Progress In Electromagnetics Research B, Vol. 100, 173–189, 2023

Deep Learning Based Modelling of Three-Dimensional Magnetic Field

Van T. Nguyen[1, *], Steffen Bollmann[2], Michael Bermingham[1], Ha X. Nguyen[3], and Matthew S. Dargusch[1]

Abstract—Computation of the magnetic field generated by permanent magnets is essential in the design and optimization of a wide range of applications. However, the existing methods to calculate the magnetic field can be time-consuming or ungeneralised. In this research, a deep learning-based fast-computed and generalised model of three-dimensional (3D) magnetic field is studied. The volumetric deep neural network model (V-Net) which consists of a contracting part to learn the geometrical context and an expanding part to enable the concise localization was applied. We synthetically generated the ground truth datasets from permanent magnets of different 3D shapes to train the V-Net. The accuracy and efficiency of this deep learning model are validated. Predicting on 50 random samples, the V-Net took 4.6 s with a GPU T4 and 23.2 s with the CPU whereas the others took a few hundreds to thousands of seconds. Therefore, the deep learning model can be potentially utilised to replace the other methods in the computation and study of the magnetic field for the design and optimization of magnetic devices (the codes used in this research are published openly in https://github.com/vantainguyen/3D_V-Net_MagneticField).

1. INTRODUCTION

Permanent magnets have been effectively implemented in many applications including in non-invasive treatments and diagnoses [1, 2], magnetic gears and couplings [3–5], contactless sensing, and robotics [6, 7]. In order to facilitate the design and optimization processes [8] of the magnetic devices, the computation of the magnetic field distribution generated by utilized permanent magnets is essential.

The conventional method to compute the magnetic field from an object is Finite Element Method (FEM) [9]; however, this method can be time-consuming and may not be suitable for an optimization process of magnetic devices with large variables. Therefore, there have been many attempts to investigate different techniques to tackle this issue. For magnetic field of basic shapes such as cylinders [10], polyhedron [11], elliptical cylinders [12], and cones [13, 14], analytical and semi-analytical models can be derived which are faster in terms of computational expenses than FEM. Recently, machine learning models based on multilayer perceptron (MLP) have been demonstrated to be more computationally efficient than the semi-analytical models and FEM [15]. One of the drawbacks of the current analytical, semi-analytical and MLP-based models is that they are limited to some special geometries of permanent magnets; in other words, they are lack of generalization. Moreover, the semi-analytical and MLP-based models are faster than the FEM; however, they can be time-consuming for the 3D field simulation.

Received 14 May 2023, Accepted 14 June 2023, Scheduled 1 July 2023

[*] Corresponding author: Van Tai Nguyen (vantai.nguyen@uqconnect.edu.au).

[1] School of Mechanical and Mining Engineering, The University of Queensland, Brisbane St Lucia, QLD 4072, Australia. [2] School of Information Technology and Electrical Engineering, The University of Queensland, Brisbane St Lucia, QLD 4072, Australia.

[3] School of Mechanical Engineering, Hanoi University of Science and Technology, No. 1, Dai Co Viet, Hanoi, Vietnam.

With the ability of modelling two-dimensional (2D) magnetic field of random shapes in a single model, deep learning has been applied [16]. However, the 2D magnetic field modelling can be useful when the shapes of permanent magnets are symmetrical. Moreover, the magnetic field distribution in real-world is three dimensional (3D), thus, a fast-computed and accurate modelling of the 3D field can be useful in the design and optimization processes. Therefore, in this article, we present a deep learning-based model of 3D magnetic field which can construct the field from arbitrarily magnetized geometrical shapes in a single model. We model the 3D magnetic field based on the V-Net [17] deep learning architecture. The computational efficiency and generalization of this deep learning model are demonstrated, and it can be a candidate to replace the need of FEM and the others for the computation of the magnetic field.

2. METHOD

The volumetric deep neural network model (V-Net) is applied to learn the magnetic field distribution generated from permanent magnets. The details of this model and data generation method to train the model are discussed as follows.

2.1. Deep Learning Model

Inspired by the volumetric deep neural network [17] which was initially developed to solve the volumetric medical image segmentation, our deep learning model (V-Net) consists of three major components: the contracting part (encoder), expanding part (decoder), and skip connections (Fig. 1). The contracting part contains six down-sampling convolutional layers; each convolutional layer has a filter kernel $3 \times 3 \times 3$, same padding [18], and stride $2 \times 2 \times 2$ which enables the reduction of the input sizes by half; leaky rectified linear unit (ReLU) with a slope coefficient 0.3 is utilized as the activation function to add the non-linearity to the computation, and batch normalization is applied after each convolution layer. The application of the activation function and batch normalization can help prevent the gradient vanishing and exploding issues [18]. The number of filters (feature maps) of the preceding convolutional layer doubles the number of filters of the previous one; the number of filters of the first convolutional layer is called the filter base. The expanding part contains five up-sampling (transpose) convolutional layers. Each convolutional layer has a $3 \times 3 \times 3$ filter kernel, same padding, and $2 \times 2 \times 2$ striding which allows

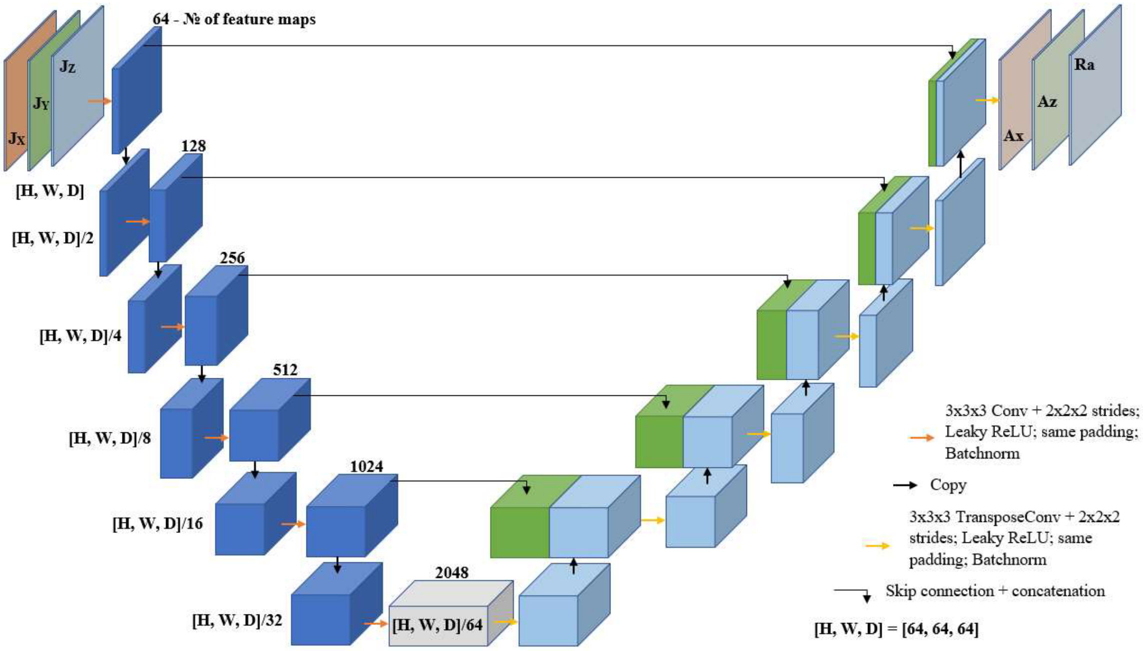

64 - № of feature maps
Jx
JY
JZ
128
[H, W, D]
[H, W, D]/2
256
[H, W, D]/4
512
[H, W, D]/8
1024
[H, W, D]/16
2048
[H, W, D]/32
[H, W, D]/64
Ax
Az
Ra
3x3x3 Conv + 2x2x2 strides; Leaky ReLU; same padding; Batchnorm
Copy
3x3x3 TransposeConv + 2x2x2 strides; Leaky ReLU; same padding; Batchnorm
Skip connection + concatenation
[H, W, D] = [64, 64, 64]

Figure 1. 3D V-Net architecture for 3D magnetic field modelling.

increasing input sizes. The same as the contracting part, batch normalization and leaky ReLu activation are applied after each convolutional operation. The skip connections with concatenation enable the V-Net to apply the fine-grained details in the encoder part to construct the output of the decoder part, which helps compensate the information loss due to the down-sampling of the inputs in the contracting part. The number of filters in the expanding part are equal to that in the symmetrical contracting part.

