Formatted: Header

Simulation-Trained Machine Learning for Segmenting Néel-Type Skyrmions

Alec Bender, Arthur McCray

Understanding the behavior of magnetic skyrmions is fundamental to the next generation of data storage devices. Skyrmion lattices can be studied using *in situ* Lorentz Transmission Electron Microscopy (LTEM), but the images are difficult to interpret. In this work, we have developed a machine learning technique to perform instant segmentation on LTEM images of skyrmion lattices and thus extract quantitative information. We used micromagnetics software to simulate 10,000 skyrmion lattices, from which we created ground truths of skyrmion sizes and positions and further simulated corresponding LTEM images. We then trained a convolutional neural network (CNN) on these simulated LTEM images and ground truths using supervised learning. Our results demonstrate that the CNN can accurately identify skyrmion locations and extent in both simulated and experimental data, providing a technique for quantitative analysis of skyrmion lattices going forward.

Formatted: Footer

1. Introduction

Magnetic skyrmions are vortex-like spin structures observed in ferromagnetic thin films [1]. As shown Fig. 1, their magnetizations are stabilized by the Dzyaloshinskii-Moriya interaction (DMI) which arises from a lack of inversion symmetry in the material or heterostructure. The governing energy terms in skyrmionics materials, including the DMI, exchange, anisotropy, and dipolar energies, stabilize ordered skyrmion lattices that can be harnessed for future memory devices. Skyrmions also possess high stability against perturbations due to topological protection and thus must an energy barrier must be overcome to destroy them. This topological protection also allows skyrmions to exhibit high mobility under an electrical current. These properties of magnetic skyrmions make them appealing contenders for future memory uses [2].

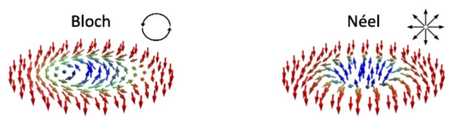

Fig. 1: Bloch and Néel-type skyrmions

Lorentz transmission electron microscopy (LTEM) is a technique for observing the magnetic and structural properties of many materials. When performing LTEM, samples are located in a field-free region of the microscope which allows viewing the unaltered magnetic domain structure of the sample. Electrons pass through the magnetic domains and are deflected by the Lorentz force, as shown in Fig. 2. In order to observe the magnetic domain walls, the microscope must be defocused, allowing domain walls to appear as bright or dark regions relative to the background. For observing Néel skyrmions and domain walls in particular, the sample must also be tilted in the microscope for the skyrmions to be observed. The skyrmions within the image appear as brighter or darker regions relative to the background as seen in Fig. 3(b). The images give data on where groups of skyrmions are located, but attaining specific quantitative information is difficult.

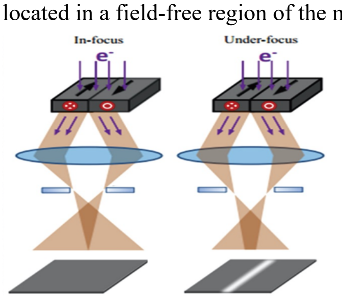

Fig. 2: Diagram of LTEM
Chinese Phys. B 27 (2018): 066802

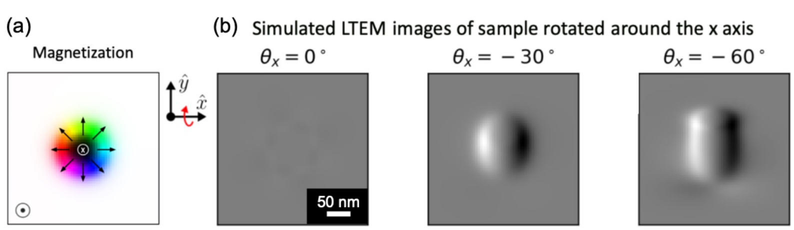

Fig. 3: (a) Skyrmion in Mumax (b) Skyrmion in simulated LTEM image at -25 μm defocus and varying tilts.

