



OPEN CNN-based classifier for automated identification of magnetic states in spin dynamics simulations

Amal Aldarawsheh¹✉, Ahmed Alia^{2,3}✉ & Stefan Blügel¹

The identification and classification of different magnetic states are essential for understanding the complex behavior of magnetic systems. Traditional approaches that rely on handcrafted features or manual inspection often fall short, particularly when dealing with subtle or topologically complex spin textures. In this study, we present an automated deep learning model that employs an EfficientNetV1B0 Convolutional Neural Network to classify nine distinct magnetic states, including both ferromagnetic (FM) and antiferromagnetic (AFM) spin textures such as AFM skyrmions and AFM stripe domains. The spin configurations are generated through atomistic spin dynamics simulations using the *Spirit* code, then visualized with VFRending to produce RGB images, which serve as inputs to the classification model. To train and evaluate the model, we created a new dataset of manually labeled RGB images. Evaluation results show that the proposed model achieves an accuracy and F1-score of 99%, outperforming the other deep learning baselines evaluated in this study.

The application of machine learning and deep learning in condensed matter physics has opened new pathways for understanding and engineering complex physical systems. These techniques are now widely used to represent quantum states¹, discover phase transitions^{2–4}, and classify magnetic phases across vast parameter spaces^{5–10}. A domain where these methods have had significant impact is the study of topological magnetic textures, such as magnetic skyrmions. These nanoscale, swirling spin structures are stabilized by antisymmetric Dzyaloshinskii–Moriya interactions (DMI) and are topologically protected against small perturbations^{11–14}. Their compact size, high stability, and efficient current-driven mobility make skyrmions highly promising for applications in spintronics, high-density memory, and neuromorphic computing^{11,15–23}. In addition to skyrmions, a rich variety of emergent magnetic states, including stripe domains, in-plane vortices, and Néel-type structures, have drawn attention for their technological relevance and for enriching our understanding of magnetic phase behavior in low-dimensional systems^{24–31}. Classifying magnetic states from raw simulation data or experimental images remains a challenging task, particularly as simulations continue to generate increasingly complex spin textures and cover larger parameter spaces. This growing complexity underscores the need for automated classification methods capable of handling large-scale datasets and subtle variations in spin configurations.

Recently, Convolutional Neural Networks (CNNs) have demonstrated strong performance in image-based classification tasks. A key advantage of CNNs is their ability to automatically learn meaningful features from labeled images without the need for manual feature engineering³². EfficientNet^{33,34}, ResNet³⁵, and MobileNet^{36,37} are popular families of CNN architectures. Among these, EfficientNetV1B0 stands out as a simple yet effective model, achieving high classification performance with fewer parameters and lower computational cost^{33,38,39}. These advancements in deep learning have motivated researchers to explore the use of CNNs for extracting physical insights from spin texture images generated through simulations or experiments.

Recent literature has demonstrated that treating spin configurations as image data enables effective training of CNNs, improving classification performance and facilitating comparison with image-based experimental measurements^{2,40}. CNNs have been applied to classify complex spin textures like ferromagnetic (FM) spirals, bimerons, skyrmion crystals, and skyrmion gases with high accuracy^{41–44}, and have also been used to extract physical parameters such as chirality, magnetization, and Hamiltonian coefficients from spin configurations^{40,43}. For example, Cimen et al.⁴⁴ designed a customized Skyr-Net network to classify seven FM micromagnetic spin textures from MuMax3 simulations, achieving near-perfect accuracy. Iakovlev et al.¹⁰ employed basic machine-learning models to distinguish complex non-collinear magnetic states in 2D materials, reporting 91% accuracy using the K-nearest neighbor algorithm. Similarly, Albarrán et al.⁴¹ implemented a CNN to classify multiple spin

¹Peter Grünberg Institute, Forschungszentrum Jülich and JARA, 52425 Jülich, Germany. ²Institute for Advanced Simulation (IAS-7), Forschungszentrum Jülich, 52425 Jülich, Germany. ³Faculty of Information Technology and Artificial Intelligence, An-Najah National University, Nablus, Palestine. ✉email: a.aldarawsheh@fz-juelich.de; a.alia@fz-juelich.de

phases, such as spirals, bimerons, skyrmion crystals, and paramagnetic states based on Monte Carlo simulations, reaching 97.5% accuracy. Furthermore, Salcedo-Gallo et al.⁴⁵ developed a CNN to classify FM, stripe, bimeron-skyrmion, and skyrmion phases from Monte Carlo simulations. Their model achieved high accuracy and was used to construct phase diagrams by varying magnetic parameters.

By contrast, machine-learning studies involving antiferromagnetism remain more limited in this image-based classification context and have mainly addressed related but distinct tasks, such as classification of magnetic order from electronic-structure descriptors, inference of magnetic structures from atomic coordinates, or prediction of the conditions for AFM skyrmion formation, rather than image-based multiclass classification of AFM spin textures themselves^{46–48}. Although some image-based studies have included an AFM ordered state as one class among several magnetic structures^{49,50}, to the best of our knowledge, previous image-based automatic classification studies have not explicitly treated AFM skyrmions and AFM stripe domains as separate target classes within a unified multiclass framework.

This gap is important because AFM textures are of considerable interest due to their faster spin dynamics, absence of stray fields, and potential for high-density, low-power spintronic devices^{51–55}. Nevertheless, most existing studies focus mainly on FM systems and a narrow set of phases, typically uniform FM states, spirals, and skyrmions.

Moreover, a key limitation of many previous efforts lies in their use of low-dimensional or simplified representations of spin textures, such as scalar observables or *z*-projected components^{10,43,45}, which obscure critical features encoded in the full three-dimensional spin structure. These limitations underscore the need for models that are trained on physically consistent, atomistic data capable of capturing the full diversity of magnetic textures.

In this study, we present an automated CNN-based model for classifying visualized spin-configuration files into nine distinct magnetic states. These configurations are generated through atomistic spin dynamics simulations and subsequently visualized as RGB images using the VFRendring tool⁵⁶. Figure 1 illustrates one representative example for each magnetic state. For the classification task, we adapt and train the EfficientNetV1B0 architecture. The original network is integrated with a fully connected layer followed by a Softmax activation function with nine output neurons, forming the final classification layer. The adapted architecture is then trained from scratch on a newly labeled image dataset, where each image represents a spin configuration.

Compared with many earlier CNN-based studies^{10,41–45}, our proposed model incorporates both FM and AFM spin textures, including in-plane and out-of-plane configurations. In particular, it explicitly includes AFM skyrmions and AFM stripe domains as dedicated classes within a unified multiclass CNN framework trained on atomistic spin-texture images. In addition, the present study considers nine magnetic states across different lattice types using atomistic spin-dynamics data, thereby broadening prior CNN-based magnetic-state classification studies to a nine-class setting.