The inputs of the V-Net are 3D $64 \times 64 \times 64 \times 3$ tensors with three channels; each channel contains the information about the magnitude of projection of the vector magnetization $\vec{J}$ on the coordinate system $OXYZ$ (Fig. 2). In other words, the red, green, and blue channels are correspondent to the magnetization projections on the OX (J_X), OY (J_Y), and OZ (J_Z) coordinates, respectively, with the value in each channel being the projection magnitude. Moreover, the outputs are 3D $64 \times 64 \times 64 \times 3$ tensors with three channels; the red, green, and blue channels are the axial (Ax), azimuthal (Az), and radial (Ra) components of the magnetic field, respectively.

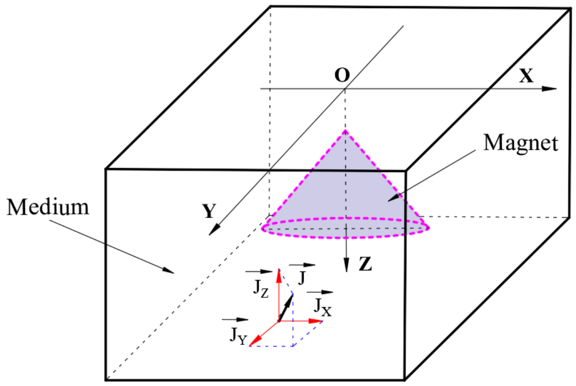


Figure 2. Permanent magnet and medium setting.

2.2. Data generation and Preprocessing

The magnetic field generated from a permanent magnet can be computed by solving the Maxwell's equations (Eqs. (1) and (2)) taking the material properties into account (Eq. (3)) in magnetostatics with FEM [9].

$$\vec{\nabla} \times \vec{H} = 0 \tag{1}$$

$$\vec{\nabla} \cdot \vec{B} = 0 \tag{2}$$

$$\vec{B} = \mu_0 \vec{H} + \vec{J} \tag{3}$$

where H, B, μ_0, and J are the magnetic field intensity, magnetic flux density, permeability of the vacuum medium and magnetization, respectively.

However, the FEM is time-consuming to compute a large number of samples, thus we utilised faster-computed semi-analytical expressions derived from Eqs. (1), (2), and (3) and validated against the FEM (Appendix A) to simulate the ground truth (label) magnetic field.

The equations of the magnetic field are coded and executed using the object-oriented programming language Python. The missing data in the simulated magnetic field components are limited to less than one percent. Moreover, these missing data are filled using the K-nearest neighbours technique [19] with four neighbours.

In order to cope with the variations in the magnetic field components, they are normalized by the standard scaler method (H^{norm}, Eq. (4)). Moreover, the data normalization can lead to the improvement of the model's numerical stability and training speed [20].

$$H^{norm} = \frac{H - \mu}{\sigma}, \tag{4}$$

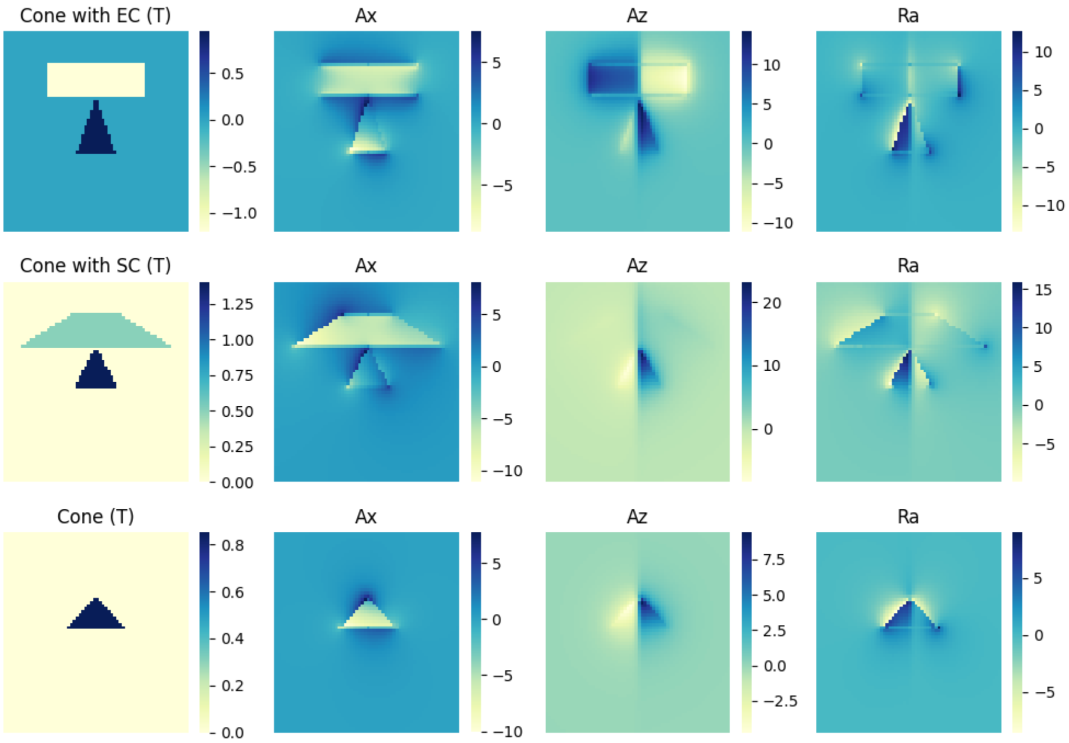


Figure 3. Examples of generated data; SC: Sectional cone, EC: Elliptical cylinder, T: Tesla, Ax, Az, and Ra are the axial, azimuthal and radial components of the magnetic field.

where μ and σ are the mean and standard deviation of each field component (H).

In this study, we generated 13500 random label samples of 3D permanent magnets with different shapes and configurations (geometrical parameters are listed in Tables B1–B3, Appendix B) including cones [13, 14]; more complex configurations are created by adding a random elliptical cylinder [12] to the cones and adding a random sectional cone to the cones. For a visualization purpose, some examples (2D sections) of the generated data after applying the normalization are depicted in Fig. 3. For testing the robustness of the V-Net model, we generated 50 random samples from the out of training intervals (Table B4, Appendix B) and 50 random samples from an unseen pattern (circular cylinder) in the training data (Table B5, Appendix B). In total, 160 GB of data were generated.

The mean squared error function (Eq. (5)) is utilised as the loss function which is to be minimised during the training process.

$$\mathcal{L} = \frac{1}{N}\sum_{1}^{N}(H_i - \widehat{H_\iota})^2, \tag{5}$$

where H and $\widehat{H}$ are the label and predicted magnetic fields, and N is the total number of the computed points in space.

In order to evaluate the performance of the deep learning model, three metrics are implemented.

The first one is the cosine similarity (Eq. (6)) whose range is between -1 (the worst fit) and 1 (the best fit). It measures the closeness of two arrays (vectors) of the predicted and label values correspondingly.

$$Cosimi(H, \widehat{H}) = \frac{H \cdot \widehat{H}}{||H||\,||\widehat{H}||}, \tag{6}$$

The second measure is the R squared (R^2) (Eq. (7)) which is the proportion of the variation in the dependent variable which can be explained by the independent variables. R^2 has a range between 0 and

1 (the best fit), and it can be less than 0 for the worse performance of the deep learning model.

$$R^2(H, \widehat{H}) = 1 - \frac{\sum_1^N (H_i - \widehat{H_\imath})^2}{\sum_1^N (H_i - E[H])^2}, \tag{7}$$

The third metric is the Pearson correlation coefficient (ρ) (Eq. (8)) which measures the linear relationship between two datasets. This metric has a range of -1 (oppositely correlated) and 1 (perfectly correlated).

$$\rho(H, \widehat{H}) = \frac{cov(H, \widehat{H})}{\sigma_H \sigma_{\widehat{H}}}, \tag{8}$$

where σ_H and $\sigma_{\widehat{H}}$ are the standard deviations of the ground truth and predicted field, and $cov(H, \widehat{H})$ is the covariance between them.

