



OPEN

Efficient *few-shot* machine learning for classification of EBSD patterns

Kevin Kaufmann¹, Hobson Lane^{2,3}, Xiao Liu⁴ & Kenneth S. Vecchio^{1,4}✉

Deep learning is quickly becoming a standard approach to solving a range of materials science objectives, particularly in the field of computer vision. However, labeled datasets large enough to train neural networks from scratch can be challenging to collect. One approach to accelerating the training of deep learning models such as convolutional neural networks is the transfer of weights from models trained on unrelated image classification problems, commonly referred to as transfer learning. The powerful feature extractors learned previously can potentially be fine-tuned for a new classification problem without hindering performance. Transfer learning can also improve the results of training a model using a small amount of data, known as *few-shot* learning. Herein, we test the effectiveness of a *few-shot* transfer learning approach for the classification of electron backscatter diffraction (EBSD) pattern images to six space groups within the ($4/m\bar{3}2/m$) point group. Training history and performance metrics are compared with a model of the same architecture trained from scratch. In an effort to make this approach more explainable, visualization of filters, activation maps, and Shapley values are utilized to provide insight into the model's operations. The applicability to real-world phase identification and differentiation is demonstrated using dual phase materials that are challenging to analyze with traditional methods.

Data science-based methods to materials development and analysis have gained great popularity in recent years^{1–13}. Deep learning algorithms are of significant interest owing to their excellent performance without significant feature engineering, and the ubiquity of these methods will likely continue owing to the outperformance of systems directly designed by humans. While often difficult to assess how and why these ‘black box algorithms’ are capable of performing these tasks, these methods can provide significant value or spark new insights^{14,15}. Application of these tools to image-based tasks in materials science has proved to be useful for classification^{16–19}, segmentation^{20–22}, and other objectives^{23–25}. While deep learning provides significant opportunities for the advancement of materials science, robust application of these tools often requires much larger datasets than are typically available within the materials science community. Utilizing the knowledge deep neural networks have learned from other domains offers an opportunity to develop models in these domains, where data is sparse and further collection and labeling could be slow or tedious^{26–28}.

Convolutional neural networks (CNNs) are a class of deep learning models that have proven effective for analyzing image data²⁹. Before a CNN can be applied to a given task, it must learn to assign importance (learnable weights and biases) to various aspects of the image that maximize the network's differentiation capabilities. Two general strategies exist for training convolutional neural networks: (1) the weights can be randomly initialized, or (2) the weights can be transferred from a model pre-trained on a separate but related task, often in a nearby domain with significantly more data, and then refined for the current objective. The first approach, commonly referred to as “training from scratch”, requires a large dataset to avoid overfitting and perform robustly on new, real-world examples. The second approach, referred to as “transfer learning”, can significantly reduce the number of training examples required, accelerate the training process, and retain or exceed the performance garnered by training from scratch^{27,30–32}. The transfer learning method is motivated by the human ability to intelligently apply previously learned knowledge to solve new problems faster or with better solutions³³. Despite the potential, knowledge transfer from a given source domain is not guaranteed to improve performance in the target domain and can in fact hinder performance^{26,32}. Furthermore, one of the requirements to use this approach is that the images in the new domain must conform to the processed shape and structure determined at the outset of the previous training. For models pre-trained on ImageNet³⁴, a library of over one million images labeled with one thousand classes, the expected input is usually 299×299 pixels with 3-channels (one each for RGB). When dealing

¹Department of NanoEngineering, UC San Diego, La Jolla, CA 92093, USA. ²Tangible AI LLC, San Diego, CA 92037, USA. ³Department of Healthcare Research and Policy, UC San Diego-Extension, San Diego, CA 92037, USA. ⁴Materials Science and Engineering Program, UC San Diego, La Jolla, CA 92093, USA. ✉email: kvecchio@eng.ucsd.edu

with one-channel grayscale images, such as those typically collected from electron diffraction studies, the decision to use transfer learning necessitates the stacking of a single image into a pseudo-color image³¹.

The number of labeled images that can reasonably be collected must also be considered for the appropriate training and application of a CNN. Computer vision research has recently been motivated by children's ability to learn novel visual concepts almost effortlessly after accumulating sufficient past knowledge³⁵. In deep learning and computer vision, learning visual models of object categories has notoriously required tens of thousands of training examples³⁶; however, recent research has demonstrated that it is possible to classify images accurately using relatively few labeled examples with the appropriate combination of pretraining of the CNN layers on unrelated image classification training sets^{37,38}, adversarial or unsupervised learning^{39,40}, network pruning⁴¹, and micro architecture tuning⁴². Once several categories have been learned the hard way, learning new categories should become more efficient. The increased efficiency allows for a lesser number of images to be used in training, referred to as a “*few shots*”.