Model and methods

In this section, we first present the architecture of the classification model, which is designed to categorize input RGB images into nine magnetic states (see Fig. 1). This is followed by a description of the preparation of the labeled image dataset required for training and evaluating the model. Further details are provided in the following subsections.

Adapted EfficientNetV1B0-based classification model

CNNs are a class of deep learning inspired by the organization of the animal visual cortex and are widely used in vision-related applications, including image classification⁵⁷. One of their main advantages is the ability to automatically and adaptively learn relevant features from data without requiring handcrafted feature

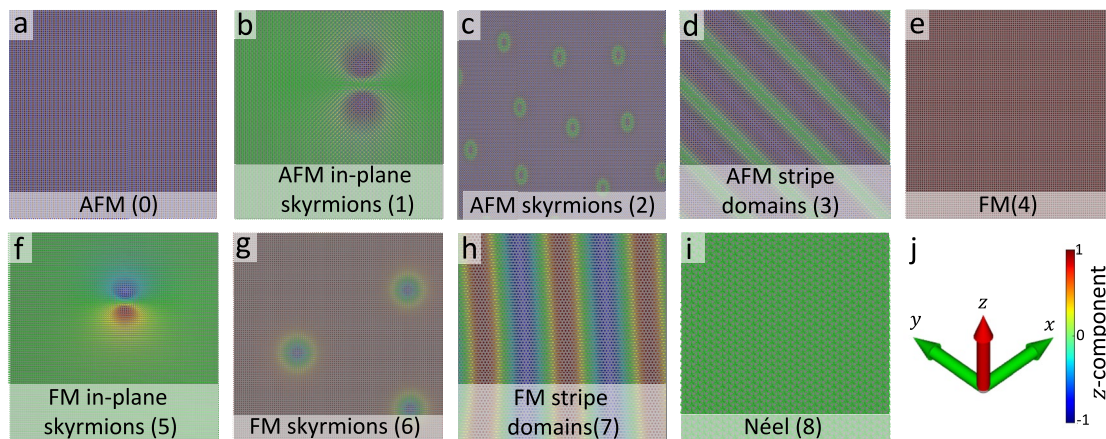


Fig. 1. Sample snapshots from the dataset. (a) AFM (0), (b) AFM in-plane skyrmions (1), (c) AFM skyrmions (2), (d) AFM stripe domains (3), (e) FM (4), (f) FM in-plane skyrmions (5), (g) FM skyrmions (6), (h) FM stripe domains (7), and (i) Néel state (8), (j) visualization of the color code used.

engineering³². A typical CNN architecture consists of repeated convolution and pooling layers for feature extraction, followed by one or more fully connected layers that map the extracted features to output neurons corresponding to classification categories⁵⁸.

The design of the CNN architecture plays a crucial role in improving model performance. Several efficient and high-performing architectures have been developed, such as InceptionResNetV2⁵⁹, DenseNet121⁶⁰, MobileNet³⁶, MobileNetV2³⁷, MobileNetV3Small⁶¹, ResNet50³⁵, ResNet101³⁵, Xception⁶², and EfficientNetV1B0³³. This final architecture stands out for its simplicity and efficiency, delivering high performance on challenging benchmark image classification tasks^{38,39}. It achieves these results with significantly fewer parameters than comparable models and reduced training costs. Additionally, the experiments presented in this article (see “[Experiments and results](#)” section) demonstrate that EfficientNetV1B0 achieves the highest accuracy compared to other popular CNN architectures in the task of magnetic state classification. As a result, our model is built based on the EfficientNetV1B0 CNN architecture.

As illustrated in Fig. 2, the feature extraction component of EfficientNetV1B0 is used to capture informative representations from each magnetic state image, while the classification part is adapted to support nine-class classification. For feature extraction (Fig. 2, top box), EfficientNetV1B0 begins with a standard 3×3 convolutional operation that processes the input magnetic state image of size $224 \times 224 \times 3$. This is followed by 16 Mobile Inverted Bottleneck Convolution (MBConv) blocks to extract the deep features. MBConv blocks form the core of the architecture, and include one MBConv1 block with a 3×3 kernel, six MBConv6 blocks with 3×3 kernels, and nine MBConv6 blocks with 5×5 kernels. Figure 3 shows the architecture of MBConv. Each MBConv block consists of several operations: it starts with a pointwise 1×1 convolution that expands the number of channels, followed by a depthwise convolution (3×3 or 5×5) that applies a spatial filter to each channel independently, thereby reducing the number of learnable parameters. Batch normalization and the Swish activation function⁶³ are applied after each convolution to stabilize training and promote non-linearity.

A key innovation in MBConv blocks is the integration of the Squeeze-and-Excitation module⁶⁴, which enhances the network’s representational capacity by learning dynamic channel-wise weights. This module first applies global average pooling to compress spatial information into a channel descriptor, followed by two fully connected layers using Swish and Sigmoid activations to compute attention weights for each channel. These learned weights are multiplied by the original feature maps to emphasize the most informative features. Finally, a second 1×1 pointwise convolution reduces the feature depth back to the original dimension, and a residual connection is added to preserve gradient flow and mitigate vanishing gradient issues during training. The main difference between MBConv6 and MBConv1 is the depth of the block and the number of operations performed in each block; MBConv6 is six times that of MBConv1. Note that MBConv6, 5×5 performs the identical operations as MBConv6, 3×3 , but MBConv6, 5×5 applies a kernel size of 5×5 , while MBConv6, 3×3 uses a kernel size of 3×3 .