3. EXPERIMENTS

The ground truth data are split into training, validation, and testing datasets with a ratio of 80/10/10. In order to facilitate the loading data during training V-Net, the data were converted into TFRecord format [21] which stores a sequence of binary records (three shards for training data, one shard for validation, and one shard for testing datasets). The initial learning rate was set to 0.001 as the default value from Tensorflow Adam optimizer [22], and this learning rate was decreased by a factor of 1/10 after 10 epochs of no improvement in the validation loss; the minimum learning rate was 1e-5. The training batch size was set to 16 which was suitable for our computing facility. Early stopping was implemented to terminate the training process when no improvement in the validation loss was observed for consecutive 15 epochs. Inspired by the U-Net architecture [23] the number of filters was chosen. Moreover, we trained the V-Net with two different initial numbers of filters 32 and 64 (other hyperparameters remain the same). The V-Net with 64 filters stopped at the 47th epoch whereas the V-Net with 32 filters took more than 80 epochs to complete (Fig. 4). Regardless of the training epoch, the 64 filters V-Net took 5.76 hours, and the 32 V-Net consumed 3.73 hours. However, we achieved the convergence of the V-Net with 64 filters with lower validation loss (0.0018) than the one with 32 filters (0.0032). Therefore, for further analysis of the performance of the V-Net model, we chose 64 filters V-Net.

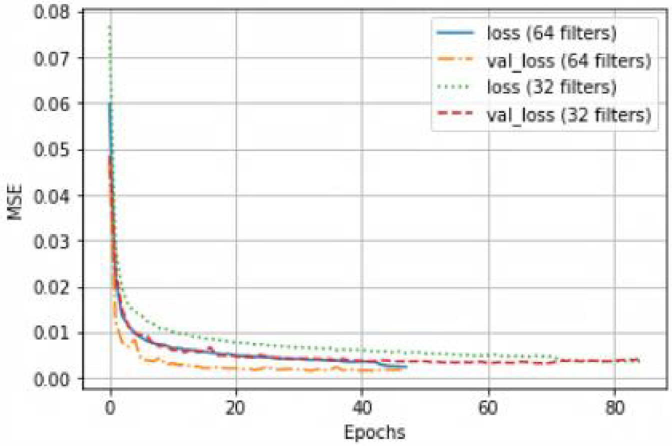
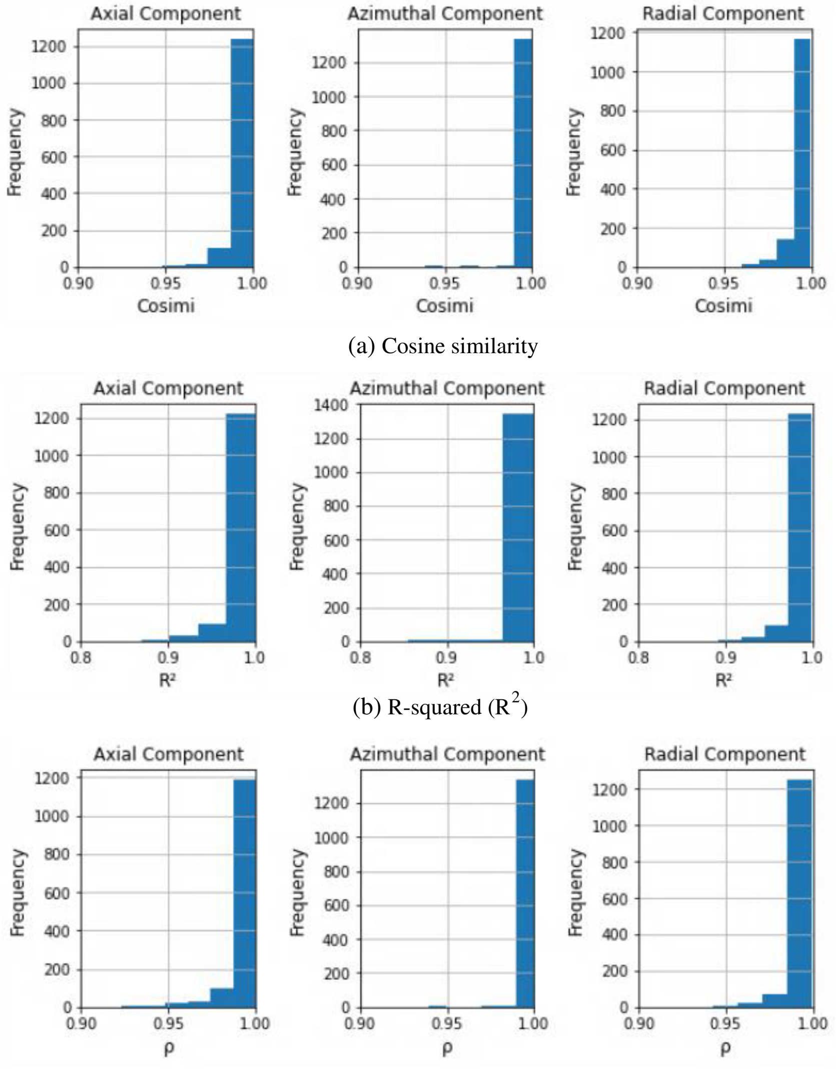


Figure 4. Loss and validation loss (val_loss) of the training process with 32 and 64 filters; MSE: Mean squared errors.

4. RESULTS

The trained deep learning model V-Net with 64 filters base was applied to compute the magnetic field generated from 1350 random samples of the unseen testing dataset. The histograms of the cosine similarity, R-squared and Pearson correlation coefficient of the predictions and labels are depicted in Fig. 5. These figures show that the majority of the metric values are in the interval of 0.9 and 1. Moreover, the percentages of the returned metrics greater than 0.95 for the (axial, azimuthal and radial components) are (99.85, 99.63, 99.78) (Cosimi in %), (94.44, 99.26, 96.74) (R^2 in %), and (98.96, 99.63, 99.70) (ρ in %), respectively. These results demonstrate that the deep learning model can accurately predict the magnetic field distribution from unseen data samples.



(a) Cosine similarity

(b) R-squared (R^2)

(c) Pearson correlation coefficient (ρ)

Figure 5. Comparison metrics between the labels and predictions of the testing dataset; (a) Cosine similarity (Cosimi), (b) R-squared (R^2), (c) Pearson correlation coefficient (ρ).

