DTU Library

Magnetic field prediction using generative adversarial networks

Pollok, Stefan; Olden-Jørgensen, Nataniel; Jørgensen, Peter Stanley; Bjørk, Rasmus

Published in:
Journal of Magnetism and Magnetic Materials

Link to article, DOI:
10.1016/j.jmmm.2023.170556

Publication date:
2023

Document Version
Publisher's PDF, also known as Version of record

Journal of Magnetism and Magnetic Materials 571 (2023) 170556

Contents lists available at ScienceDirect

Journal of Magnetism and Magnetic Materials

journal homepage: www.elsevier.com/locate/jmmm

Research article

Magnetic field prediction using generative adversarial networks

Stefan Pollok *, Nataniel Olden-Jørgensen, Peter Stanley Jørgensen, Rasmus Bjørk

Department of Energy Conversion and Storage, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark

A R T I C L E I N F O

Dataset link: https://github.com/spollok/magfield-prediction

Keywords:
Deep learning (DL)
Magnetic field prediction
Magnetostatics
Generative adversarial networks (GANs)
Physics-informed neural networks (PINNs)

A B S T R A C T

Many scientific and real-world applications are built on magnetic fields and their characteristics. To retrieve the valuable magnetic field information in high resolution, extensive field measurements are required, which are either time-consuming to conduct or even not feasible due to physical constraints. To alleviate this problem, we predict magnetic field values at a random point in space from a few point measurements by using a generative adversarial network (GAN) structure. The deep learning (DL) architecture consists of two neural networks: a generator, which predicts missing field values of a given magnetic field, and a critic, which is trained to calculate the statistical distance between real and generated magnetic field distributions. By minimizing the reconstruction loss as well as physical losses, our trained generator has learned to predict the missing field values with a mean reconstruction test error of 6.45% when a large single coherent region of field points is missing, and 10.04%, when only a few point measurements in space are available. This is better by about a factor of two compared to conventional methods such as linear interpolation, splines, and biharmonic equations. We verify the results on an experimentally validated magnetic field.

1. Introduction

Magnetic fields are used in a multitude of scientific and real-world applications, from MRI scanners to electric motors. In all of these applications, the magnetic field must be optimized for the given technology, which typically requires that the magnetic field is characterized. However, to characterize a magnetic field, it has to be determined throughout the volume of interest, regardless of whether the magnetic field is measured using a Hall sensor in an experimental setup or the field is computed using a simulation framework such as analytical modeling [1] or finite element analysis [2]. Determining the magnetic field with increasing resolution is computationally expensive, as is measuring the field in a large number of points for characterizing the field of an experimental setup.

The problem of obtaining a detailed magnetic field from a set of measurements or simulation points is known in a number of domains. In robotics, Gaussian processes have been used to interpolate and extrapolate magnetic field values from a few given data points [3]. As the computational complexity of Gaussian processes renders the approach more or less useless in practice, when the number of observations becomes large, i.e., more than several thousand measurements, the authors model a scalar potential function instead of the 3-D magnetic field. In addition to that, the presented method uses an approximation of the covariance function to model the ambient magnetic field, which inherits information of magnetic field disturbances from the surrounding indoor environment. A robot uses this information to perform localization, and the subsequent robot navigation is highly dependent on the quality of the magnetic field estimation. Le Grand et al. [4] use a simple linear interpolation of the measured mesh points to perform a mapping from coarse, expensive magnetic field measurements to a fine magnetic field estimate, which inherently is a low-order approximation.

In magnetohydrodynamics, the dynamics of conducting fluids have to be described. The predictions of the charged fluid particle trajectories rely on the exact magnetic field values in each location. Given numerical results of a magnetic field evaluation on a discrete grid, the divergence-free magnetic field values at any point in space are obtained by relating the magnetic field to its vector potential using Fourier transforms [5]. The resultant vector potential is then interpolated using cubic splines. Bernauer et al. [6] use similarly a spline-based interpolation.

In geophysics, least-squares collocation is used for the interpolation of the earth anomaly map from given magnetic field measurements of different sources [7]. Another approach to model the geomagnetic field on the Earth's surface is to interpolate the external magnetic field disturbances by Spherical Elementary Current Systems [8].

Moreover, problems exist, where magnetic field values simply cannot be obtained and have to be interpolated. For instance, the photospheric magnetic field in the Sun's polar region is unavailable in specific locations and in order to infer large-scale characteristics, the missing field data is interpolated [9]. Here, the estimation of missing field data

* Corresponding author.
E-mail address: spol@dtu.dk (S. Pollok).

https://doi.org/10.1016/j.jmmm.2023.170556
Received 1 April 2022; Received in revised form 31 January 2023; Accepted 17 February 2023
Available online 21 February 2023

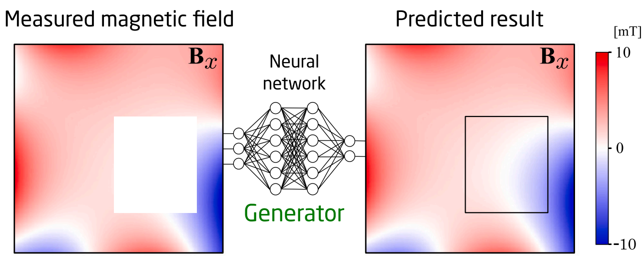


Fig. 1. Simplified overview of the novel DL approach for magnetic field prediction. A generator neural network is trained to predict the missing magnetic field values.

is performed with third-order 2-D polynomial functions fitted to the given data by least squares.

As demonstrated above, the interpolation methods used for magnetic fields differ between domains, as does the numerical accuracy and computational cost of the implementation. Here, we present a novel technique for interpolating and extrapolating magnetic field values based on deep learning (DL). DL is a data-driven approach, where the parameters of artificial neural networks are trained to optimize an objective function, and is proven to be an universal approximator [10]. Recent advances in this research area have led to outstanding results in natural language processing [11], speech recognition [12], and computer vision [13]. The technique has not only proven to be beneficial for the mentioned engineering tasks, DL architectures are now used to solve challenges in natural sciences, e.g., material discovery [14] or drug design [15].

In magnetism research, neural networks have been used, e.g., for approximately solving Maxwell's equations for electromagnetic structures [16] or for solving Maxwell's equations in an inverse manner [17], i.e., inferring the magnetic structure from a given magnetic field. Recently, physics-informed neural networks (PINNs) [18] have been formulated to embed the nonlinear partial differential equations of a physical domain into the DL architecture. That setup makes the prior knowledge of the problem's physics available to the DL method and therefore respects the given constraints during training. An instance of PINNs was adapted to the area of magnetism [19], where Maxwell's equations describe the underlying physical laws of magnetostatics and micromagnetism.

The underlying physical laws have been embedded in a recently emerging DL architecture called generative adversarial networks (GANs) [20]. In that setup, two neural networks are trained: a generator, which outputs a desired target sample, and a discriminator, which checks whether an output sample is real or artificially generated. By adding loss terms to the generator, which relate the generated target samples to the underlying stochastic differential equations, stochastic processes can be approximated [21].

In computer vision, GANs have been used to generate images based on conditions [22]. Song et al. [23] learn a diffusion process from data to noise. By approximating the reverse-time stochastic differential equation, an image can be retrieved from only a few given parts of the original image. Another promising work [24] learns to predict a probability distribution for each pixel value based on its preceding pixel neighbors. Further developments in their architecture and learning procedure have led to the ability to fill in missing pixel values of an RGB image to create a visually appealing and consistent output [25]. Zheng et al. [26] enhance existing work for semantic image inpainting by adding loss terms, which introduce the physical constraints of a geostatistical problem, i.e., to infer the heterogeneous geological field of a few point measurements.