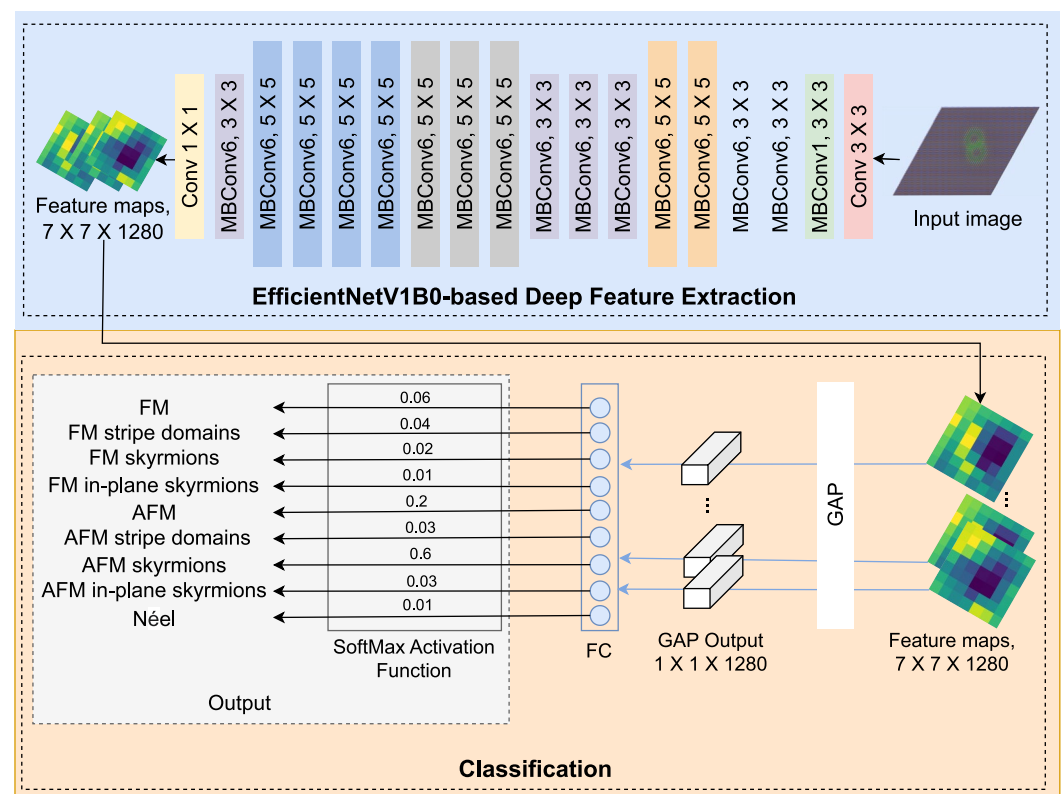


Fig. 2. The architecture of the proposed model.

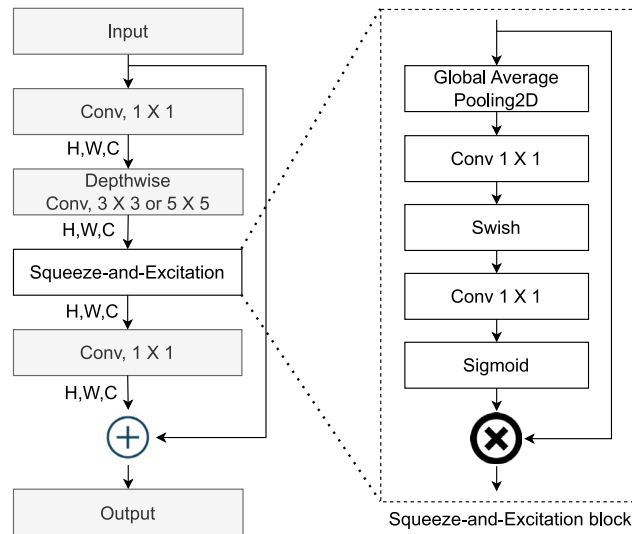


Fig. 3. The architecture of MBConv block.

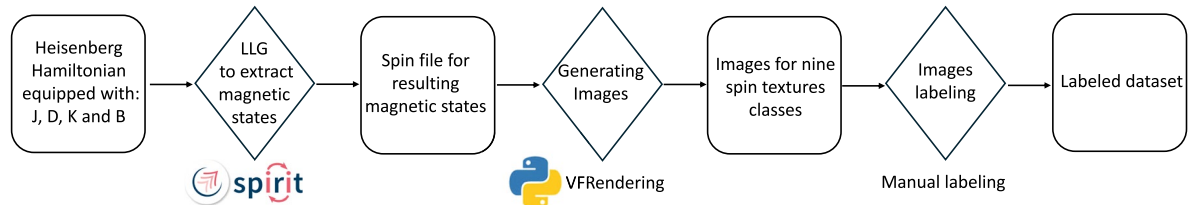


Fig. 4. Dataset preparation diagram.

This model supports nine-class classification by employing a 1×1 convolutional layer, followed by 2D global average pooling, a fully connected layer with nine output neurons, and a Softmax activation function (Fig. 2, bottom box). To enable the model to classify magnetic states into nine distinct classes, the adapted EfficientNetV1B0 CNN architecture must be trained using a labeled dataset consisting of RGB images corresponding to the nine magnetic state categories. The next section describes the preparation of this required dataset.

Dataset preparation

Figure 4 shows the main steps for creating the labeled dataset used to train and evaluate the adapted EfficientNetV1B0 model, as well as the other models included in the evaluation. To generate the magnetic states that represent the samples in this dataset, we employed a classical atomistic spin model defined by the two-dimensional Heisenberg Hamiltonian with the inclusion of Heisenberg exchange interactions (J), Dzyaloshinskii–Moriya interaction (DMI), uniaxial anisotropy K , and Zeeman coupling:

$$\mathcal{H} = - \sum_{\langle ij \rangle} J_{ij} \mathbf{n}_i \cdot \mathbf{n}_j - \sum_{\langle ij \rangle} D_{ij} \cdot (\mathbf{n}_i \times \mathbf{n}_j) - K \sum_i (n_i^z)^2 - \sum_i \mathbf{B} \cdot \mathbf{n}_i, \quad (1)$$

where i and j are site indices carrying each magnetic moment. \mathbf{n} is a unit vector of the magnetic moment. J_{ij} is the Heisenberg exchange coupling strength, being < 0 for AFM interaction, between an atom on site i and another atom on site j . A similar notation is adopted for the DMI vector D and the magnetic anisotropy energy K . The latter favors the out-of-plane orientation of the magnetization if being > 0 , or in-plane orientation if being < 0 . Finally \mathbf{B} is the external magnetic field. The Heisenberg exchange interaction values were adjusted to model both FM and AFM systems. In the FM case, we included the first nearest neighbors with FM coupling, while in the AFM case, we included up to third-nearest-neighbor interactions, particularly on triangular lattice, following our earlier findings on the emergence of intrinsic AFM skyrmions^{54,55}. Moreover, different DMI orientations were employed to stabilize various types of spin textures, including Néel-type, Bloch-type, and antiskyrmion configurations. The sign of the anisotropy term K was varied to support both in-plane and out-of-plane spin alignments. Additionally, a magnetic field along the z -direction was applied in selected FM simulations to enhance the stability of skyrmions. To avoid the nonuniversal effects of boundary conditions, we used a supercell on the 200×200 spin lattice containing 40,000 sites with periodic boundary conditions. Simulations were

performed using the *Spirit* atomistic spin dynamics package⁶⁵, which integrates the Landau–Lifshitz–Gilbert (LLG) equation to evolve spin configurations toward their ground states. We considered four distinct lattice geometries; square, triangular, rhombic, and rectangular, to ensure broad coverage of structural symmetries and magnetic interaction scenarios. The Hamiltonian parameters were systematically varied over broad intervals to stabilize the nine targeted magnetic states across the considered lattice geometries (Fig. 1). Representative parameter ranges for each magnetic state and lattice type are provided in Supplementary Table S1.