For the purpose of visual inspection (comparison), Figs. 6, 7, and 8 display the ground truth and prediction of the magnetic field distribution of a random sample (this sample will be available on the public Github repository) drawn from the unseen testing dataset in the Axial, Sagittal, and Coronal planes. It can be seen that it is hardly distinguishable by human naked eyes between the ground truth and the predicted results using the V-Net; moreover, the mean (μ) and standard deviation (σ) of the

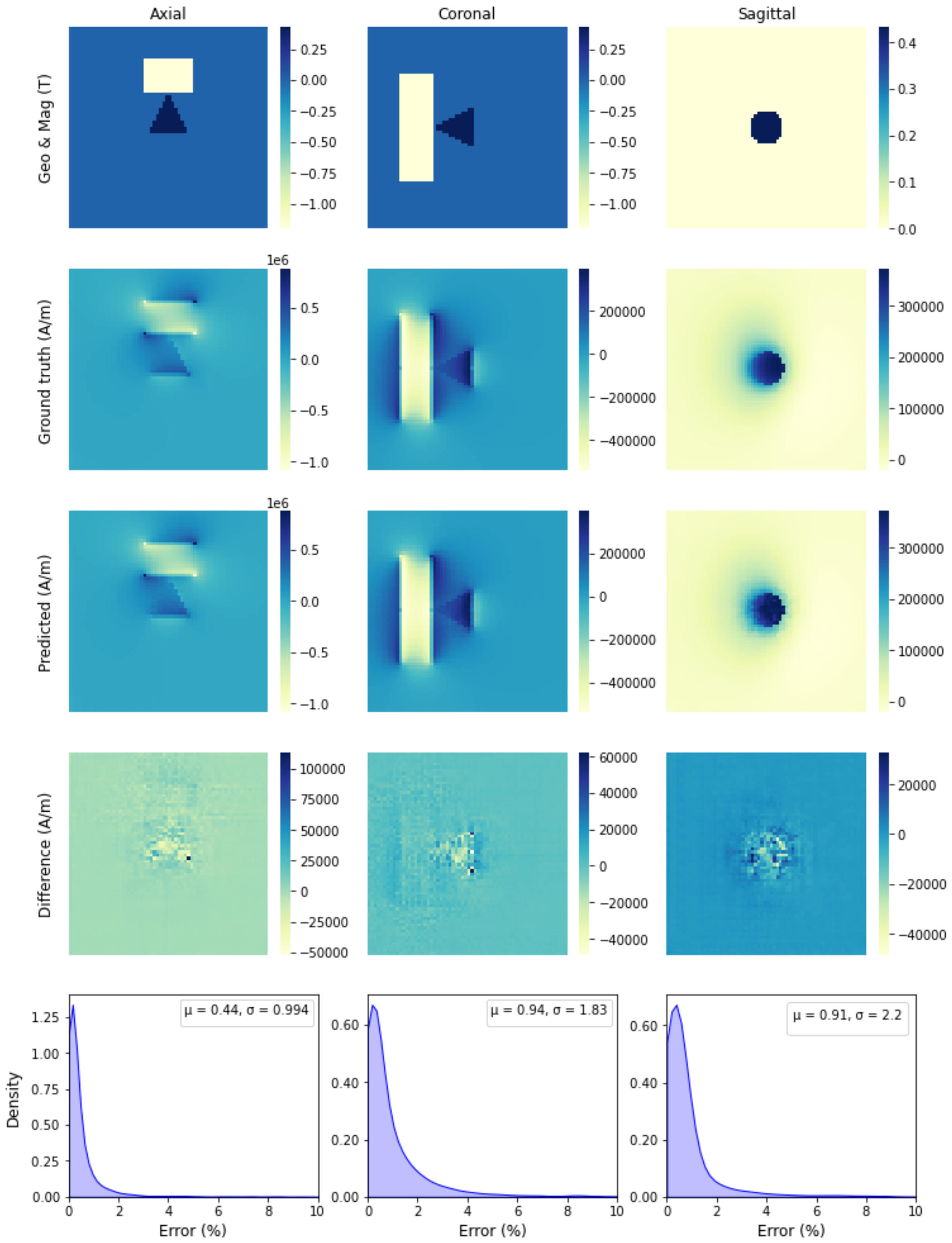


Figure 6. Visual comparison of the axial component; Geo & Mag denotes the geometry and magnetization of the magnet.

normalised errors (differences between the two results) computed using Eq. (9) are comparatively small ($\mu \leq 2.4\%$, $\sigma \leq 7.08\%$).

$$Error = \frac{|H - \widehat{H}| \cdot 100\%}{IQR_H(5, 95)}, \tag{9}$$

where $IQR_H(5, 95)$ is the interquartile range between the 5th and 95th percentiles of the corresponding ground truth field.

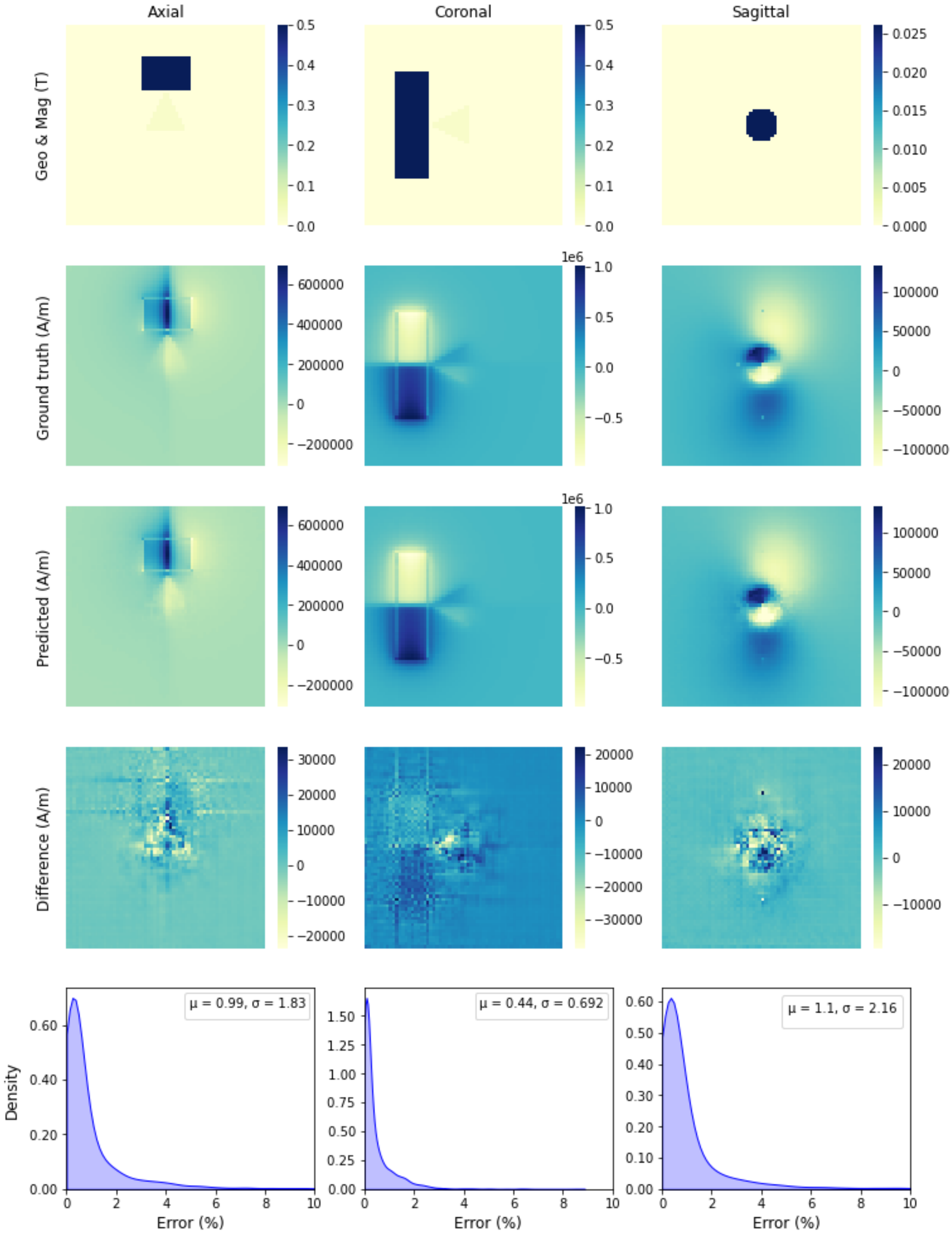


Figure 7. Visual comparison of the azimuthal component; Geo & Mag denotes the geometry and magnetization of the magnet.

