672–7684, 2018.

[2] T. Sato, K. Watanabe, and H. Igarashi, “Multimaterial topology optimization of electric machines based on normalized gaussian network,” *IEEE transactions on magnetics*, vol. 51, no. 3, pp. 1–4, 2015.

[3] M. Garibaldi, C. Gerada, I. Ashcroft, and R. Hague, “Free-form design of electrical machine rotor cores for production using additive manufacturing,” *Journal of Mechanical Design*, vol. 141, no. 7, 2019.

[4] A. Credo, G. Fabri, M. Villani, and M. Popescu, “Adopting the topology optimization in the design of high-speed synchronous reluctance motors for electric vehicles,” *IEEE Transactions on Industry Applications*, vol. 56, no. 5, pp. 5429–5438, 2020.

[5] F. Guo and I. P. Brown, “Simultaneous magnetic and structural topology optimization of synchronous reluctance machine rotors,” *IEEE Transactions on Magnetics*, vol. 56, no. 10, pp. 1–12, 2020.

[6] R. C. P. Silva, T. Rahman, M. H. Mohammadi, and D. A. Lowther, “Multiple operating points based optimization: Application to fractional slot concentrated winding electric motors,” *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1719–1727, 2017.

[7] S. Doi, H. Sasaki, and H. Igarashi, “Multi-objective topology optimization of rotating machines using deep learning,” *IEEE transactions on magnetics*, vol. 55, no. 6, pp. 1–5, 2019.

[8] A. Khan, M. H. Mohammadi, V. Ghorbanian, and D. Lowther, “Efficiency map prediction of motor drives using deep learning,” *IEEE Transactions on Magnetics*, vol. 56, no. 3, pp. 1–4, 2020.

[9] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, “Imagenet large scale visual recognition challenge,” *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[10] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[12] H. Sasaki and H. Igarashi, “Topology optimization accelerated by deep learning,” *IEEE Transactions on Magnetics*, vol. 55, no. 6, pp. 1–5, 2019.

[13] A. Khan, V. Ghorbanian, and D. Lowther, “Deep learning for magnetic field estimation,” *IEEE Transactions on Magnetics*, vol. 55, no. 6, pp. 1–4, 2019.

[14] T. Aoyagi, Y. Otomo, H. Igarashi, H. Sasaki, Y. Hidaka, and H. Arita, “Prediction of current-dependent motor torque characteristics using deep learning for topology optimization,” in *2021 23rd International Conference on the Computation of Electromagnetic Fields (COMPUMAG)*, 2022.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[16] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.