Following convergence, the underlying spin configurations generated on 200×200 lattices were visualized using the VFRendering script⁵⁶, which converts each spin-configuration file into an RGB image. These images were then resized to 224×224 pixels to match the CNN input size. In our RGB images, the arrow color encodes the out-of-plane (z) component of the spin, ranging from red (up) to blue (down), as shown in Fig. 1j, while the arrow orientation reflects the in-plane (x,y) direction, thereby preserving the full 3D spin vector information. This full-vector representation is particularly important for the in-plane cases of FM, AFM, and Néel states, for which the z -projection alone provides only limited visual contrast, whereas the arrow orientation preserves the in-plane spin direction pattern needed to distinguish these states. Supplementary Fig. S1 and Supplementary Table S2 compare the full-arrow and z -projection only representations and provide the corresponding quantitative similarity analysis.

To ensure consistency across the dataset and to avoid uncontrolled visual variation in the CNN inputs, the rendering parameters were kept fixed throughout image generation, including the colormap, arrow representation, and background settings.

After image generation, the images were manually labeled into nine distinct magnetic states mentioned earlier: AFM ground state (0), AFM in-plane skyrmions (1), AFM skyrmions (2), AFM stripe domains (3), FM ground state (4), FM in-plane skyrmions (5), FM skyrmions (6), FM stripe domains (7), and Néel state (8). The resulting dataset consists of 6,503 RGB images distributed into nine classes representing distinct magnetic states.

To create holdout data for model training and evaluation, the full labeled dataset was randomly divided class-wise into three subsets: 70% for training, 15% for validation, and 15% for testing. This split ratio is widely adopted in the deep learning community and provides a balanced approach for training and evaluating model performance⁶⁶. Table 1 presents the number of samples in each magnetic state class across the training, validation, and test sets.

Evaluation metrics

To evaluate the performance of our proposed model, we employed macro-accuracy and macro F1-score metrics that are well-suited for multi-class problems. The following provides a detailed explanation of these metrics:

1. *Macro-accuracy*, calculated by computing the accuracy independently for each class and then averaging the results. This ensures equal contribution from all classes, regardless of class imbalance. It is particularly appropriate in our case since some magnetic states are underrepresented in the dataset (e.g., Néel state), and overall accuracy alone may mask poor performance on minority classes.
2. *F1-score* is the harmonic mean of precision and recall:

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

In other words, F1-score provides a balanced measure of classification performance by incorporating precision and recall. Precision is defined as the proportion of correctly predicted instances of a class among all instances predicted as that class:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Class	Training	Validation	Test	Total
AFM	500	115	105	720
AFM in-plane skyrmions	627	146	128	901
AFM skyrmions	627	145	128	900
AFM stripe domains	482	110	100	692
FM	525	115	110	750
FM in-plane skyrmions	522	121	105	748
FM skyrmions	520	121	105	746
FM stripe domains	523	117	110	750
Néel	206	46	44	296
Total	4532	1036	935	6503

Table 1. Distribution of spin-texture images across the nine magnetic state classes for training, validation, and test sets.

Parameter	Value
Optimizer	Adam
Loss function	Categorical cross-entropy
Learning rate	0.001
Batch size	32
Epochs	50

Table 2. The hyperparameter values used in the training process.

Model	Trainable parameters	Macro accuracy	Precision	Recall	F1-score
EfficientNetV1B0	~ 4.02 M	99%	99%	99%	99%
MobileNet	~ 3.22 M	97%	97%	97%	97%
ResNet50	~ 23.55 M	94%	93%	94%	94%
Xception	~ 20.83 M	97%	97%	97%	97%
MobileNetV2	~ 2.24 M	66%	78%	65%	65%
DenseNet121	~ 6.96 M	97%	98%	97%	97%
ResNet101	~ 42.57 M	96%	96%	96%	96%
InceptionResNetV2	~ 54.29 M	97%	97%	97%	97%
MobileNetV3Small	~ 0.93 M	97%	97%	97%	97%

Table 3. Comparison of the proposed EfficientNetV1B0-based model with eight widely used CNN architectures. Reported metrics include the total number of trainable parameters, macro accuracy, precision, recall, and F1-score for each model.

and Recall (or sensitivity) is defined as the proportion of correctly predicted instances of a class among all actual instances of that class:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}.$$

In our analysis, we use the *macro F1-score*, which averages the F1-scores of all classes equally, providing a comprehensive view of the model's performance across the nine magnetic states.

In summary, these metrics were chosen to ensure that our model performs consistently across all classes, and not just those that dominate the dataset.

Experiments and results

This section presents the implementation details and training parameter settings. It also discusses the results of various experiments conducted to evaluate the performance of the proposed classifier.

Implementation details and hyperparameters

All experiments and implementations in this study were conducted on Google Colaboratory Pro, utilizing a 15 GB NVIDIA GPU and 12.7 GB of system RAM, utilizing Python along with Keras and TensorFlow libraries. Table 2 shows hyperparameters used during the training process. The default values commonly used for CNN training on ImageNet in Keras were adopted for the learning rate and batch size. Additional parameters were fine-tuned through experimentation to achieve optimal performance with magnetic states classification. The model was trained for 50 epochs with early stopping after 10 epochs being enabled to avoid overfitting.

Experimental results

Here, a series of experiments were conducted to evaluate the performance of the proposed classification model. The model was trained on the training and validation sets, and its performance was assessed on the test set using the macro accuracy, recall, precision and macro F1-score metrics. For comparative analysis, the proposed EfficientNetV1B0-based model was compared with eight widely used CNN architectures: InceptionResNetV2⁵⁹, DenseNet121⁶⁰, MobileNet³⁶, MobileNetV2³⁷, MobileNetV3Small⁶¹, ResNet50³⁵, ResNet101³⁵, and Xception⁶². Each architecture was adapted to the nine-class classification task using the same classification head. To ensure a fair comparison, all models were implemented using the same hardware and software environments, hyperparameters (Table 2), evaluation metrics, and dataset (Table 1).