The trained V-Net is also utilized to predict the magnetic field distribution from 50 random cone samples generated from geometrical parameters outside the training intervals (Table B4, Appendix B). We observed that the accuracy of the prediction slightly decreases (Fig. 9) compared to the accuracy of the testing dataset (in the training intervals); the percentages of the returned metrics greater than 0.95 for the (axial, azimuthal and radial components) are (92.0, 100.0, 28.0) (Cosimi in %), (8.0, 94.0, 0.0) (R^2 in %), and (90.0, 100.0, 28.0) (ρ in %), respectively. The accuracy of the predicted results (Fig. 10) compared to the ground truth decreases further for the unseen pattern from the training

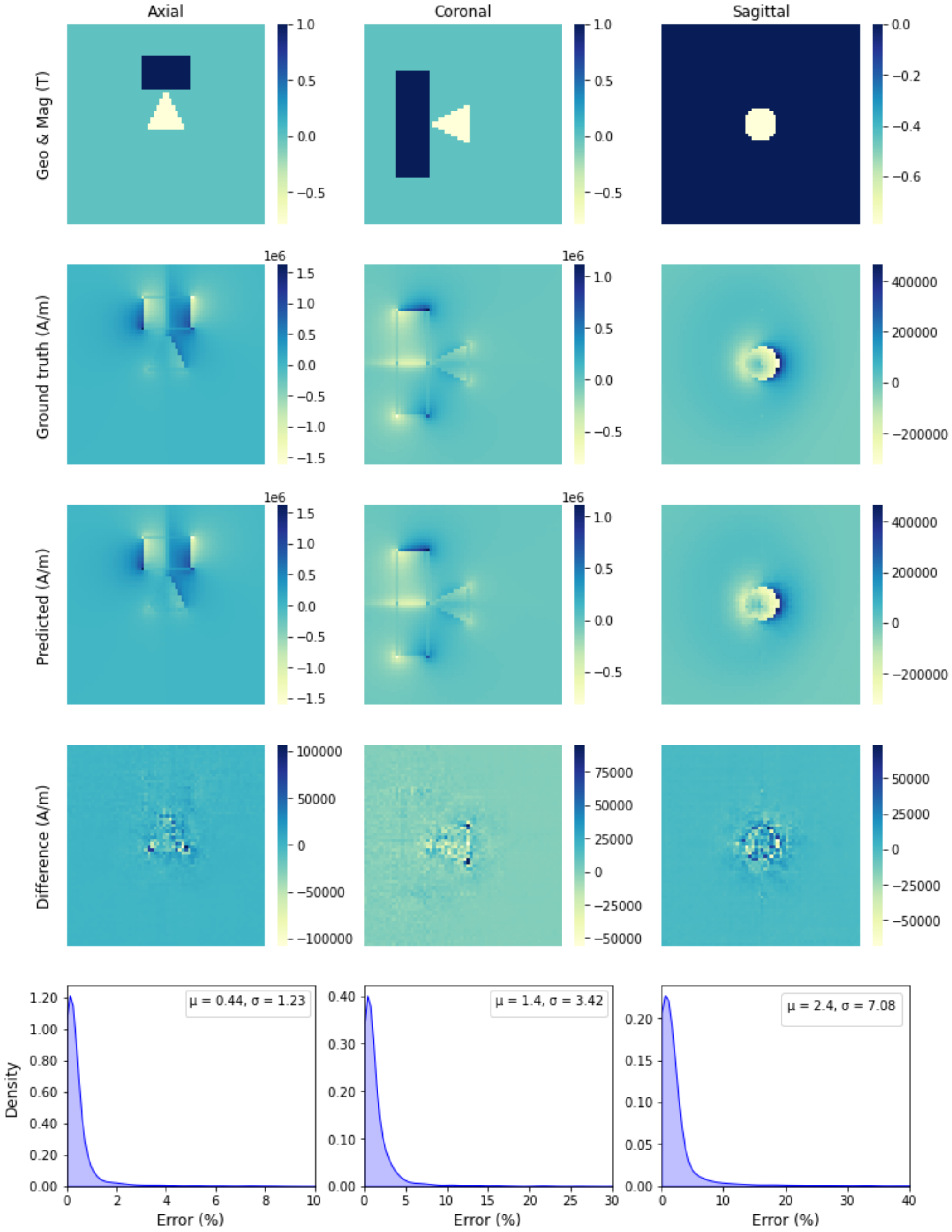
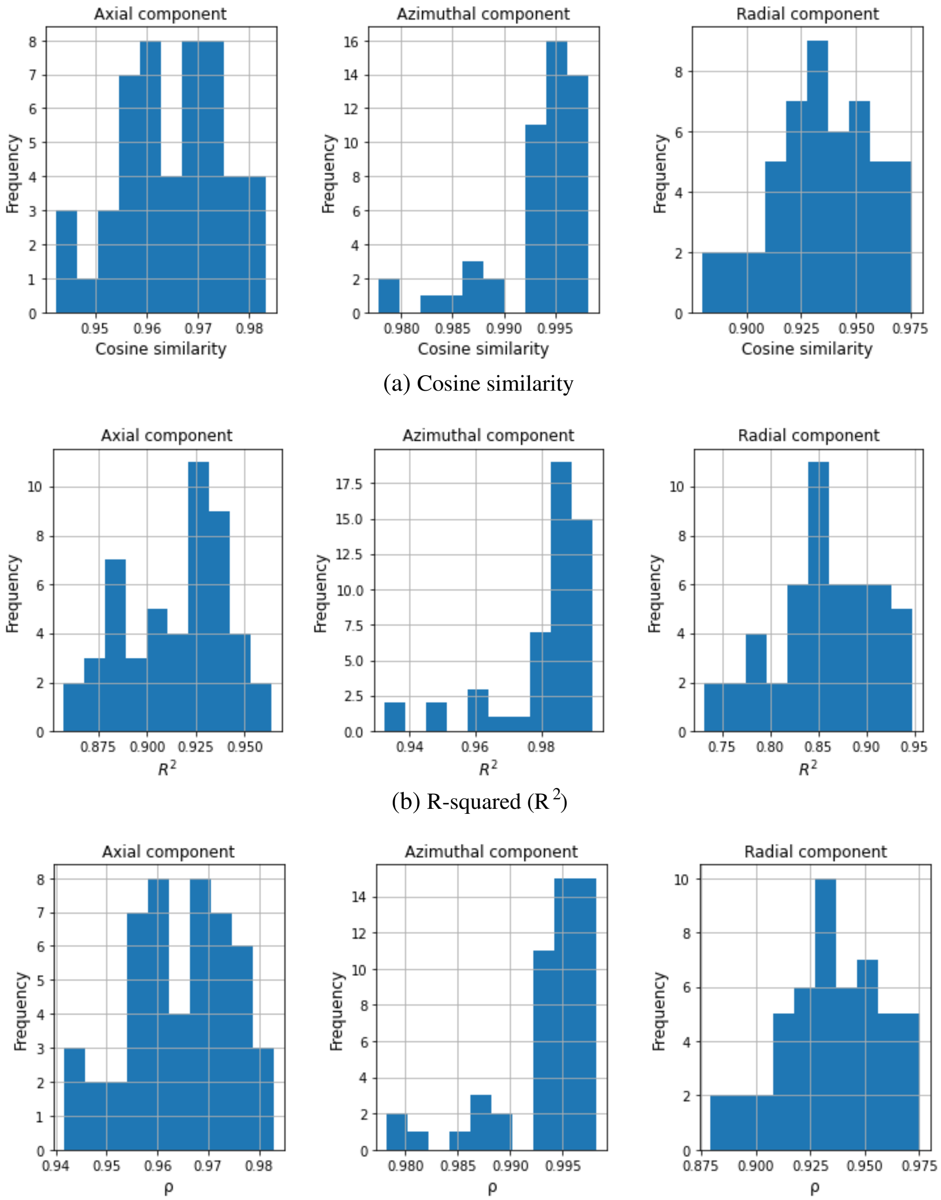


Figure 8. Visual comparison of the radial component; Geo & Mag denotes the geometry and magnetization of the magnet.

dataset (we have included the cone, complex geometries formed by the cone and the sectional cone; the cone and the elliptical cylinder in the training dataset, but we tested the performance of the V-Net on the permanent magnets of a circular cylindrical shape). For these unseen pattern data, the percentages of the returned metrics greater than 0.95 for the (axial, azimuthal and radial components) are (6.0, 62.0, 0.0) (Cosimi in %), (0.0, 24.0, 0.0) (R^2 in %), and (4.0, 62.0, 0.0) (ρ in %), respectively. These are a common phenomenon observed with the supervised machine learning [15]. Moreover, it can be addressed by retraining V-Net with the data generated from the desired intervals (the transfer learning techniques [24] can be implemented to accelerate the training process).