Here, we present a novel approach, where a physics-informed GAN is used to predict missing field values of a magnetic field. Whereas in principle the inpainting region of an image can be of any color as long as it is appealing for the human eye, the distribution and the behavior of a magnetic field are governed by Maxwell's equations. By embedding the physical constraints into the loss function of our DL method, we show that the quality of our predicted field regions can be improved. A generator neural network G shall reconstruct the real, underlying magnetic field $\mathbf{B}$ by inter- and extrapolation on sparsely measured field values $\mathbf{B}_{sparse}$:

$$\mathbf{B} = G(\mathbf{B}_{sparse}). \quad (1)$$

Based on the partly measured field values, we consider two distinct problems. One of that is inpainting, where the magnetic field is given around an area of unknown field values, which are then interpolated by our method as shown in Fig. 1. The second task, which we call outpainting, is a combination of inter- and extrapolation. Hereby, magnetic field values are sparsely measured and the trained neural network has to generate the missing values.

To the best of our knowledge, this is the first application of GANs to magnetic field prediction. We extend previous work [25] to an outpainting task and embed the physical behavior of magnetic fields into additional losses, which the generator neural network is trained on. In addition to the performance of our novel method, we provide an extensive comparison to other state-of-the-art methods used for magnetic field prediction in literature and also compare to magnetic fields measured in a physical experiment. Hereby, we measure a magnetic field produced by multiple hard magnets with a Hall sensor and predict missing field values with our trained generator network.

2. Physics-informed GANs

A method capable of addressing the introduced problem is DL with physics-informed GANs. By adversarial supervision, a generator neural network G learns to produce samples that match the distribution of the ground-truth training data, which are magnetic fields in our case. Given some measured magnetic field values in a predefined area as input, the trained G outputs a complete magnetic field from the learned distribution, which is constrained to match the given measurements and to meet the physical properties of magnetic fields.

2.1. Wasserstein GANs with gradient penalty

In the original formulation of GANs, two neural networks are competing in a min-max game. A generator network G maps a sample $\mathbf{z}$ of a simple noise distribution to a sample $\mathbf{x}$ of the model distribution $\mathbb{P}_g$ as $G(\mathbf{z};\theta_g)$, where θ_g are the trainable network parameters. Simultaneously, a discriminator network D is trained to output a scalar for a given sample in the form $D(\mathbf{x};\theta_d)$, where θ_d are its network parameters to be optimized. The idea is that G tries to fool D by generating samples that resemble the ones taken from the real target space distribution $\mathbb{P}_r$. On the other hand, D improves in distinguishing real from generated samples during training. That should force G to generate even more realistic samples by increasing the similarity between $\mathbb{P}_g$ and $\mathbb{P}_r$. To achieve the described behavior, the training objective for G and D is defined as follows:

$$\min_G \max_D \mathop{\mathrm{E}}_{\mathbf{x}\sim\mathbb{P}_r}\left[\log\left(\frac{\mathbb{P}_r(x)}{\mathbb{P}_r(x)+\mathbb{P}_g(x)}\right)\right] + \mathop{\mathrm{E}}_{\tilde{\mathbf{x}}\sim\mathbb{P}_g}\left[\log\left(\frac{\mathbb{P}_g(\tilde{x})}{\mathbb{P}_r(\tilde{x})+\mathbb{P}_g(\tilde{x})}\right)\right], \quad (2)$$

where $\mathbf{x} \sim \mathbb{P}_r$ denotes that a sample $\mathbf{x}$ is drawn from the real target distribution $\mathbb{P}_r$, $\tilde{\mathbf{x}} = G(\mathbf{z};\theta_g)$, and E is the expected value, which is a generalization of the weighted average in probability theory. If $\mathbb{P}_r = \mathbb{P}_g$, it can be shown that the objective reaches a global minimum of $-\log 4$, and we would have obtained an ideal generator G. However, this training procedure turns out to be unstable in practice due to mode collapsing of the discriminator and vanishing gradients.

In Wasserstein GANs with gradient penalty (WGAN-GP) [27], these problems are alleviated by defining D as a critic, which outputs the Wasserstein-1 distance W [28] between $\mathbb{P}_g$ and $\mathbb{P}_r$ instead. This is a

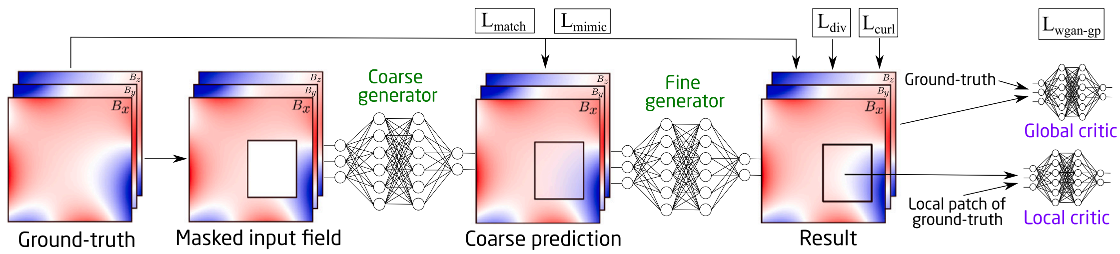


Fig. 2. Illustration of the DL architecture used for magnetic field prediction. A two-step generation process, which consists of down- and upsampling across multiple convolutional layers, produces missing field values of a masked input magnetic field. The result is evaluated by a local and a global critic, which are again neural networks consisting of several convolutional layers. In addition to the $L_{wgan-gp}$, the $l1$ reconstruction losses, L_{match} and L_{mimic}, and the physical losses, L_{div} and L_{curl}, are calculated for updating the parameters of the generator networks in order to minimize the overall loss function.

statistical distance, which describes the similarity between two probability distributions as a way of optimal transport. Informally, one can imagine $\mathbb{P}_g$ and $\mathbb{P}_r$ as the mass distribution of two differently shaped piles of earth of the same total mass. The Wasserstein-1 distance then determines the minimal distance of transporting mass units to transform one pile of earth into the other one [29]. The WGAN-GP objective function is defined as:

$$\min_{G} \max_{D \in \mathcal{D}} \mathop{\mathrm{E}}_{\mathbf{x} \sim \mathbb{P}_r} [D(\mathbf{x}; \theta_d)] - \mathop{\mathrm{E}}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [D(\tilde{\mathbf{x}}; \theta_d)], \tag{3}$$

where $\mathcal{D}$ is the set of 1-Lipschitz functions. In our case, the set of 1-Lipschitz functions is defined as:

$$\frac{\| D(\hat{\mathbf{x}}_1; \theta_d) - D(\hat{\mathbf{x}}_2; \theta_d) \|_2}{\| \hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_2 \|_2} \leq 1, \tag{4}$$

where $\hat{\mathbf{x}}$ is a sample drawn from the probability distribution $\mathbb{P}_{\hat{\mathbf{x}}}$, which is combining $\mathbb{P}_r$ and $\mathbb{P}_g$ by sampling uniformly from straight lines between pairs of points sampled from these two distributions. As D is a neural network and fully differentiable, it implicitly follows that $\nabla_{\hat{\mathbf{x}}} D(\hat{\mathbf{x}}; \theta_d) \leq 1$ has to hold true, if $D \in \mathcal{D}$. Further, we refer to D as critic from here on. Under an optimal critic, the generator network parameters θ_g are trained to minimize $W(\mathbb{P}_g, \mathbb{P}_r)$.