As shown in Table 3 and in Fig. 5, the EfficientNetV1B0-based model performed well in classifying the nine magnetic states in our dataset. It achieved a macro accuracy, precision, recall, and F1-score of 99%, indicating high classification performance on the held-out test set spanning the nine magnetic states considered in this study. As illustrated by the confusion matrix in Fig. 5, the model correctly predicted nearly all samples across

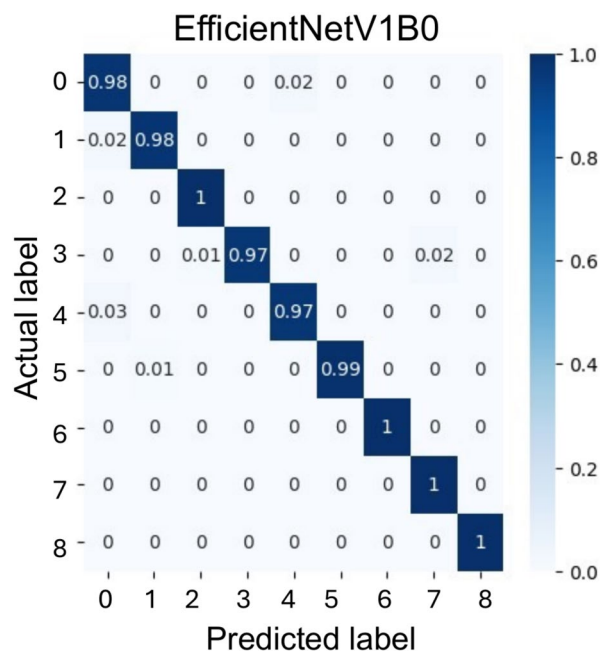


Fig. 5. Confusion matrix of the EfficientNetV1B0-based classification model. The class labels 0–8 correspond to AFM, AFM in-plane skyrmions, AFM skyrmions, AFM stripe domains, FM, FM in-plane skyrmions, FM skyrmions, FM stripe domains, and the Néel state, respectively.

Class	Samples	Recall (%)	95% Wilson confidence interval (%)
AFM	105	98	93.3–99.5
AFM in-plane skyrmions	128	98	93.3–99.2
AFM skyrmions	128	100	97.1–100
AFM stripe domains	100	97	91.5–99
FM	110	97	92.3–99.1
FM in-plane skyrmions	105	99	94.8–99.8
FM skyrmions	105	100	96.5–100
FM stripe domains	110	100	96.6–100
Néel	44	100	92–100

Table 4. Class-wise recall (%) of the EfficientNetV1B0 model on the test set, with 95% Wilson confidence intervals.

classes, including challenging categories such as AFM and FM skyrmions, as well as their in-plane variants. Misclassifications were rare and primarily occurred between states with subtle differences, such as in-plane AFM and FM ground states or stripe domains.

To further assess the robustness of the reported performance, we considered the effect of class imbalance in the dataset. The class distribution is not uniform, with the Néel class representing the smallest category (Table 1). We therefore examined the confusion matrix in Fig. 5 together with the class-wise recall values and their 95% Wilson confidence intervals⁶⁷ shown in Table 4. The EfficientNetV1B0 model showed consistently high performance across all classes, with class-wise recall ranging from 97% to 100%. In particular, the Néel class achieved 100% recall on the test set (44/44), corresponding to a 95% Wilson confidence interval of 92–100%. Thus, within the present test set, no reduction in class-wise recall was observed for the minority class.

Additionally, Fig. 6 provides qualitative insight into the features learned by the EfficientNetV1B0-based CNN classifier. For the representative samples, the Grad-CAM⁶⁸ responses are concentrated on signal-bearing regions of the spin configurations and are not dominated by empty background areas or image boundaries. Since the rendered dataset was generated using fixed visualization parameters, this qualitative analysis suggests that the classifier is primarily responding to task-relevant spatial features rather than to obvious artifacts of the image-generation pipeline.

The comparison with other CNN architectures further shows that EfficientNetV1B0 achieved the best overall combination of accuracy, precision, recall, F1-score, and trainable-parameter count among the evaluated models, indicating that it is effective for distinguishing the visual patterns present in the rendered magnetic-texture dataset used in this study. These results are consistent with the ability of EfficientNetV1B0 to extract spatial