Table 1. Computational time comparison.

Methods	**Execution time (s)**
V-Net	4.6 with a GPU T4 and 23.2 with the CPU
FEM [9, 14, 25]	> 39000
SAE [14, 15]	> 39000
MLP [15]	743.8 with a GPU T4 and 12019.3 with the CPU

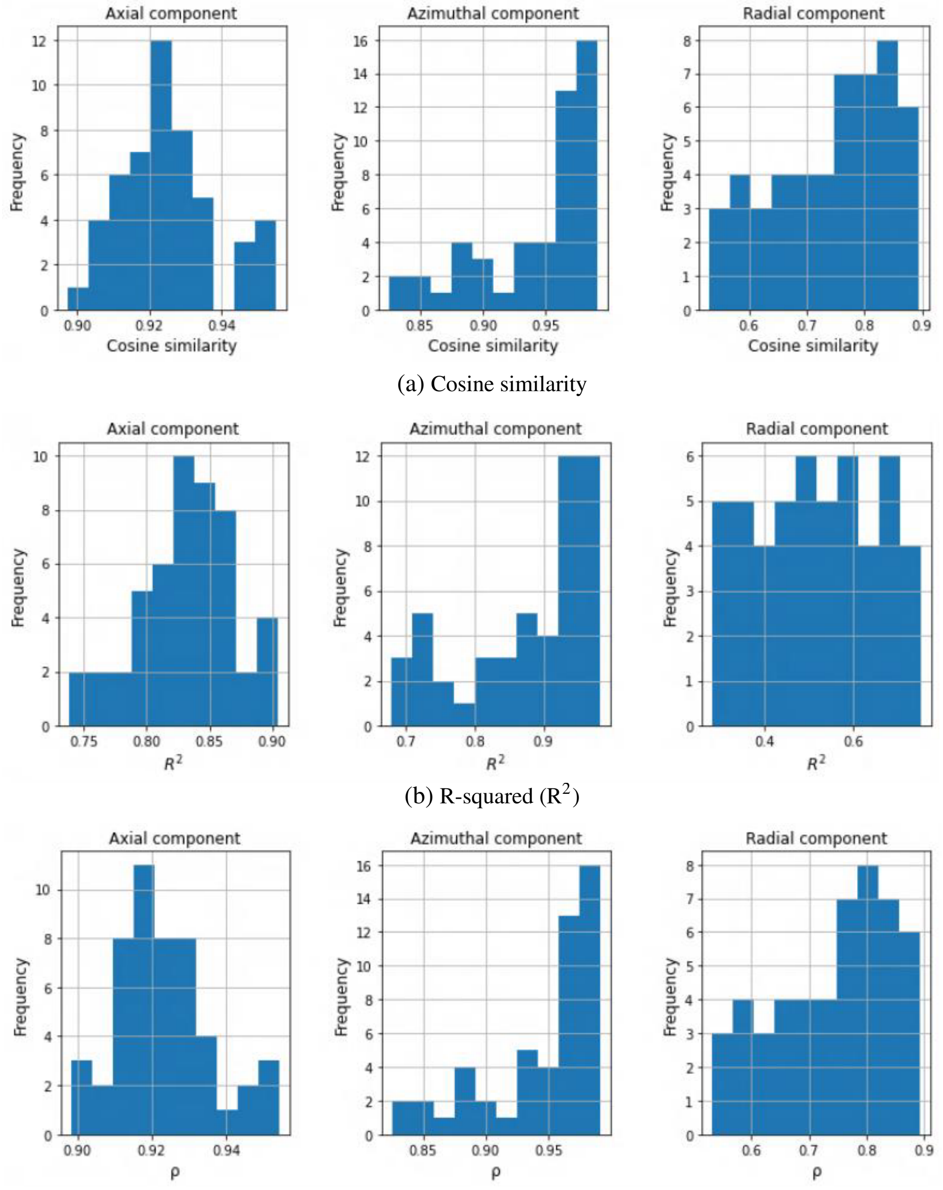


(a) Cosine similarity

(b) R-squared (R ^{2})

(c) Pearson correlation coefficient (ρ)

Figure 9. Comparison metrics between the labels and predictions of the out-of-training dataset; (a) Cosine similarity, (b) R-squared (R^2), (c) Pearson correlation coefficient (ρ).



(a) Cosine similarity

(b) R-squared (R^2)

(c) Pearson correlation coefficient (ρ)

Figure 10. Comparison metrics between the labels and predictions of the unseen pattern dataset; (a) Cosine similarity, (b) R-squared (R^2), (c) Pearson correlation coefficient (ρ).

The V-Net has been validated to be able to accurately predict the magnetic field distribution of unseen samples from the training datasets. Moreover, this model is efficient in terms of the computational expenses. Compared with FEM simulated on the Electromagnetic Simulation Software® (EMS) (EMWorks, Inc., Montreal, Quebec, Canada) and the semi-analytical expressions (SAEs), the V-Net consumes mush less time to be executed than the others. It took 23.2 seconds with a CPU and only 4.6 seconds with a GPU (Tesla T4) for the V-Net's prediction on 50 random samples, whereas it took more than 39000 seconds for the FEM and SAEs to be executed (Table 1), and the multilayer perceptron model MLP [15] took 12019.3 s with a CPU and 743.8 s with a GPU. An example of the parameters utilized for the FEM simulation is listed in Table B6, Appendix B. The three models are executed on the same personal computer with a processor Intel Core i7-9700, 3.0 GHz. The programming language Python 3.9.5 was applied to execute the V-Net, MLP, and SAEs.

5. DISCUSSION AND CONCLUSION

With the advancement of the computational power including the achievement of the graphics processing units (GPUs), deep learning has been demonstrated to be able to learn a wide range of tasks from medical image processing to physics-informed problems [26, 27]. In this study, we applied a deep learning model V-Net to learn the magnetic field distribution created from permanent magnets in 3D space. The ground truth data were synthetically generated from validated semi-analytical expressions.

For the unseen testing dataset, the percentages of the returned metrics greater than 0.95 for the (axial, azimuthal, and radial components) are (99.85, 99.63, 99.78) (Cosimi in %), (94.44, 99.26, 96.74) (R^2 in %) and (98.96, 99.63, 99.70) (ρ in %). Moreover, it is hardly visually distinguishable between the label and predicted fields (Figs. 6–8). These metrics demonstrate that the labels and predictions are in a good agreement. When applying the trained V-Net to compute the field of the samples generated from out of training intervals and unseen pattern ones, the performance of the V-Net decreased. This is a commonly seen phenomenon in the supervised learning where the machine learning model needs to be trained on the similar patterns or within the intervals of the testing datasets to obtain the most possible accuracy. In order to improve the prediction accuracy for datasets from the out of training intervals or unseen patterns, the V-Net can be retrained on the desired intervals; in this case, transfer learning where the pre-trained V-Net which already learned some representations of the training data can be utilised to accelerate the training process.

In this proof-of-concept study, we have trained the deep learning model on datasets generated by permanent magnets whose geometries are the different combinations of basic shapes. The source codes are published in an open Github repository (https://github.com/vantainguyen/3D_V-Net_MagneticField) for the replication and extension of the deep learning model for other desired geometries. Transfer learning [24] where pre-trained model can be utilised to expedite the training process.

It is also demonstrated that the V-Net is much more efficient in terms of the computational expenses than the FEM, SAEs, and MLP models. This means that the V-Net can be a candidate to replace the others in the computation of the magnetic field distribution for the design and optimization of magnetic devices.