2.2. Loss function for magnetic field prediction

In WGAN-GP, a gradient penalty term is added to the standard WGAN loss function:

$$\begin{aligned} L_{wgan-gp} = & \underbrace{\mathop{\mathrm{E}}_{\mathbf{x} \sim \mathbb{P}_r} [D(\mathbf{x}; \theta_d)] - \mathop{\mathrm{E}}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [D(\tilde{\mathbf{x}}; \theta_d)]}_{L_{wgan}} \\ & + \underbrace{\lambda_{gp} \mathop{\mathrm{E}}_{\hat{\mathbf{x}} \sim \mathbb{P}_{\hat{\mathbf{x}}}} [(\| \nabla_{\hat{\mathbf{x}}} D(\hat{\mathbf{x}}; \theta_d) \|_2 - 1)^2]}_{L_{gp}}, \end{aligned} \tag{5}$$

where λ_{gp} is the gradient penalty coefficient. The additional loss term ensures that the norm of the gradients of the critic parameters θ_d is close to 1 for adherence to Eq. (5), which lets D be an optimal realization of the set of 1-Lipschitz functions. The loss $L_{wgan-gp}$ is then backpropagated to update the network parameters of the generator and the critic.

Our work is inspired by generative image inpainting from the research area of computer vision, where GANs are trained to inpaint the missing region of a corrupted image $\mathbf{x}_{sparse}$. Ideally, the generated result $\tilde{\mathbf{x}}$ shall match $\mathbf{x}_{sparse}$ in all the image pixels available and mimic the ground-truth full image $\mathbf{x}$. Hence, an $l1$ loss L_{match} between the predicted result $\tilde{\mathbf{x}}$ and the given input image $\mathbf{x}_{sparse}$, and a second $l1$ loss L_{mimic} between $\tilde{\mathbf{x}}$ and the ground-truth training sample $\mathbf{x}$ are formulated:

$$\begin{aligned} L_{match} &= \| \mathbf{x}_{sparse} \odot (\mathbf{1} - \mathbf{m}) - \tilde{\mathbf{x}} \odot (\mathbf{1} - \mathbf{m}) \|_1, \\ L_{mimic} &= \| \mathbf{x} \odot \mathbf{m} - \tilde{\mathbf{x}} \odot \mathbf{m} \|_1, \end{aligned} \tag{6}$$

where $\mathbf{m}$ is a binary mask with a pixel value of 1 for missing magnetic field values and a value of 0 if field measurements are available. The symbol $\odot$ denotes, here and throughout the paper, the Hadamard product. The result of the Hadamard product is a matrix filled in each element i, j with the element-wise product of the entries i, j of the two original matrices. All matrices involved in this mathematical operation are of the same dimension.

For magnetic field prediction, we have additional information of the underlying physics of magnetic fields. We not only want to generate a visual appealing result, we also want the generated magnetic field to be constrained by Maxwell's equations. With addition of physical loss terms to the loss function, our DL method becomes physics-informed, and it can be seen as a regularization for generating magnetic fields. Samples from our target space distribution $\mathbb{P}_r$ are discrete magnetic fields values on a regular grid, $\mathbf{B} = \mathbf{x}$. The first physical loss term is Gauss's law for magnetism, which states that:

$$L_{div} = \nabla \cdot \tilde{\mathbf{B}} \stackrel{!}{=} 0, \tag{7}$$

where $\tilde{\mathbf{B}}$ is the generated magnetic field prediction. If we further assume the absence of electric current density or changing electric field over time, Ampère's circuital law can be simplified to:

$$L_{curl} = \nabla \times \tilde{\mathbf{B}} \stackrel{!}{=} 0. \tag{8}$$

Our final loss function used during training is formulated as follows:

$$\begin{aligned} L = & \lambda_{wgan\text{-}gp} L_{wgan-gp} + \lambda_{match} L_{match} + \lambda_{mimic} L_{mimic} \\ & + \lambda_{div} L_{div} + \lambda_{curl} L_{curl}, \end{aligned} \tag{9}$$

where $\lambda_{wgan-gp}$, λ_{match}, λ_{mimic}, λ_{div}, and λ_{curl} are the coefficients for each single loss term and define their relative importance.

Algorithm 1 WGAN-GP for magnetic field prediction

```
while G has not converged do
    for 5 iterations do
        Sample magnetic fields B from training data;
        Generate random masks m for B;
        Construct input fields B_sparse ← B ⊙ (1 − m);
        Get result B̃ ← B_sparse + G(B_sparse, m) ⊙ m
        Update D with L_wgan−gp
    end for
    Update G with L_match, L_mimic, L_div, L_curl, and L_wgan−gp
end while
```

2.3. Neural network architecture

The DL architecture used for magnetic field prediction is adapted from Yu et al. [25] and visualized in Fig. 2. To demonstrate the concept and to make visualization of the results easier, we choose to input a 3-D magnetic field measured in a 2-D rectangular area and output an inter- or extrapolated 3-D magnetic field in this area. These fields are multiplied with a binary mask $\mathbf{m}$ during training as follows:

$$\mathbf{B}_{sparse} = \mathbf{B} \odot (\mathbf{1} - \mathbf{m}). \tag{10}$$

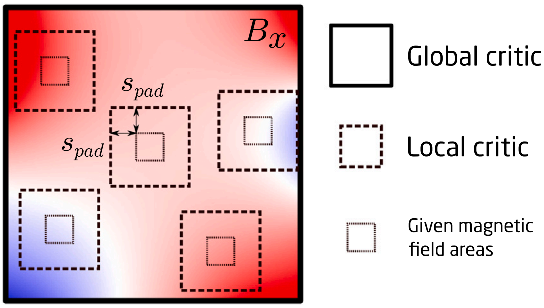
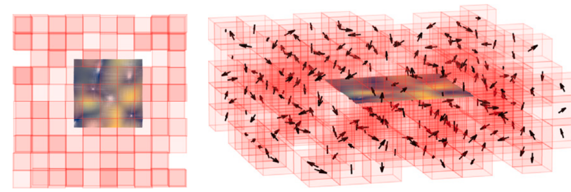


Fig. 3. Magnetic field regions of the 2-D measurement area, which the two neural networks serving as local and global critic use as input for the outpainting task.



(a) Top view (b) Side view

Fig. 4. Virtual experimental setup. Multiple hard magnets shaped as cubic prisms are placed randomly in a grid of resolution $10 \times 10 \times 5$. The magnetic field samples used for training and testing of our novel method are computed in a 2-D area enclosed by this structure. It is assured that no magnetic material can be found in the measurement area.

The two-step generating process is designed in the style of residual learning [30] and can be described as follows:

$$\begin{aligned}\mathbf{B}_{coarse} &= G_{coarse}(\mathbf{B}_{sparse}, \mathbf{m}),\\ \tilde{\mathbf{B}} &= G_{fine}(\mathbf{B}_{coarse} \odot \mathbf{m} + \mathbf{B}_{sparse}, \mathbf{m}).\end{aligned} \tag{11}$$

A generator network G_{coarse} generates a coarse prediction by applying a sequence of convolutional layers on $\mathbf{B}_{sparse}$ and the applied mask $\mathbf{m}$. First, the input field is downsampled to a smaller resolution with an increased number of channels, so that the same amount of information can be preserved with subsequent convolutions being computationally less expensive. Second, several convolutions with differently scaled filters are performed on the downsampled image to increase the field-of-view of the model and to enable encoding at multiple scales. Finally, the data is upsampled with interpolations to the original size, which results in a coarse prediction $\mathbf{B}_{coarse}$.