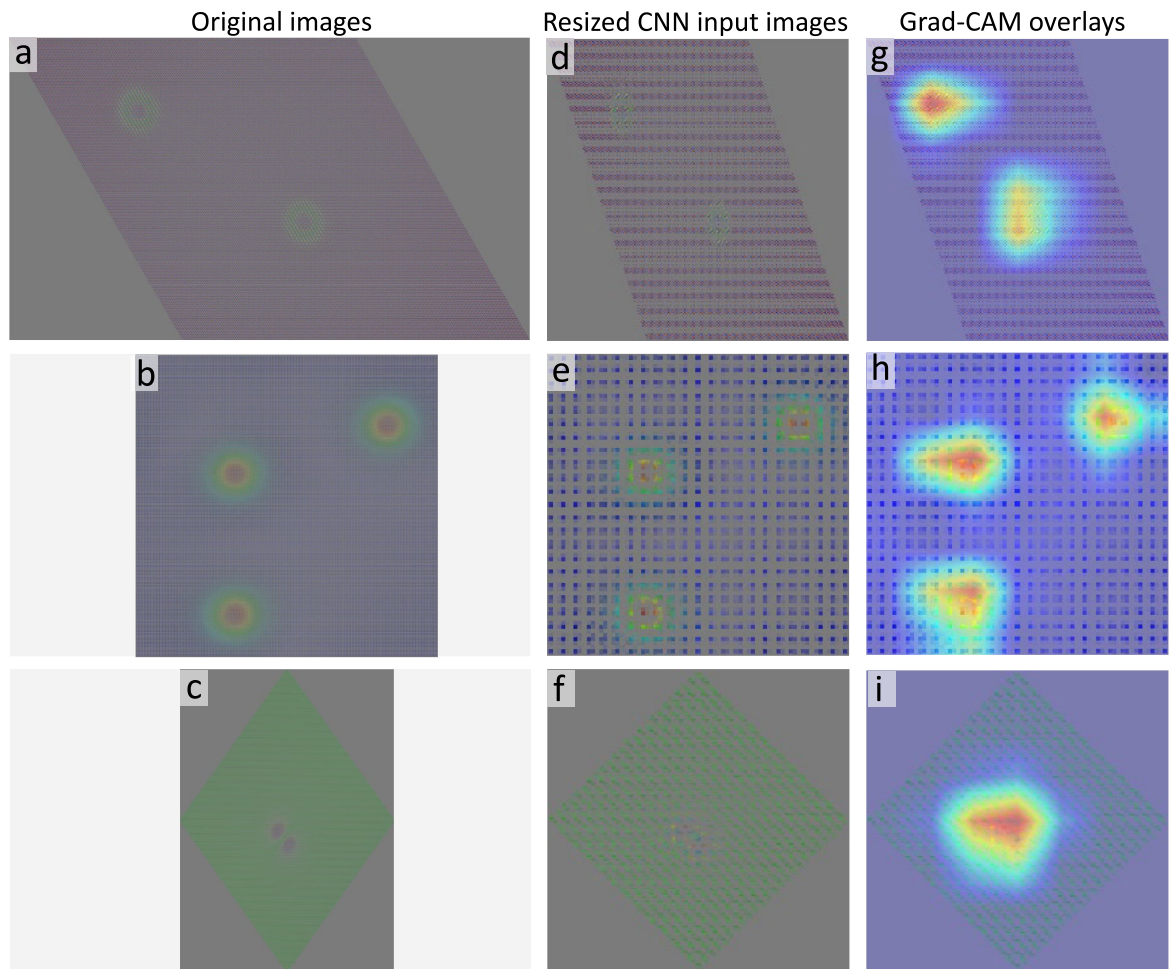


Fig. 6. Qualitative Grad-CAM visualizations for three representative samples obtained with the EfficientNetV1B0-based CNN classifier. The first column (a–c) shows the original rendered images displayed with preserved aspect ratios, the second column (d–f) shows the resized images used as CNN input (224×224 pixels), and the third column (g–i) presents the corresponding Grad-CAM overlays. Brighter regions indicate stronger model attention and are concentrated on signal-bearing regions of the spin configurations.

features efficiently from spin-texture images. While several models, such as DenseNet121, MobileNet, and InceptionResNetV2, also showed comparable performance, others like MobileNetV2 lagged behind, particularly in recall and F1-score. Figure 7 presents the confusion matrices for all models, providing a detailed breakdown of their classification strengths and weaknesses across the nine magnetic state classes.

Because the workflow starts from spin-configuration files before visualization, it is not inherently restricted to *Spirit*-generated inputs and may be extended to outputs from other simulation tools that provide OVF spin files. As a preliminary external-code check, we evaluated the trained model on 20 rendered images generated from MuMax3⁶⁹ spin-configuration files, comprising 10 FM skyrmion states, 5 FM stripe-domain states, and 5 FM states, all of which were classified correctly. This provides an initial indication of cross-code compatibility for these classes. Overall, these results support the use of CNN-based methods to automate the analysis of rendered spin-dynamics configurations, enabling more scalable, efficient, and objective interpretation of magnetic phenomena.

Conclusion, limitations and future work

In this work, we introduced an automated deep learning framework for classifying nine distinct magnetic states, including both FM and AFM spin textures, generated from atomistic spin dynamics simulations. The framework is based on an adapted EfficientNetV1B0 CNN architecture with a nine-class output layer and was trained from scratch on a newly generated labeled dataset of spin-texture images. Evaluated on data spanning multiple lattice geometries and magnetic regimes, the framework explicitly incorporates AFM skyrmions and AFM stripe domains within a unified classification model. Under the controlled conditions considered here, the model accurately classified rendered images of simulated in-plane and out-of-plane magnetic configurations, achieving 99% accuracy and macro F1-score.

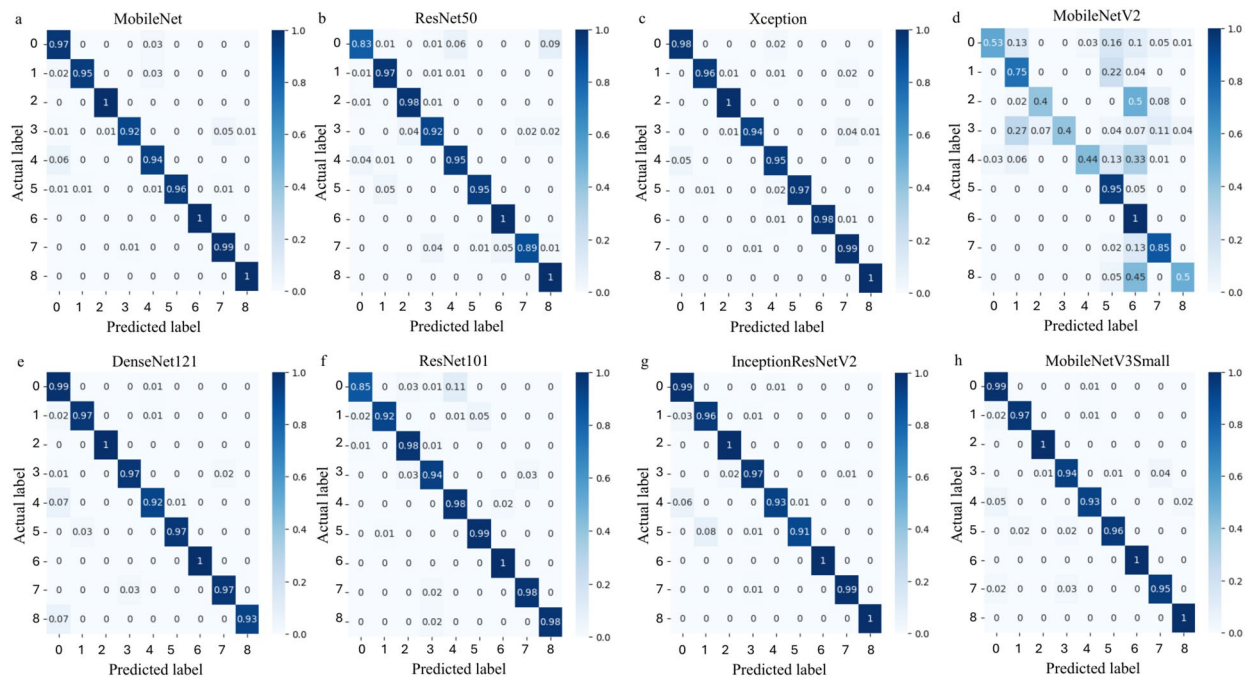


Fig. 7. Confusion matrices of the other CNN models trained on the magnetic state classification dataset. Each matrix illustrates the model's classification performance across the nine target classes: AFM (0), AFM in-plane skyrmions (1), AFM skyrmions (2), AFM stripe domains (3), FM(4), FM in-plane skyrmions (5), FM skyrmions(6), FM stripe domains(7), and the Néel state (8). Panels correspond to (a) MobileNet, (b) ResNet50, (c) Xception, (d) MobileNetV2, (e) DenseNet121, (f) ResNet101, (g) InceptionResNetV2, and (h) MobileNetV3Small. Diagonal values indicate correct predictions, while off-diagonal values indicate misclassifications.

A limitation of the present study is that it relies on synthetic, noise-free images generated from simulated spin configurations using a fixed rendering protocol. Accordingly, the reported results do not establish robustness to variations in visualization settings such as spin projection, colormap, arrow representation, or background fraction. Because the CNN learns from spatial and visual features in the rendered images, such variations may alter the image statistics and affect classification performance. In addition, many experimental techniques provide only partial magnetic information, for example predominantly in-plane or out-of-plane contrast depending on the imaging modality. Extending the present framework to experimental data will therefore require modality-specific datasets together with retraining or fine-tuning of the classifier. Furthermore, the present study considers ground-state and metastable configurations and does not address finite-temperature effects, where thermal fluctuations may lead to less well-defined spin textures and potentially affect classification performance.