Formatted: Header

Here we present a neural network with the ability to discern close-packed skyrmions from one another and provide the ability to distinguish individual skyrmion size and shape. We performed supervised learning and trained a convolutional neural network (CNN) with simulated LTEM images and corresponding ground truths containing skyrmion sizes and locations.

2. Methods

The basic workflow for attaining the final prediction model is shown below in Fig. 4. Mumax simulates skyrmion magnetizations that are then converted into training images through PyLorentz and ground truths. The images are used for input training data for a CNN developed in PyTorch. Once fully trained, the CNN can be directly applied to experimental data.

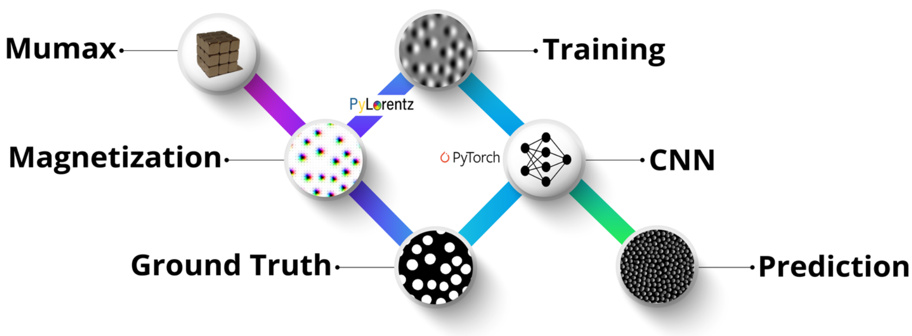

Fig. 4: Flowchart outlining the steps taken in this project, from initial micromagnetic simulations in Mumax to applying the CNN to experimental data.

A. Simulating Néel skyrmion lattice magnetizations

The Mumax micromagnetics package is used to simulate Néel skyrmion lattice magnetizations, an example of which can be seen in Fig. 3(a) [3]. The black center represents the perpendicular magnetism while the colors are the magnetic domains moving radially outward. We created a Python script to generate input files that determined the properties within the simulation. All simulations had the same field of views, cell size, and were relaxed from a random initial magnetization. The magnetic energy term parameters along with the skyrmion size and density were randomly varied. These parameters were bounded in a range that would minimize stripe domains and non-skyrmion images. By defining regions in the simulation model, certain magnetic parameters could be changed across the sample; therefore, there would be more diversity to appear more similar to experimental images. Overall, we performed 10,000 simulations using three separate Nvidia Tesla M10 GPUs, which took one week.

B. Defining the ground truth

The magnetization outputs from Mumax were then used to define ground truths about skyrmion location and size. The ground truth, shown in Fig. 5(b), is a binary image containing

Formatted: Footer

the information of whether each pixel is part of a skyrmion or background. The magnitude of a certain pixel's magnetism from one to zero is determined by the z-component of the magnetization. Then in order to provide simply a binary of one and zero, a cut-off of what is and is not a skyrmion is done by way of a triangle threshold, an automated thresholding method that tries to maximize variation across two classes [4].

C. Simulating LTEM training images

The training LTEM images, an example of which is shown in Fig 5(c), are created through PyLorentz, an open-source software suite developed for quantitative image analysis of LTEM images. We are utilizing PyLorentz's capabilities to simulate LTEM images from micromagnetic simulation output. PyLorentz takes as parameters that define sample properties and that define the microscope for which the image is simulated. Randomly generated noise is applied to the simulated images to mimic true noise that would occur during experimental imaging, shown in Fig. 5(d). Converting the magnetization images into training images and ground truths only takes a few minutes for each process.