A second generator network G_{fine} takes $\mathbf{B}_{coarse}$ and $\mathbf{B}_{sparse}$ as input, and produces $\tilde{\mathbf{B}}$ in a similar manner as the coarse generator. Parallel to that, the magnetic field is split up into small patches of 3×3 pixels in a second branch. The relative importance between these patches and the missing pixels is calculated, which is then used for an improved reconstruction. The idea behind that so-called contextual attention branch is to overcome the locality in the convolutional layers and to enhance it with a global information flow from distant magnetic field pixels. The convolution and the attention branch are concatenated before upsampling to the original resolution.

On $\tilde{\mathbf{B}}$, the losses L_{match}, L_{mimic}, L_{div}, and L_{curl} can be directly calculated. For the adversarial loss $L_{wgan-gp}$, we need to employ a critic neural network, which tells us the Wasserstein-1 distance between the original and the generated magnetic fields. Iizuka et al. [31] show that it is beneficial to split the critic into a global critic network, which evaluates the whole image, and a local critic network, which determines the quality of the filled-in regions.

In our framework, we extend the setup to work computationally efficient also with the outpainting task, which can be seen as an inverted inpainting task. Hereby, only small regions of magnetic field measurements across the 2-D area are given. The missing field values around have to be inter- and extrapolated. Implementing this task in the framework of Yu et al. [25] is straightforward. However, special care has to be taken when creating the local patches for the outpainting task. Instead of naively inverting the mask values and calculating the local patch for nearly the whole image, we define small boxes with padding size s_{pad} around the given field patch as shown in Fig. 3.

As the convolutional neural networks used in G_{coarse} and G_{fine} are resolution-independent, the size and shape of $\mathbf{B}_{sparse}$ can vary during inference time. Similarly, the applied missing regions can be arbitrarily chosen by setting the mask pixel values to 1. The complete training procedure is summarized in Algorithm 1. As usual with GANs, the neural network parameters of the critic are updated five times before the next update for the generator parameters is performed.

3. Experiments

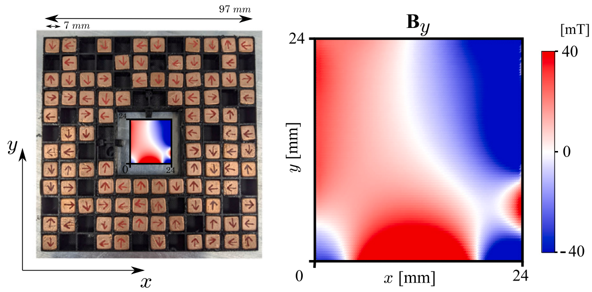
To check the performance of our novel method for magnetic field prediction, we introduce a virtual setup, where our open-source micromagnetism and magnetostatic modeling framework MagTense [1] is used to place a 3-D construct of hard magnets around a 2-D area and to compute its resulting magnetic field. As shown in Fig. 4, multiple hard magnets are placed randomly with a probability of 50% in a grid of resolution $10 \times 10 \times 5$. Each magnet is shaped as a cubic prism with a fixed side length of 0.1 cm and has a remanent magnetization of 1.2 T, but with the easy axes of the single magnets randomly varying. While the field here is generated with cubic magnets, there is no loss of generality, as the different magnetizations and locations of the hard magnets produce a huge variety of magnetic fields in the central area of the grid. However, in a future work it will be interesting in a future work to train on magnetic fields generated by, e.g., cylindrical magnets or spheres.

The enclosed 2-D rectangular area in the center is left free of magnet material and varying in side length ranging from 0.1 to 0.4 cm. This produces multiple field length scales and a changing number of magnets at the edge of that area between different realizations. We compute the resulting 3-D magnetic field with a resolution 256×256 pixels and store 20,000 samples of these into a dataset, which is then used to train our neural networks. Additionally, we store a layer of magnetic field calculations above and a layer below to later be able to calculate the divergence and curl of the magnetic field with a finite difference method.

Moreover, we build a physical setup with real neodymium (NdFeB) magnets in our laboratory and measure the magnetic field with a Hall sensor. We print a 3-D holder with 12×12 spots and place 97 NdFeB magnets of cubic shape with a side length of 0.7 cm. The hard magnets have a magnitude of 1.29–1.32 T and their easy axes lie in the xy-plane. However, production variations lead to small deviations from that plane. In the center of the holder is a hole of size 6 cm $\times$ 6 cm, similar to our virtual experiment. As ground-truth data, we then measure the magnetic field in the enclosed 2-D area. In Fig. 5, this specific setup is depicted along with the y-component of the magnetic field.

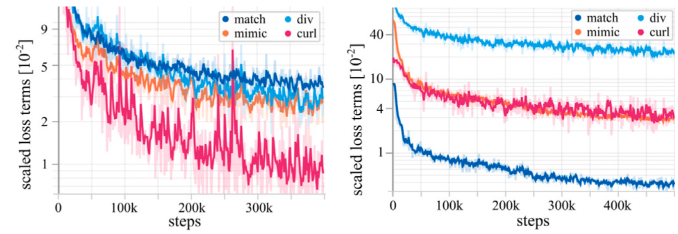
For each of these setups, we then perform an inpainting and an outpainting task. For inpainting, a single region in the 2-D measurement area is missing and has to be interpolated. For outpainting, small regions across the measurement area are given and the missing magnetic field values are inter- and extrapolated with the given information. For outpainting, we assume that the sources generating the field are not present within the region where the field is being outpainted, thus $\nabla \times \mathbf{B} = 0$ applies. This is a realistic condition, as this can also be ensured experimentally or in applications.

For the inpainting problem, this is in principle a closed domain Poisson problem with suitable boundary conditions and if the values of the field are known fully on the boundary, these uniquely determine the solution, i.e., the field in the inpainting region. However, the finite



(a) Top view (b) Measured magnetic field

Fig. 5. The physical experimental setup. Fig. 5(a) shows the 12 × 12 holder with 97 NdFeB magnets. In the enclosed 2-D area, we measure the magnetic field. The y-component $\mathbf{B}_y$ is visualized in Fig. 5(b). For brevity, we omit $\mathbf{B}_x$ here and as the easy axes of the single hard magnets are in the xy-plane, it follows that $\mathbf{B}_z = 0$.



(a) Inpainting - 144 × 144 (b) Outpainting - 16 × 16

Fig. 6. Overview of the scaled reconstruction and physical loss terms during training. During inpainting, the loss terms are in the same range, whereas in the outpainting task, L_{div} is an order of magnitude larger than the scaled L_{mimic} and L_{curl}. The scaled L_{match} is an order of magnitude smaller compared to these.

resolution on the boundary limits the uniqueness of the solution, also given the fact that the magnetic field is a 3-D quantity and the boundary values are only known on a 2-D slice. Secondly, it is often the case that the more training data can be supplied to the DL architecture, the better predictions the trained model will give, assuming that the training data is coherent and samples the region where magnetic field prediction is performed. Hence, we make as much information as possible available to the neural network, including the field outside the boundary of the inpainting region, to overcome the mentioned limitations of only the boundary values given.