Future work will therefore focus on extending the framework to experimental image data through modality-specific dataset construction and retraining, as well as to projection-limited inputs in which only partial magnetic information is available. It will also be important to evaluate robustness with respect to visualization parameters such as colormap and background fraction, and to further integrate explainable AI techniques to better interpret the model's classification decisions. Overall, the present study demonstrates that a unified CNN-based framework can accurately classify a diverse set of simulated spin textures, including both FM and AFM configurations.

Data availability

The datasets generated and analyzed during the current study, together with the trained models, are available at: <https://github.com/Amalaldarawsheh/CNN-MagSpin>.

Received: 3 March 2026; Accepted: 15 May 2026

Published online: 01 June 2026

References

- Carleo, G. & Troyer, M. Solving the quantum many-body problem with artificial neural networks. *Science* **355**, 602–606 (2017).
- Rem, B. S. et al. Identifying quantum phase transitions using artificial neural networks on experimental data. *Nat. Phys.* **15**, 917–920 (2019).
- Wang, L. Discovering phase transitions with unsupervised learning. *Phys. Rev. B* **94**, 195105 (2016).
- Van Nieuwenburg, E. P., Liu, Y.-H. & Huber, S. D. Learning phase transitions by confusion. *Nat. Phys.* **13**, 435–439 (2017).

5. Broecker, P., Carrasquilla, J., Melko, R. G. & Trebst, S. Machine learning quantum phases of matter beyond the fermion sign problem. *Sci. Rep.* **7**, 8823 (2017).
6. Ch'ng, K., Carrasquilla, J., Melko, R. G. & Khatami, E. Machine learning phases of strongly correlated fermions. *Phys. Rev. X* **7**, 031038 (2017).
7. Zhang, Y., Melko, R. G. & Kim, E.-A. Machine learning \mathbb{Z}_2 quantum spin liquids with quasiparticle statistics. *Phys. Rev. B* **96**, 245119 (2017).
8. Zhang, Y. & Kim, E.-A. Quantum loop topography for machine learning. *Phys. Rev. Lett.* **118**, 216401 (2017).
9. Carrasquilla, J. & Melko, R. G. Machine learning phases of matter. *Nat. Phys.* **13**, 431–434 (2017).
10. Iakovlev, I., Sotnikov, O. & Mazurenko, V. Supervised learning approach for recognizing magnetic skyrmion phases. *Phys. Rev. B* **98**, 174411 (2018).
11. Mühlbauer, S. et al. Skyrmion lattice in a chiral magnet. *Science* **323**, 915 (2009).
12. Romming, N. et al. Writing and deleting single magnetic skyrmions. *Science* **341**, 636 (2013).
13. Fert, A., Reyren, N. & Cros, V. Magnetic skyrmions: Advances in physics and potential applications. *Nat. Rev. Mater.* **2**, 17031 (2017).
14. Göbel, B., Mertig, I. & Tretiakov, O. A. Beyond skyrmions: Review and perspectives of alternative magnetic quasiparticles. *Phys. Rep.* (2020).
15. Sampaio, J., Cros, V., Rohart, S., Thiaville, A. & Fert, A. Nucleation, stability and current-induced motion of isolated magnetic skyrmions in nanostructures. *Nat. Nanotechnol.* **8**, 839 (2013).
16. Fert, A., Cros, V. & Sampaio, J. Skyrmions on the track. *Nat. Nanotechnol.* **8**, 152 (2013).
17. Cortés-Ortuño, D. et al. Thermal stability and topological protection of skyrmions in nanotracks. *Sci. Rep.* **7**, 4060 (2017).
18. Moreau-Luchaire, C. et al. Additive interfacial chiral interaction in multilayers for stabilization of small individual skyrmions at room temperature. *Nat. Nanotechnol.* **11**, 444–448 (2016).
19. Wiesendanger, R. Nanoscale magnetic skyrmions in metallic films and multilayers: A new twist for spintronics. *Nat. Rev. Mater.* **1**, 1–11 (2016).
20. Zhang, X., Ezawa, M. & Zhou, Y. Magnetic skyrmion logic gates: Conversion, duplication and merging of skyrmions. *Sci. Rep.* **5**, 9400 (2015).
21. Puebla, J., Kim, J., Kondou, K. & Otani, Y. Spintronic devices for energy-efficient data storage and energy harvesting. *Commun. Mater.* **1**, 24 (2020).
22. Song, K. M. et al. Skyrmion-based artificial synapses for neuromorphic computing. *Nat. Electron.* **3**, 148–155 (2020).
23. Yokouchi, T. et al. Pattern recognition with neuromorphic computing using magnetic field-induced dynamics of skyrmions. *Sci. Adv.* **8**, eabq5652 (2022).
24. Parkin, S. S. P., Hayashi, M. & Thomas, L. Magnetic domain-wall racetrack memory. *Science* **320**, 190–194 (2008).
25. Spethmann, J., Grünebohm, M., Wiesendanger, R., von Bergmann, K. & Kubetzka, A. Discovery and characterization of a new type of domain wall in a row-wise antiferromagnet. *Nat. Commun.* **12**, 3488 (2021).
26. Ma, C. et al. Electric field-induced creation and directional motion of domain walls and skyrmion bubbles. *Nano Lett.* **19**, 353–361 (2018).
27. Grebenchuk, S. et al. Topological spin textures in an insulating van der Waals ferromagnet. *Adv. Mater.* **36**, 2311949 (2024).
28. Göbel, B., Mook, A., Henk, J., Mertig, I. & Tretiakov, O. A. Magnetic bimerons as skyrmion analogues in in-plane magnets. *Phys. Rev. B* **99**, 060407 (2019).
29. Ezawa, M. Compact merons and skyrmions in thin chiral magnetic films. *Phys. Rev. B* **83**, 100408 (2011).
30. Capriotti, L., Trumper, A. E. & Sorella, S. Long-range Néel order in the triangular Heisenberg model. *Phys. Rev. Lett.* **82**, 3899 (1999).
31. White, S. R. & Chernyshev, A. Néel order in square and triangular lattice Heisenberg models. *Phys. Rev. Lett.* **99**, 127004 (2007).
32. Gu, J. et al. Recent advances in convolutional neural networks. *Pattern Recognit.* **77**, 354–377 (2018).
33. Tan, M. & Le, Q. EfficientNet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, 6105–6114 (PMLR, 2019).
34. Tan, M. & Le, Q. EfficientNetV2: Smaller models and faster training. In *International Conference on Machine Learning*, 10096–10106 (PMLR, 2021).
35. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778 (2016).
36. Howard, A. G. MobileNets: Efficient convolutional neural networks for mobile vision applications. [arXiv:1704.04861](https://arxiv.org/abs/1704.04861) (2017).
37. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. & Chen, L.-C. MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4510–4520 (2018).
38. Alia, A., Maree, M., Chraïbi, M. & Seyfried, A. A novel Voronoi-based convolutional neural network framework for pushing person detection in crowd videos. *Complex Intell. Syst.* **10**, 5005–5031 (2024).
39. Alia, A., Maree, M. & Chraïbi, M. A hybrid deep learning and visualization framework for pushing behavior detection in pedestrian dynamics. *Sensors* **22**, 4040 (2022).
40. Wang, D. et al. Machine learning magnetic parameters from spin configurations. *Adv. Sci.* **7**, 2000566 (2020).
41. Gómez Albarracín, F. A. & Rosales, H. D. Machine learning techniques to construct detailed phase diagrams for skyrmion systems. *Phys. Rev. B* **105**, 214423 (2022).
42. Araz, J. Y., Criado, J. C. & Spannowsky, M. Identifying magnetic antiskyrmions while they form with convolutional neural networks. *J. Magn. Magn. Mater.* **563**, 169806 (2022).
43. Feng, D., Guan, Z., Wu, X., Wu, Y. & Song, C. Classification of skyrmionic textures and extraction of Hamiltonian parameters via machine learning. *Phys. Rev. Appl.* **21**, 034009 (2024).
44. Cimen, M., Onel, A. C., Yarımbiyik, A. E., Arikian, M. & Tulum, G. Sky-net: Deep learning approach for image classification of magnetic structures. *J. Magn. Magn. Mater.* **621**, 172912 (2025).
45. Salcedo-Gallo, J., Galindo-González, C. & Restrepo-Parra, E. Deep learning approach for image classification of magnetic phases in chiral magnets. *J. Magn. Magn. Mater.* **501**, 166482 (2020).
46. Jang, Y., Kim, C. H. & Go, A. Classification of magnetic order from electronic structure by using machine learning. *Sci. Rep.* **13**, 12445 (2023).
47. Merker, H. A. Machine learning magnetism classifiers from atomic coordinates. *iScience* <https://doi.org/10.1016/j.isci.2022.105192> (2022).
48. Saini, S., Shukla, A. K., Nehete, H., Bindal, N. & Kaushik, B. K. Machine learning-based prediction of antiferromagnetic skyrmion formation. *IEEE Trans. Electron Devices* **71**, 2774–2780 (2024).
49. Hasib, F. I., Swarna, N. F. & Alam, M. A. Classification of different magnetic structures from image data using deep neural networks. In *2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, 1–6 (IEEE, 2021).
50. Bokul, S., Shukur, S. S. M. A., Ahmed, S., Bhowmick, T. K. & Alam, M. A. Classification of non-topological magnetic configurations using machine learning. In *2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, 1–5 (IEEE, 2020).
51. Gomonay, O., Jungwirth, T. & Sinova, J. High antiferromagnetic domain wall velocity induced by Néel spin-orbit torques. *Phys. Rev. Lett.* **117**, 017202 (2016).
52. Jungwirth, T., Marti, X., Wadley, P. & Wunderlich, J. Antiferromagnetic spintronics. *Nat. Nanotechnol.* **11**, 231–241 (2016).