ACKNOWLEDGMENT

We would like to thank the EMWORKS Company for providing the license for the EMS® software to conduct the Finite Element Analyses. Moreover, Van Tai is grateful to the University of Queensland for providing him with the Research Training Scholarship.

APPENDIX A. SEMI-ANALYTICAL EXPRESSIONS OF THE MAGNETIC FIELD

For a permanent magnet with the $OXYZ$ coordinate system depicted in Fig. A1, the axial, azimuthal and radial components of the magnetic field generated by this magnet at computed point K in cylindrical coordinate system (r, φ, z) can be expressed as Eqs. (A1), (A2), and (A3) [15]. For some simplified geometries such as cones and elliptical cylinders, these expressions can be further simplified (please refer to [12, 14] for more details).

$$\begin{aligned} &H_{Axial}(K) \\ &= \frac{1}{4\pi\mu_0} \oint\!\!\!\oint_S \frac{\left(J_X \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \vec{l}_X + J_Y \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \vec{l}_Y + J_Z \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \vec{l}_Z \right) \cdot \vec{n}\,(z - z_Q)}{\left({r_Q}^2 + r^2 - 2 r_Q r \cos(\varphi - \varphi_Q) + (z - z_Q)^2 \right)^{\frac{3}{2}}} ds, \end{aligned} \tag{A1}$$

$$H_{Azimuthal}(K) = \frac{1}{4\pi\mu_0}\oiint_S \frac{\left(J_X[1\ 0\ 0]\begin{bmatrix}1\\0\\0\end{bmatrix}\vec{l}_X + J_Y[0\ 1\ 0]\begin{bmatrix}0\\1\\0\end{bmatrix}\vec{l}_Y + J_Z[0\ 0\ 1]\begin{bmatrix}0\\0\\1\end{bmatrix}\vec{l}_Z\right)\cdot\vec{n}r_Q\sin(\varphi-\varphi_Q)}{\left(r_Q^2 + r^2 - 2r_Q r\cos(\varphi-\varphi_Q) + (z-z_Q)^2\right)^{\frac{3}{2}}}ds \tag{A2}$$

$$H_{Radial}(K) = \frac{1}{4\pi\mu_0}\oiint_S \frac{\left(J_X[1\ 0\ 0]\begin{bmatrix}1\\0\\0\end{bmatrix}\vec{l}_X + J_Y[0\ 1\ 0]\begin{bmatrix}0\\1\\0\end{bmatrix}\vec{l}_Y + J_Z[\ 0\ 0\ 1\ ]\begin{bmatrix}0\\0\\1\end{bmatrix}\vec{l}_Z\right)\cdot\vec{n}\left(r - r_Q\cos(\varphi-\varphi_Q)\right)}{\left({r_Q}^2 + r^2 - 2r_Q r\cos(\varphi-\varphi_Q) + (z-z_Q)^2\right)^{\frac{3}{2}}}ds, \tag{A3}$$

where the unit vectors of axes OX, OY, and OZ are denoted as $\vec{l}_X$, $\vec{l}_Y$ and $\vec{l}_Z$, respectively; the unit vector normal to the magnet's surface at the source point Q is represented as $\vec{n}$.

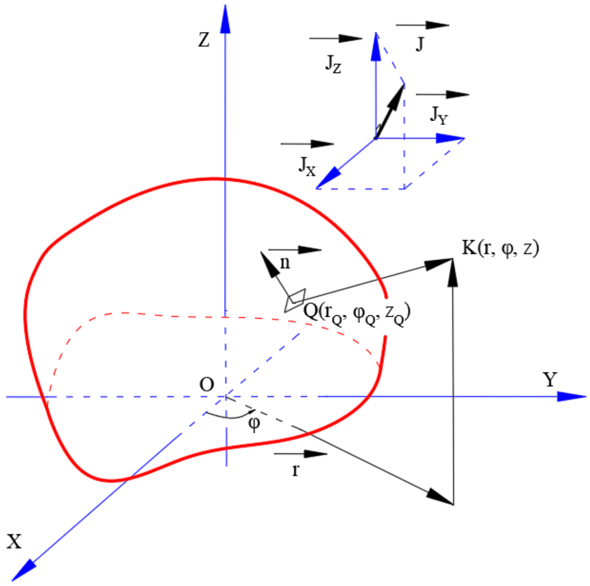


Figure A1. Computational 3D geometry of a permanent magnet.

APPENDIX B. GEOMETRICAL PARAMETERS

Table B1. Training intervals for cones (Fig. B1).

Parameters	Minimum values	Maximum values
$J_{X,Y,Z}$ (T)	-1.5	1.5
β (rad)	$\pi/10$	$\pi/4$
h (mm)	6	20

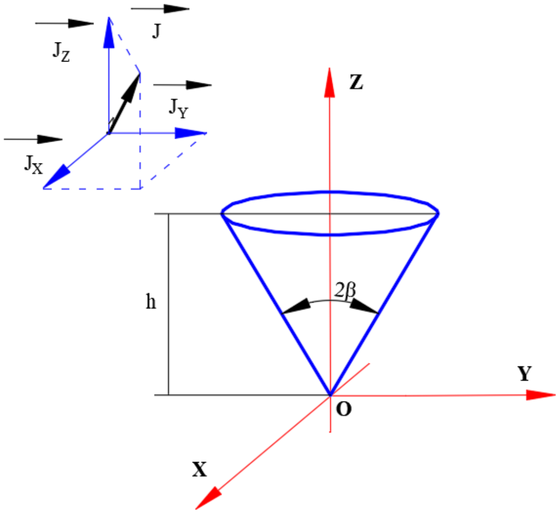


Figure B1. Geometrical parameters of a cone.

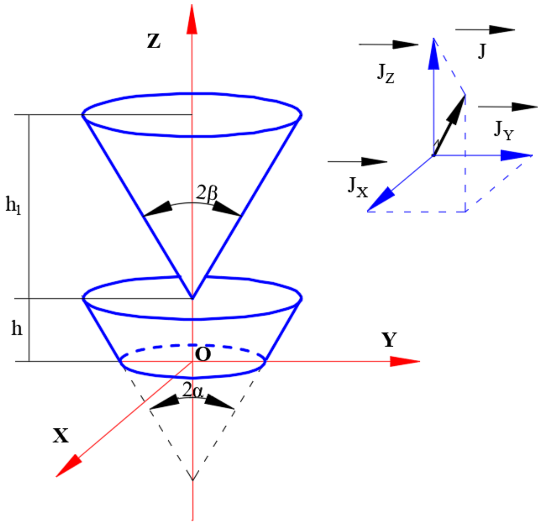


Figure B2. Geometrical parameters of the complex geometry (sectional cone + cone).

Table B2. Training intervals for the complex geometry (sectional cone + cone) (Fig. B2).

Parameters	Minimum values	Maximum values
$J_{X,Y,Z}$ (T)	-1.5	1.5
α (rad)	$\pi/10$	$\pi/3$
h (mm)	8	15
β (rad)	$\pi/10$	$\pi/4$
h_1 (mm)	6	20

Table B3. Training intervals for another complex geometry (elliptical cylinder + cone) (Fig. B3).

Parameters	Minimum values	Maximum values
$J_{X,Y,Z}$ (T)	-1.5	1.5
Semi-axis a (mm)	6	20
Semi-axis b (mm)	6	20
h (mm)	6	10
β (rad)	$\pi/10$	$\pi/4$
h_1 (mm)	6	20

Table B4. Out of training intervals (cones) (Fig. B1).

Parameters	Minimum values	Maximum values
$J_{X,Y,Z}$ (T)	-1.5	1.5
φ (rad)	$\pi/10$	$\pi/4$
h (mm)	21	25

Table B5. Unseen Patterns from the training dataset (circular cylinders) (Fig. B4).