For inpainting, we extend the vision of the local critic to be 4 pixels larger across the masked area with the idea that the generator learns even better to predict magnetic field inserts with a smooth transition across the edge from the given to the predicted area. We define four sub-tasks with varying side lengths of the missing quadratic area. For each of the side lengths of 48, 96, 144, and 192 pixels, we train a separate generator neural network. To generalize better to unseen mask sizes, we further vary the side length of the masks between batches up to 25% from the side length it is trained for. Each training batch consists of 25 samples for which the loss function of Eq. (9) is calculated. The neural network parameters of the generator and the critic are then updated with the gradients of this batch loss using the Adam optimizer [32] with a learning rate of 1e–4. On an NVIDIA GeForce RTX 3090, this results in an almost full GPU memory usage of its 24 GB. We train each of the setups for at least 300,000 iterations, where the training time differs for different mask sizes. On average, it takes approximately 0.5 s/batch. As starting point, we take the values for the penalty coefficients directly from Yu et al. [25]. The newly introduced hyperparameters λ_{div} and λ_{curl} are set based on manual inspection of the error magnitude in order to scale L_{div} and L_{curl} to a similar range as the other loss terms. We have visualized the scaled loss terms used for the generator updates in Fig. 6. Eventually, the coefficients are defined as follows: $\lambda_{wgan-gp} = 0.001$, $\lambda_{gp} = 10$, $\lambda_{match} = 7.2$, $\lambda_{mimic} = 3.6$, $\lambda_{div} = 500$, and $\lambda_{curl} = 30{,}000$.

For outpainting, we introduce a setup, where 20 regions of 16 × 16 magnetic field values across the measurement area are given, and a second setup with only 20 point measurements being available. For each of these setups, we train a generator neural network similarly to the inpainting task. Now, we use a batch size of 48 samples, which leads to better convergence in this task. Running for 500,000 iterations on two NVIDIA GeForce RTX 3090 in parallel, training time results in 0.63 s/batch. As the divergence and curl losses become larger in the outpainting task, we adjust $\lambda_{match} = 10$, $\lambda_{mimic} = 2.4$, $\lambda_{div} = 120$, and $\lambda_{curl} = 24{,}000$. Fig. 6(b) indicates to further decrease λ_{div} to obtain more similar ranges of the scaled loss terms in future training runs. A more extensive hyperparameter search is likely to reveal an improved set of hyperparameters.

The code, pretrained models, and examples are available at: https://github.com/spollok/magfield-prediction.

Table 1
MAPE [%] of inpainting task for different mask sizes. The method with the lowest MAPE on each sub-task with 250 test samples is marked in bold. Our method is retrained on each of the specific sub-tasks. The subscripts in the last row indicate that we evaluate our method trained on one mask size only.

	48	96	144	192
Linear	5.10	16.17	27.76	37.93
Spline	**0.52**	**3.87**	10.93	19.45
Biharmonic [34]	**0.52**	4.04	10.55	17.78
WGAN-GP [25]	3.91	4.81	8.49	11.51
Ours	3.10	4.46	**6.45**	**9.40**
Ours_{144}	3.35	4.06	**6.45**	15.43

4. Results

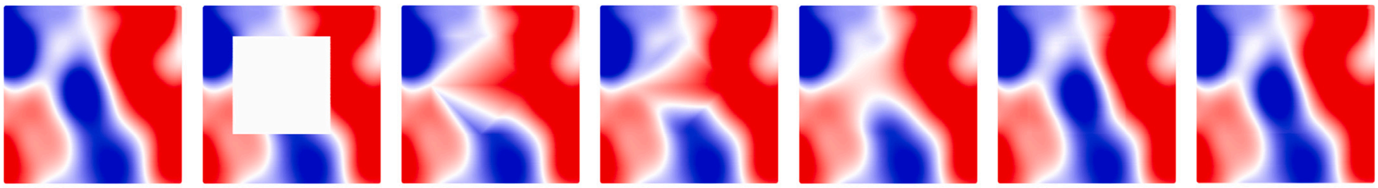
In the following section, we evaluate our novel method for magnetic field prediction and benchmark its performance with current state-of-the-art methods found in literature. In addition to our method, we solve the tasks with a linear and a cubic spline-based interpolation from SciPy [33], biharmonic equations [34], a Scikit [35] implementation of Gaussian processes [36] with a radial-basis function kernel, and the adapted WGAN-GP method from Yu et al. [25] without the additional physical terms in the loss function. We skip evaluating Gaussian processes for the inpainting task as its computational complexity scales with $\mathcal{O}(n^3)$, where n is the number of given magnetic field measurements, and becomes computationally prohibitive for this task. Moreover, we do not apply linear interpolation for the outpainting task as its implementation does not support extrapolation to magnetic field points outside the convex hull of the given field measurements.

In principle, 3-D magnetic fields can be represented by a specific set of spherical harmonics [37] or some other set of alternative basis vectors. For a true 3-D problem, if the field was known completely on the boundary of the inpainting region, the coefficients of the spherical harmonics could be fit to the available magnetic field measurements, and then magnetic field values could be predicted in the missing field points. However, we consider here the problem where only a 2-D measurement area of such a 3-D magnetic field is considered. Using 2-D eigenfunctions, which are fitted to the boundary values of this 2-D measurement area, do not allow for magnetic field predictions, as the field varies in the third dimension. Therefore, we limit the comparison of our DL approach to the state-of-the-art methods mentioned above, and leave the potential use of spherical harmonics for 3-D magnetic field prediction in a 2-D measurement area open for future research.

4.1. Virtual setup - Inpainting

In Table 1, we compare our method to four other methods used in literature for magnetic field prediction based on the mean absolute percentage error (MAPE) between ground-truth and the predicted magnetic field values. It is defined as:

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{\sum \|(\mathbf{x}_i - \tilde{\mathbf{x}}_i)\odot \mathbf{m}\|_1}{\sum \|\mathbf{x}_i \odot \mathbf{m}\|_1}, \tag{12}$$



(a) Ground-truth (b) Masked input (c) Linear (d) Spline (e) Biharmonic [34] (f) WGAN-GP [25] (g) Ours

Fig. 7. Qualitative analysis of the inpainting task with a mask side length of 144 pixels. Only the trained methods can grasp the curved behavior of the missing magnetic field. Our method further enhances smoothness and predicts a more correct curvature compared to the standard WGAN-GP approach.

where n is the total number of test samples, which have not been used during training of the neural networks. Each benchmark is run with $n = 250$. Unsurprisingly, linear interpolation shows poor performance the larger the area of missing magnetic field values becomes. An interpolation method based on cubic splines performs the best for small masks with side lengths of 48 and 96 pixels. The biharmonic interpolation performs similarly good, whereas the learning-based WGAN-GP methods, the DL architecture from Yu et al. [25] and ours, become valuable with increasing amount of missing magnetic field values to predict. Our method, the physics-informed version of WGAN-GP, performs best on the masks with side lengths of 144 and 192 pixels. When further comparing the correctness of the magnetic field physics, our method has the lowest divergence loss with 0.20 mT/pixel and the lowest curl loss with 0.71 μT/pixel on the task with a mask side length of 192 pixels. Here, the spline-based method has a divergence loss of 0.41 mT/pixel and a curl loss of 2.37 μT/pixel.

To emphasize the advantage of our method on larger masks, we employ a qualitative analysis of the inpainting task with a mask side length of 144 pixels. The given magnetic field has a shape of 256 × 256 pixels for each of the three components $\mathbf{B}_x$, $\mathbf{B}_y$, and $\mathbf{B}_z$. For visualization purposes, we only show the ground-truth field distribution of $\mathbf{B}_x$, which is depicted in Fig. 7(a). After masking the ground-truth, Fig. 7(b) serves as the input for the interpolation methods. Only the WGAN-GP methods in Figs. 7(f) and 7(g) grasp the shape in the missing area correctly. The other methods produce sub-optimal results, which can be partly explained with the missing information from the other two magnetic field components, $\mathbf{B}_y$ and $\mathbf{B}_z$, as these methods are interpolating missing values of one component at a time and hence do not include potentially useful, available information of the magnetic field. On the other hand, the learning-based WGAN-GP approaches act directly on all the three components and process them together. If we had interpolated the magnetic scalar potential ψ_M with only one component, the other state-of-the-art methods might potentially have reveal better predictions. Subsequently, the resulting magnetic field can then be derived with $\mathbf{B} = -\nabla\psi_M$. However, in a real-world setup, normally the three magnetic field components are measured, and when these are measured incompletely, one does not know ψ_M.