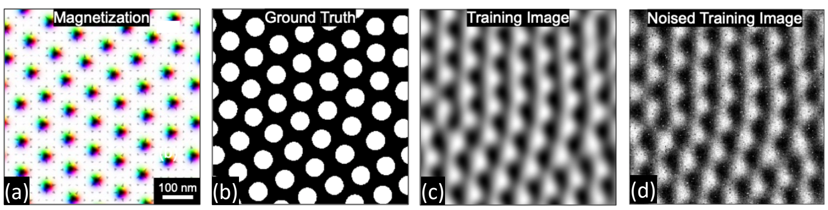

Fig. 5: Image processes

D. Machine learning with a convolutional neural network

A neural network is a computing system based on biological brains comprised of connected nodes that mimic neurons. Training the CNN consists of inputting the training data, comparing the prediction label, and receiving a loss. Improvement occurs by backpropagating through the layers and optimizing connections. We used supervised learning where the program is given the training images and the corresponding ground truths. CNNs are used frequently for image processing and computer vision. As shown in Fig. 6, CNNs consist of two basic operations: a convolutional layer where the input array is optimized to extract notable features and a pooling layer that reduces dimension and parameter to reduce computational costs. We specifically used a fully-conventional CNN based on a four-level U-Net architecture. The U-Net architecture was originally designed for medical image segmentation and lacks a fully connected layer, such that all operations are strictly local [5]. This is used for our case of segmentation, because the location of one skyrmion is not dependent on the location of another skyrmion beyond its neighbors; therefore, communication between the locations of skyrmions as a whole is unnecessary. The CNN was developed through PyTorch, an open-source machine learning framework, and took just under an hour to train over 500 epochs. After only 100 epochs, the

Formatted: Header

Formatted: Footer

Formatted: Header

CNN proved to be sufficient according to the test and training loss in Fig. 7. 400 epochs showed little to no improvement without overtraining.

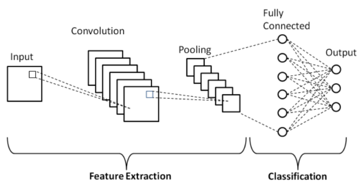

Fig. 6: Diagram of a CNN
Applied Sciences 9 (2019): 10.3390/app9214500

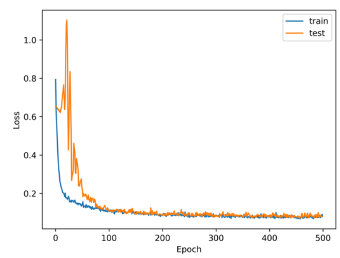

Fig. 7: CNN loss for 500 epochs

3. Results

The trained CNN can be applied to segment both experimental and simulated skyrmion images. The output of the CNN is a prediction map, where each pixel has a magnitude from one to zero that represents the CNN's confidence that it is part of a skyrmion. When applying the CNN to an image, it is capable of accurately identifying skyrmions and allows users the capability to identify individual skyrmion size and shape.

A. Simulated Data

The CNN can be used on simulated data, Fig. 8(a). The original ground truth, Fig. 9(b), and the CNN output, Fig. 8(c), appear similar to one another, demonstrating the high accuracy of the CNN. Thresholding the output, Fig. 8(d), slightly increases the radius of the output, which can be seen when overlaying it on the ground truth, Fig. 8(e), where the skyrmions in the ground truth fit within the skyrmions of the threshed output