53. Baltz, V. et al. Antiferromagnetic spintronics. *Rev. Mod. Phys.* **90**, 015005 (2018).
54. Aldarawsheh, A. et al. Emergence of zero-field non-synthetic single and interchained antiferromagnetic skyrmions in thin films. *Nat. Commun.* **13**, 7369 (2022).
55. Aldarawsheh, A., Sallermann, M., Abusaa, M. & Lounis, S. A spin model for intrinsic antiferromagnetic skyrmions on a triangular lattice. *Front. Phys.* **11**, 1175317 (2023).
56. Puerling, T. VFRendering: Python script for visualizing spin textures. <https://iffgit.fz-juelich.de/puerling/VFRendering/-/blob/6ec84d9e308d7c97e8dd9563943f11c5e9132aa7/render.py> (2023).
57. Younesi, A. et al. A comprehensive survey of convolutions in deep learning: Applications, challenges, and future trends. *IEEE Access* **12**, 41180–41218 (2024).
58. Yamashita, R., Nishio, M., Do, R. K. G. & Togashi, K. Convolutional neural networks: An overview and application in radiology. *Insights Imaging* **9**, 611–629 (2018).
59. Szegedy, C., Ioffe, S., Vanhoucke, V. & Alemi, A. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 31, no. 1 (2017).
60. Huang, G., Liu, Z., Van Der Maaten, L. & Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4700–4708 (2017).
61. Howard, A. Searching for MobileNetV3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* 1314–1324 (2019).
62. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 1251–1258 (2017).
63. Ramachandran, P., Zoph, B. & Le, Q. V. Searching for activation functions. *arXiv preprint arXiv:1710.05941* (2017).
64. Hu, J., Shen, L. & Sun, G. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7132–7141 (2018).
65. Müller, G. P. et al. Spirit: Multifunctional framework for atomistic spin simulations. *Phys. Rev. B* **99**, 224414 (2019).
66. Genc, B. & Tunc, H. Optimal training and test sets design for machine learning. *Turk. J. Electr. Eng. Comput. Sci.* **27**, 1534–1545 (2019).
67. Brown, L. D., Cai, T. T. & DasGupta, A. Interval estimation for a binomial proportion. *Stat. Sci.* **16**, 101–133 (2001).
68. Selvaraju, R. R. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision* 618–626 (2017).
69. Vansteenkiste, A. The design and verification of MuMax3. *AIP Adv.* <https://doi.org/10.1063/1.4899186> (2014).

Acknowledgements

The authors thank Nikolai Kiselev and Thorben Püring for fruitful discussions. This work was financially supported by the Deutsche Forschungsgemeinschaft (DFG) through CRC 1238 (Project C01) and the European Research Council grant 856538 (project “3D MAGIC”).

Author contributions

A.A. conceived the idea and designed the study. She generated and curated the dataset, performed data labeling, and implemented, modified, and trained the convolutional neural network models. A.A. contributed to model development, implementation, and performance evaluation. S.B. supervised the project and contributed to the conceptual development of the work. All authors discussed the results and contributed to the writing and revision of the manuscript.

Funding

Open Access funding enabled and organized by Projekt DEAL.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1038/s41598-026-54755-y>.

Correspondence and requests for materials should be addressed to A.A. or A.A.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2026