Regarding the inference time, i.e., the computation time for predicting the missing magnetic field values, the WGAN-GP methods are with 2.48 s competitive with linear interpolation (1.58 s) and cubic splines (1.70 s), as shown in Table 3. The biharmonic equations method needs 43.64 s for one test sample.

For the same task of inpainting with a masked area of size 144 × 144 pixels, we calculate the pixel-wise MAE dependent on the closest given magnetic field measurement. It can be seen in Fig. 8 that our method outperforms other interpolation methods the further a magnetic field value to be predicted is away from the measured region. This occurs around a distance of 17 pixels from the mask edge. On smaller distances, there remains a small MAE of around 4 mT, which makes the edge of the predicted field region visually distinguishable from given magnetic fields with low field values. In contrast to the other interpolation methods, the generator G_{fine} generates a full image with a resolution of 256 × 256 pixels, from which only the masked area is used as prediction for missing field values. Even though the generator

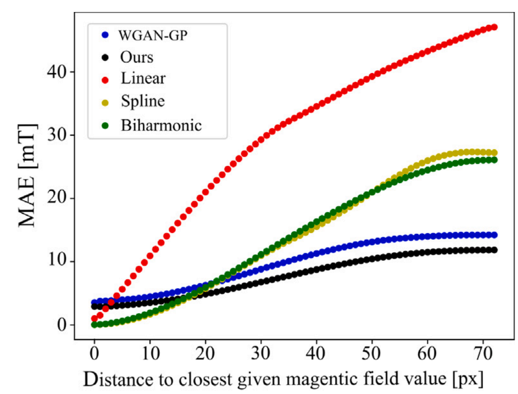


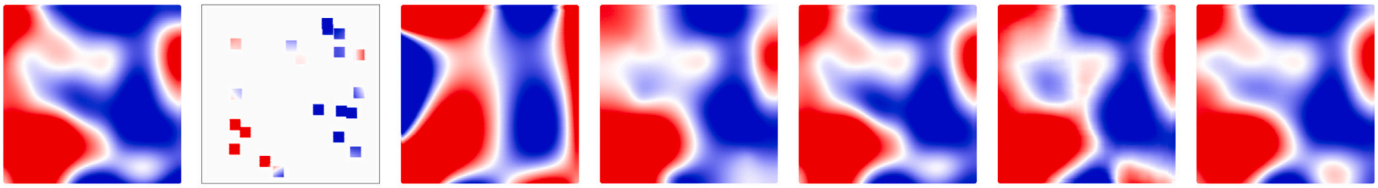
Fig. 8. Pixel-wise MAE for distance to next given measurement in the input magnetic field on the inpainting task with a masked area of size 144 × 144 pixels.

is trained with L_{match}, it does not succeed to generate magnetic fields that match the given measurements exactly at the edges.

We further investigate how resilient our method is to different mask sizes during inference time compared to the mask size the generator neural network was trained on. When using a generator, which was trained on mask sizes with a shape of 144 × 144 pixels, then, as shown in the last row of Table 1, the MAPE for smaller mask sizes is similar to the generator specifically trained on that mask size and twice as large on the mask size with a side length of 192 pixels.

4.2. Virtual setup - Outpainting

In addition to inpainting, we evaluate our method on two outpainting tasks with 100 test samples, respectively. In both tasks, there are 20 regions of field measurements in each sample available. These regions have the size of 1 × 1 pixel in the first sub-task and 16 × 16 pixels in the second one. In Table 2, the MAPE and the physical losses, L_{div} and L_{curl}, of our method are compared to the losses of four other methods found in literature to perform magnetic field prediction inside and outside the convex hull of given magnetic field points. Our learning-based, physics-informed method performs best on predicting missing magnetic field values in the setup with 20 point measurements given with an MAPE of 22.58%. The biharmonic equations, Gaussian processes, and the standard WGAN-GP based method achieve comparable results with an MAPE of slightly above 25%, whereas the interpolation method based on cubic splines leads to a large representation loss and non-physical predictions, i.e., the divergence and curl of the predicted magnetic field are substantially greater than 0. For 20 given regions across the measurement area with a side length of 16 pixels, Gaussian processes perform the best with a low MAPE of 4.74% and only a small divergence of 0.24 mT/pixel. This is comparable to the error rates in the inpainting task. Our method outperforms the other methods in the curl loss of the magnetic field, while having twice the MAPE of Gaussian processes for this sub-task. It is important to mention that Gaussian processes are implemented to process the three components



(a) Ground-truth (b) Masked input (c) Spline (d) Biharmonic [34] (e) Gaussian (f) WGAN-GP [25] (g) Ours

Fig. 9. Qualitative analysis of the outpainting task with 20 measurement regions of size 16 × 16 pixels. Visually, Gaussian processes and our method achieve to reconstruct the ground-truth magnetic field almost perfectly.

Table 2
Losses for the two outpainting tasks.

	1 × 1			16 × 16		
	MAPE [%]	L_{div} [mT/px]	L_{curl} [μT/px]	MAPE [%]	L_{div} [mT/px]	L_{curl} [μT/px]
Spline	193	7590	2228	71.81	2.99	171
Biharmonic [34]	28.58	1.27	5.21	16.84	0.79	2.77
Gaussian	25.96	0.96	4.28	**4.74**	**0.24**	1.20
WGAN-GP [25]	27.05	1.31	7.87	17.70	0.86	4.57
Ours	**22.58**	**0.87**	**3.51**	10.04	0.39	**0.99**

Table 3
Inference time [s] during different sub-tasks.

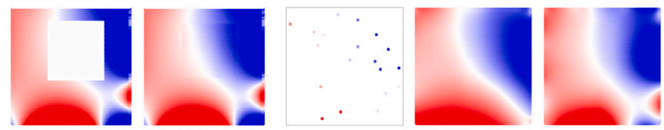
Inpainting	Linear	Spline	Biharmonic [34]	WGAN-GP [25]	Ours
144 × 144	**1.58**	1.70	43.64	2.48	2.48
Outpainting	Gaussian	Spline	Biharmonic [34]	WGAN-GP [25]	Ours
16 × 16	233.37	**0.26**	294.45	2.48	2.48

of the magnetic field separately. Here, predicting an unknown value, e.g., $B_{x,*}$, at location (x_*, y_*) is constructed as a linear combination of kernel functions $\sum_{i=1}^{n} \alpha_{x,i} k((x_*, y_*), (x_i, y_i))$, where (x_i, y_i) is the location of the ith of n available field measurements, and $\alpha_{x,i}$ incorporates the given information of $B_{x,i}$ at these locations. We refer the reader to Rasmussen [36] for a more detailed derivation of $\boldsymbol{\alpha}$. As k in Gaussian processes, we choose a radial-basis function kernel with a length scale exceeding the 2-D measurement area to resemble the smooth nature of magnetic fields. The underlying squared-exponential similarity measure of the chosen k seems to be suited well for magnetic field prediction. An example of outpainting with Gaussian processes is given in Fig. 9(e). On that sample, only our method is able to retrieve missing information similarly good and can produce a visually appealing result as shown in Fig. 9(g).