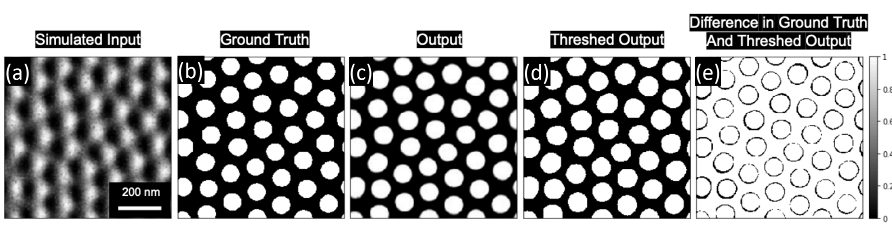

Fig. 8: Simulated results

Formatted: Footer

B. Experimental Data

The main purpose of the CNN is to classify skyrmions with experimental data, Fig. 9(a). Before the CNN can be used in this way, it needs to first be rotated to where the darker regions are on the left and the lighter regions are on the right, just as they are on the training images. This corresponds to aligning the tilt axis to the x-axis. The image is then cropped to focus on a smaller group of skyrmions to conserve GPU memory, and we ensure that the scale (nm/pixel) of the experimental image is similar to that of the simulated training. The image is then Fourier filtered, as seen in Fig. 9(b), to reduce high frequency and long-range noise and making the contrast more similar to that of the training images. These steps improve the accuracy of the CNN. Ultimately, the CNN can differentiate what is and is not a skyrmion within the experimental data. Thresholding the output and overlaying it on the filtered image allows the user to clearly see each individual skyrmion, as well as each skyrmions size, shape, and deformation, Fig. 9(c).

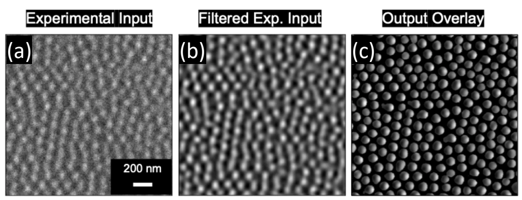

Fig. 9: Experimental results

4. Conclusion

Skyrmion lattices are of large interest right now for both fundamental science and potential computing applications, and in this work we have demonstrated a new method to analyze them. Our CNN enables immediate feedback during *in situ* experiments, making them more quantitative and providing the user insight how to improve the experiment and discover new phenomenon. We will continue to optimize the CNN to become more accurate and adapt to more general and challenging imaging conditions.

Further work will focus on applying the CNN to skyrmion nucleation and skyrmion destruction events. When skyrmions nucleate close to the Curie temperature, it can be difficult to identify whether a feature is either a stripe/maze domain or several small skyrmions in alignment. Fig. 10 shows what is likely to be a group of separate skyrmions (a) and a stripe domain (b). We cannot be fully certain what these features represent. The CNN could help distinguish between these scenarios. Skyrmion disappearance events occur during cooling, and are marked by two or more smaller skyrmions being replaced with one or more larger skyrmions. Fig. 11 shows a skyrmion lattice at 176 K (a) and the same lattice at 172 K (b). The red dots show a group of skyrmions during cooling that either merge into surrounding skyrmions, with each other, or a combination of the two. By studying the size and shape before and after these events we hope to understand how and why these events take place.

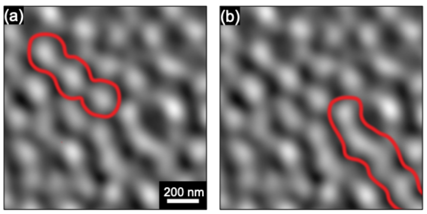

Fig. 10: Skyrmion discernment

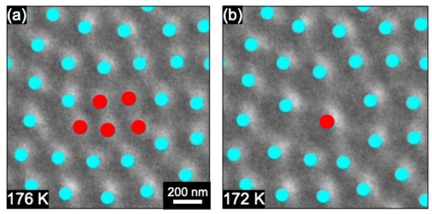

Fig. 11: Skyrmion deformation

5. References

[1] McCray, Arthur RC, et al. "Understanding complex magnetic spin textures with simulation-assisted Lorentz transmission electron microscopy." *Physical Review Applied* 15.4 (2021): 044025.

[2] Jiang, Wanjun, et al. "Skyrmions in magnetic multilayers." *Physics Reports* 704 (2017): 1-49.

[3] Vansteenkiste, Arne, et al. "The design and verification of MuMax3." *AIP advances* 4.10 (2014): 107133.

[4] Tajima, Ryosuke, and Yoichiro Kato. "Comparison of threshold algorithms for automatic image processing of rice roots using freeware ImageJ." *Field Crops Research* 121.3 (2011): 460-463.

[5] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015.