Parameters	Minimum values	Maximum values
$J_{X,Y,Z}$ (T)	-1.5	1.5
R (mm)	6	20
h (mm)	6	20

Table B6. Parameters of the cone (Fig. B1) used in the FEM [14].

Parameters	Values
$J_{X,Y,Z}$ (T)	0, 0, 1
β (rad)	$\pi/6$
h (mm)	10
Mesh elements	1291757
Mesh nodes	216346

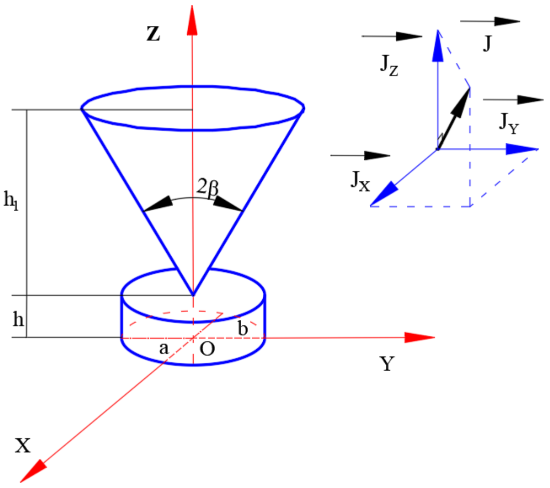


Figure B3. Geometrical parameters of the complex geometry (elliptical cylinder + cone).

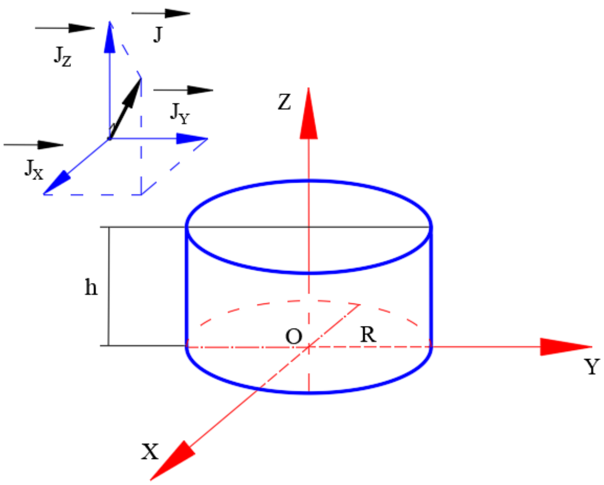


Figure B4. Geometrical parameters of a circular cylinder.

REFERENCES

1. Arias, P., L. Adn-Arcay, B. Puerta-Catoira, A. Madrid, J. Cudeiro, "Transcranial static magnetic field stimulation of M1 reduces corticospinal excitability without distorting sensorimotor integration in humans," *Brain Stimulation: Basic, Translational, and Clinical Research in Neuromodulation*, Vol. 10, No. 2, 340–342, 2017.
2. Huettel, S. A., A. W. Song, and G. McCarthy, *Functional Magnetic Resonance Imaging*, Vol. 1, Sinauer Associates Sunderland, 2004.
3. Atallah, K., S. Calverley, and D. Howe, "Design, analysis and realisation of a high-performance magnetic gear," *IEE Proceedings-Electric Power Applications*, Vol. 151, No. 2, 135–143, 2004.
4. Molokanov, O., P. Dergachev, S. Osipkin, E. Kuznetsova, and P. Kurbatov, "A novel double-rotor planetary magnetic gear," *IEEE Transactions on Magnetics*, Vol. 54, No. 11, 1–5, 2018.

5. Wu, W., H. Lovatt, and J. Dunlop, "Analysis and design optimisation of magnetic couplings using 3D finite element modelling," *IEEE Transactions on Magnetics*, Vol. 33, No. 5, 4083–4094, 1997.
6. Hu, J., J. Zou, F. Xu, Y. Li, Yanchao Fu, "An improved PMSM rotor position sensor based on linear Hall sensors," *IEEE Transactions on Magnetics*, Vol. 48, No. 11, 3591–3594, 2012.
7. Li, K., Y. Xu, Z, Zhao, and M. Q.-H. Meng, "External and internal sensor fusion based localization strategy for 6-dof pose estimation of a magnetic capsule robot," *IEEE Robotics and Automation Letters*, Vol. 7, No. 3, 6878–6885, 2022.
8. O'Connell, J. L., W. S. Robertson, and B. S. Cazzolato, "Optimization of the magnetic field produced by frustum permanent magnets for single magnet and planar halbach array configurations," *IEEE Transactions on Magnetics*, Vol. 57, No. 8, 1–9, 2021.
9. Furlani, E. P., *Permanent Magnet and Electromechanical Devices: Materials, Analysis, and Applications*, Academic Press, 2001.
10. Caciagli, A., R. J. Baars, A. P. Philipse, and B. W. M. Kuipers, "Exact expression for the magnetic field of a finite cylinder with arbitrary uniform magnetization," *Journal of Magnetism and Magnetic Materials*, Vol. 456, 423–432, 2018.
11. O'Connell, J. L., W. S. Robertson, and B. S. Cazzolato, "Simplified equations for the magnetic field due to an arbitrarily-shaped polyhedral permanent magnet," *Journal of Magnetism and Magnetic Materials*, Vol. 510, 166894, 2020.
12. Nguyen, V. T. and T.-F. Lu, "Modelling of magnetic field distributions of elliptical cylinder permanent magnets with diametrical magnetization," *Journal of Magnetism and Magnetic Materials*, Vol. 491, 165569, 2019.
13. Hart, S., K. Hart, and J. P. Selvaggi, "Analytical expressions for the magnetic field from axially magnetized and conically shaped permanent magnets," *IEEE Transactions on Magnetics*, Vol. 56, No. 7, 1–9, 2020.
14. Nguyen, V. T., "Magnetic field distribution of a conical permanent magnet with an application in magnetic resonance imaging," *Journal of Magnetism and Magnetic Materials*, Vol. 498, No. 5, 166136, 2020.
15. Nguyen, V., S. Bollmann, M. Bermingham, and M. S. Dargusch, "Efficient modelling of permanent magnet field distribution for deep learning applications," *Journal of Magnetism and Magnetic Materials*, Vol. 559, 169521, 2022.
16. Mateev, V. and I. Marinova, "Machine learning in magnetic field calculations," *2019 19th International Symposium on Electromagnetic Fields in Mechatronics, Electrical and Electronic Engineering (ISEF)*, 1–2, IEEE, 2019.
17. Milletari, F., N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," *2016 Fourth International Conference on 3D Vision (3DV)*, 565–571, IEEE, 2016.
18. Kelleher, J. D., *Deep Learning*, MIT Press, 2019.
19. Pan, R., T. Yang, J. Cao, K. Lu, and Z. Zhang, "Missing data imputation by K nearest neighbours based on grey relational structure and mutual information," *Applied Intelligence*, Vol. 43, No. 3, 614–632, 2015.
20. Shanker, M., M. Y. Hu, and M. S. Hung, "Effect of data standardization on neural network training," *Omega*, Vol. 24, No. 4, 385–397, 1996.
21. https://www.tensorflow.org/tutorials/load_data/tfrecord, (latest access on Sep. 23, 2022).
22. https://www.tensorflow.org/api_docs/python/tf/keras/optimizers/Adam, (latest access Apr. 23, 2023).
23. Ronneberger, O., P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2015.
24. Weiss, K., T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big Data*, Vol. 3, No. 1, 1–40, 2016.

25. EMS 2020 User Guide, https://www.emworks.com/portal/download, (latest access on Sep. 30, 2022).

26. Chandra, S. S., M. B. Lorenzana, X. Liu, S. Liu, S. Bollmann, and S. Crozier, "Deep learning in magnetic resonance image reconstruction," *Journal of Medical Imaging and Radiation Oncology*, Vol. 65, No. 5, 564–577, 2021.

27. Karniadakis, G. E., I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," *Nature Reviews Physics*, Vol. 3, No. 6, 422–440, 2021.