When comparing the inference times of the different methods on the outpainting task with 16 × 16 pixels, the learning-based WGAN-GP methods have the same computation time of 2.48 s as for the inpainting task. In contrast to the other methods, the size and amount of mask has no influence on the inference time of the DL approach as the computation path from given mask input to prediction of the full magnetic field stays the same. However, as mentioned before, the inference time of Gaussian processes scale with $\mathcal{O}(n^3)$ and evaluates to 233 s for a single image here. The biharmonic equations take 294 s to evaluate, while the spline-based method is the fastest with 0.26 s.

4.3. Experimental setup

To further validate the performance and generalizability of our approach, we use the trained generator of our learning-based, physics-informed WGAN-GP, which was trained on the 3-D virtual experimental setup from Fig. 4, to make predictions in the 2-D physical setup shown in Fig. 5. Therefore, we measure 8342 magnetic field points in the enclosed 2-D area of size 24 mm × 24 mm with a Hall sensor. Hereby, we obtain a resolution of 96 × 86 inside that area. We again perform an inpainting task with a mask size of 48 × 48 pixels and an outpainting task with 16 regions of 1 × 1 pixel given. The qualitative results are shown in Fig. 10, with a generator trained on the virtual setup with a mask side length of 144 pixels for inpainting, and with a generator network trained on the outpainting task with a mask size of 1 × 1 pixel. Additionally, we show the prediction of a generator, which is retrained on magnetic fields resulting from a virtual, rebuilt setup that is similar to the experimental setup in Fig. 5(a). In the new dataset, the 128 empty spots are randomly filled with hard magnets being only magnetized in the xy-plane, i.e., the z-component of their magnetization is set to 0. The inpainting results seem to agree well with the original magnetic field. The predictions of the outpainting task though show a clear visual disagreement in the lower right part of the enclosed 2-D area. However, the field prediction in that area substantially improved when retraining a generator network on magnetic fields similar to the test field. In general, such a complete retraining becomes only feasible with a large amount of data available in a new setup with different scales and magnetic sources close to the measurement area. Nevertheless, we assume that our initially trained generator can predict magnetic fields without a loss of generality. With a small amount of data available for a new experimental setup, then a fine-tuning starting from the pretrained weights of our generator neural network can already improve the performance of magnetic field prediction to an acceptable level.



(a) Inpainting (virtual) (b) Outpainting (virtual, experimental)

Fig. 10. Qualitative analysis on magnetic field prediction in the experimental 2-D setup with a generator network that was trained on the virtual setup. For outpainting, the prediction of a generator retrained on the experimental setup on the same masked input is shown as well. For each of the tasks, the masked input and the predicted magnetic field values of $\mathbf{B}_y$ are visualized.

5. Discussion

The DL approach is working well and better than other methods found in literature, when the area of missing magnetic field measurements becomes large. However, our approach has flaws when the value to predict is close to the region of given field points. As shown in Fig. 8, the MAE for standard interpolation methods becomes very low as the given closest points are used as starting points for the specific interpolation technique. On the other hand, the WGAN-GP approaches take the given field points as input to the DL architecture and over several convolutions output a prediction, which is only indirectly coupled to the available measurements. The reconstruction of the given magnetic field points is solely controlled by L_{match}, which the generator tries to minimize over several updates of its network parameters. An obvious first idea to tackle this issue would be to increase λ_{match} and hence the importance of this loss. During parameter updates, more focus will be put on matching the given original points. We trained a new

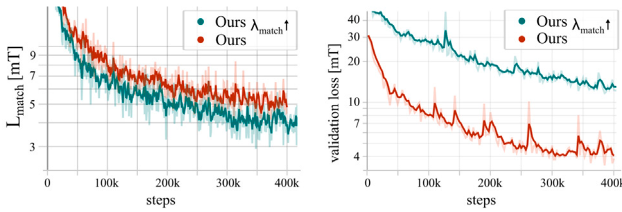


Fig. 11. Comparison of L_{match} and validation loss during training for the inpainting task with different values for λ_{match}. The red curve shows the hyperparameter set chosen throughout the study with $\lambda_{match} = 7.2$. The green curve uses a $\lambda_{match} = 1000$ to enhance to importance of this error with the intention to decrease the mismatch of generator predictions on locations with known magnetic field measurements.

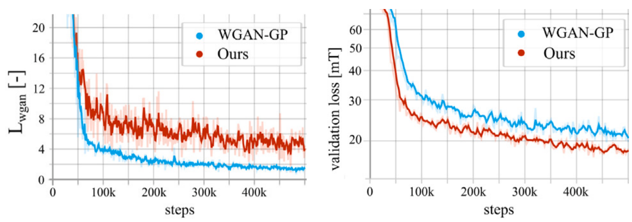


Fig. 12. Training curves for the outpainting task. The blue curve shows the progress of L_{wgan} and the validation loss with the standard WGAN-GP method. The red curve is the evolution during training, when enabling physical losses with our method.

generator network with an updated hyperparameter set. As visualized in Fig. 11(a), the green curve has indeed a lower, improved L_{match}, indicating that the prediction matches the masked input in the given areas better. But when looking at the overall validation loss in Fig. 11(b), one can see that the calculated reconstruction loss on samples unseen during training is larger throughout the training. This can be explained with an increase of the other losses, L_{div} and L_{curl}, which leads to a less physical model and worse performance on the magnetic field prediction in unknown areas. In general, a more elaborate hyperparameter search could substantially enhance the performance of our method. Due to limited available calculation time, we performed all our experiments with the set of hyperparameters stated in Section 3 without further tuning. Another approach to alleviate this behavior could be a post-processing method to smooth values at the edges of the final result or to combine it with a spline-based method close to given field points. For instance, one can predict values close to given measurements with cubic spline-based interpolation and at about a distance of 17 pixels from the next given measurements, the magnetic field predictions from our method can be more and more taken into account.

Another interesting point to discuss is the improvement of our physics-informed method compared to the standard WGAN-GP approach. Especially, on the outpainting task our method leads to a substantially lower MAPE. With Fig. 12, we want to emphasize the importance of introducing the physical losses into the DL architecture. The introduction of the physical losses influences the updates of the generator network parameters to be more general, which in return makes it easier for the critic to differentiate between real and generated samples, and L_{wgan} stays higher throughout the training. We recall that the critic tries to maximize the Wasserstein-1 distance between real and generated magnetic field distributions. Hence, $L_{wgan} = 0$ means that the critic cannot distinguish between real and generated samples anymore. Consequently, the generator network has fewer incentives to improve its generating process of magnetic field predictions. Eventually, this will result in a higher validation loss.

When looking at Fig. 10(b) one can see that the trained generator is performing well on large parts of $\mathbf{B}_y$ to predict, but has difficulties to anticipate the fast switching magnetic field in the lower right part of the measurement area. These emerge from the adjacent hard magnets at the border of the enclosed 2-D area as visualized in Fig. 5(a). In comparison to the virtual 3-D setup, the gap is smaller here and hence the magnetic field produced in the lower right area is not part of the magnetic field distribution the generator was trained on. A remedy for that could be either to include such magnetic field data in the training data in order to make the generator predictions even more general or to completely retrain on the physical 2-D setup. A further feature to implement could be an additional input parameter to indicate a specific condition, e.g., the number magnets at the border of the enclosed 2-D area or other geometrical implications such as the gap between magnets and measurement locations.

6. Conclusion

With our novel method, we are able to perform magnetic field prediction better than current state-of-the-art methods on inpainting tasks, where large parts of the magnetic field measurement relative to the overall measurement area are missing. Moreover, our physics-informed, learning-based method produces the best results when comparing it to other state-of-the-methods on an outpainting task with only point measurements (1 × 1 pixel) available. When regions (16 × 16 pixels) of measurements are given, then Gaussian processes outperform our method, however, with the inference time of magnetic field prediction being two orders of magnitude higher. In some time-critical applications such as the simultaneous mapping and localization performed in robotics mentioned in the introduction, our model could serve as a trade-off between accuracy and computational time. In future work, it can be interesting to make use of the fact that closed Poisson problems, e.g., the inpainting task in Section 4.1, can be solved from the boundary values around the missing field information. Hence, the generator neural network could be trained to predict missing field measurements from only these values in the input layer.

CRediT authorship contribution statement

Stefan Pollok: Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Drafting the manuscript, Revising the manuscript critically for important intellectual content. **Nataniel Olden-Jørgensen:** Acquisition of data, Analysis and/or interpretation of data, Revising the manuscript critically for important intellectual content. **Peter Stanley Jørgensen:** Conception and design of study, Revising the manuscript critically for important intellectual content. **Rasmus Bjørk:** Conception and design of study, Analysis and/or interpretation of data, Revising the manuscript critically for important intellectual content.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Our code, pretrained models, and examples are available at: https://github.com/spollok/magfield-prediction.

Acknowledgments

We want to thank our colleague Sina Jafarzadeh for performing the physical measurements in our laboratory manually with a Hall sensor. Every setup variation or change in resolution entailed another time-consuming measurement process. We wish to acknowledge the Villum Foundation in the Villum Synergy Programme, Denmark project entitled "Physics-aware machine learning", no. 50091, for partly sponsoring this work. All authors approved the version of the manuscript to be published.

References

[1] R. Bjørk, E.B. Poulsen, K.K. Nielsen, A.R. Insinga, MagTense: A micromagnetic framework using the analytical demagnetization tensor, J. Magn. Magn. Mater. 535 (2021) 168057.

[2] COMSOL, COMSOL Inc., 2020, [Online]. Available: http://www.comsol.com/products/multiphysics/.

[3] A. Solin, et al., Modeling and interpolation of the ambient magnetic field by Gaussian processes, IEEE Trans. Robot. 34 (4) (2018) 1112–1127.

[4] E. Le Grand, S. Thrun, 3-axis magnetic field mapping and fusion for indoor localization, in: Proc. IEEE Int. Conf. Multisens. Fusion Integr. Intell. Syst., MFI, 2012, pp. 358–364.

[5] F. Mackay, R. Marchand, K. Kabin, Divergence-free magnetic field interpolation and charged particle trajectory integration, J. Geophys. Res. 111 (A6) (2006).

[6] J.C. Bernauer, et al., Measurement and tricubic interpolation of the magnetic field for the OLYMPUS experiment, Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip. 823 (2016) 9–14.

[7] S. Maus, et al., EMAG2: A 2–arc min resolution Earth Magnetic Anomaly Grid compiled from satellite, airborne, and marine magnetic measurements, Geochem. Geophys. Geosyst. 10 (8) (2009).

[8] S.A. McLay, C.D. Beggan, Interpolation of externally-caused magnetic fields over large sparse arrays using Spherical Elementary Current Systems, Ann. Geophys. 28 (2010) 1795–1805.

[9] X. Sun, et al., A new method for polar field interpolation, Sol. Phys. 270 (2011) 9–22.

[10] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Netw. 2 (5) (1989) 359–366.

[11] J. Devlin, et al., BERT: Pre-training of deep bidirectional transformers for language understanding, 2019, in: Proc. Conf. North Am. Chap. Assoc. Comput. Linguist.: Human Lang. Technol. (NAACL-HLT), Vol. 1, 2019, pp. 4171–4186.

[12] D. Amodei, et al., Deep speech 2: End-to-end speech recognition in english and mandarin, in: Int. Conf. Mach. Learn. (PMLR), Vol. 48, 2016, pp. 173–182.

[13] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: Proc. Adv. Neural Inf. Process. Syst., Vol. 25, 2012.

[14] R. Gómez-Bombarelli, et al., Automatic chemical design using a data-driven continuous representation of molecules, ACS Cent. Sci. 4 (2) (2018) 268–276.

[15] J. Jumper, et al., Highly accurate protein structure prediction with AlphaFold, Nature 596 (7873) (2021) 583–589.

[16] A. Khan, V. Ghorbanian, D. Lowther, Deep learning for magnetic field estimation, IEEE Trans. Magn. 55 (6) (2019) 7401305.

[17] S. Pollok, R. Bjørk, P.S. Jørgensen, Inverse design of magnetic fields using deep learning, IEEE Trans. Magn. 57 (7) (2021) 2101604.

[18] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707.

[19] A. Kovacs, et al., Magnetostatics and micromagnetics with physics informed neural networks, J. Magn. Magn. Mater. 548 (2022) 168951.

[20] I.J. Goodfellow, et al., Generative adversarial nets, in: Proc. Adv. Neural Inf. Process. Syst., Vol. 27, 2014.

[21] L. Yang, D. Zhang, G.E. Karniadakis, Physics-informed generative adversarial networks for stochastic differential equations, SIAM J. Sci. Comput. 42 (1) (2020) A292–A317.

[22] M. Mirza, S. Osindero, Conditional generative adversarial nets, 2014, arXiv:1411.1784. [Online]. Available: http://arxiv.org/abs/1411.1784.

[23] Y. Song, et al., Score-based generative modeling through stochastic differential equations, in: Int. Conf. Learn. Repr., ICLR, 2021.

[24] A. van den Oord, et al., Conditional image generation with pixelCNN decoders, in: Proc. Adv. Neural Inf. Process. Syst., Vol. 29, 2016.

[25] J. Yu, et al., Generative image inpainting with contextual attention, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, 2018.

[26] Q. Zheng, L. Zeng, G.E. Karniadakis, Physics-informed semantic inpainting: Application to geostatistical modeling, J. Comput. Phys. 419 (2020) 109676.

[27] I. Gulrajani, et al., Improved training of wasserstein GANs, in: Proc. Adv. Neural Inf. Process. Syst., Vol. 30, 2017.

[28] C. Villani, Optimal Transport: Old and New, Springer, Berlin, Heidelberg, 2009.

[29] Y. Rubner, C. Tomasi, L.J. Guibas, The earth mover's distance as a metric for image retrieval, Int. J. Comput. Vis. 40 (2) (2000) 99–121.

[30] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, 2016, pp. 770–778.

[31] S. Iizuka, E. Simo-Serra, H. Ishikawa, Globally and locally consistent image completion, ACM Trans. Graph. 36 (4) (2017) 1–14.

[32] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv:1412.6980. [Online]. Available: http://arxiv.org/abs/1412.6980.

[33] P. Virtanen, et al., SciPy 1.0: Fundamental algorithms for scientific computing in python, Nature Methods 17 (2020) 261–272.

[34] S.B. Damelin, N.S. Hoang, On surface completion and image inpainting by biharmonic functions: Numerical aspects, Int. J. Math. Sci. 2018 (2018) 3950312.

[35] F. Pedregosa, et al., Scikit-learn: Machine learning in python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[36] C.E. Rasmussen, Gaussian Processes in Machine Learning, Summer School on Machine Learning, Springer, Berlin, Heidelberg, 2003, pp. 63–71.

[37] G. Tóth, B. van der Holst, Z. Huang, Obtaining potential field solutions with spherical harmonics and finite differences, Astrophys. J. 732 (2) (2011) 102.