Electron backscatter diffraction (EBSD) patterns (EBSPs) are an excellent case study for the use of *few-shot* transfer learning toward accelerating analysis of electron diffraction data. The scanning electron microscope (SEM)-based method involves the capture of 2D diffraction patterns produced from an incident electron beam scattering, diffracting, and escaping from a well-polished ‘bulk’ sample⁴³. The collected diffraction patterns contain significant structural and chemical information and are similar to those collected in other techniques such as convergent beam electron diffraction (CBED)^{44,45}. Despite the vast amount of information in the patterns, conventional EBSD has primarily focused on determining the three-dimensional orientation of individual grains in crystalline materials^{43,46–48}. Furthermore, the commercial technique typically relies on Hough-based indexing with a look-up table of interplanar angles constructed from the set of selected reflectors for phases specified by the user⁴⁹. This generally allows for phase differentiation of sufficiently distinct crystal structures^{50–52}, but the process remains susceptible to structural misclassification^{53–55}. Improvements to phase differentiation have been proposed and developed including dictionary indexing^{56–59}, spherical indexing^{60–62}, and more recently machine learning⁶³. While each offers significant advantages over the Hough-based method, these tools continue to require assumptions about the number of phases and/or their structure. For example, the dictionary-based approach requires simulation of a “master” pattern for each potential phase and every experimental pattern is matched against a dictionary of simulated patterns for all potential phases⁵⁶. The highest similarity match is selected as the most likely phase and orientation. Another available solution to phase differentiation is combining an electron backscattered pattern (EBSP) with information from energy dispersive X-ray spectroscopy (EDS) or wavelength dispersive X-ray spectroscopy (WDS)^{64–66}. While these have been adopted commercially, the singular EBSP is still analyzed with the Hough-based method and an expert user is required to evaluate the plausibility of returned matches. Applications using hand-drawn lines overlaid on individual Kikuchi diffraction patterns have been developed for determining the Bravais lattice or point group; however, they remain limited by at least one of the following: analysis time per pattern, the need for an expert crystallographer, or necessitating multiples of the same diffraction pattern with different SEM settings⁶⁷. Clearly, there remains a need for a rapid EBSP classification tool capable of functioning with one EBSP pattern while remaining suitable for even the most novice user.

Recently, the EBSD community has begun to explore the use of convolutional neural networks as a foundation for addressing modern EBSD challenges^{16,68–70}. Despite the marked advances these works have made, the requirements for simulating⁷¹ or collecting experimental diffraction patterns from a significant number of materials and crystallographic orientations remains a limiting factor. In this work, we test the validity of *few-shot* transfer learning, starting from ImageNet weights, applied to classify EBSPs to one of six space groups. Herein, space group refers to the symmetry group of a configuration in three-dimensional space. We compare the time to converge, the individual kernels (weights of the neurons), activation maps, and the performance of models trained from scratch or by transfer learning. Though there has been considerable progress on interpretability of machine learning systems, mostly in the field of eXplainable AI (XAI)⁷², fully understanding the internal mechanisms of deep neural networks is still an area of active research^{15,73}. Visualization of the most similar weights each model independently learned, their respective activation maps, and Shapley values can increase understanding of how the artificial intelligence models accomplishes its task. Building these XAI foundations can increase trust in the model's later predictions and help identify when the prediction is incorrect. In addition to evaluating each model's performance on holdout data, each model is also tested with 6900 EBSPs collected from a Ni₉₀Al₁₀ sample outside the training, testing, and validation data, and a space group map is generated from the individual classifications.

Results and discussion

Training metrics. Each model is first trained until the validation loss converged using a small number of the available diffraction patterns from each material in the six available space groups. The training and validation loss were recorded at the end of each pass through the training set (i.e. an epoch) to monitor the model's performance as the weights are updated. The validation loss is also monitored to determine when training should cease to prevent model underfitting or overfitting owing to a fixed number of training epochs. The loss function (categorical cross-entropy) is plotted in Supplementary Fig. S1 for reference. The goal of the neural network is to minimize the loss function by maximizing its prediction of the correct class. Figure 1a shows statistics for the training and validation loss for the model trained from scratch in grayscale. While the average training loss is low (0.84 ± 0.04) by the end of the first epoch, the average validation loss is observed to be 11.5 ± 2.5 . In comparison, by initializing the model with weights learned from ImageNet, the average training and validation loss are both observed to be small (0.56 ± 0.02 and 0.52 ± 0.3 , respectively) from the start (Fig. 1b). Figure 1c shows a magnified view of Fig. 1b to show the detailed training history. The improved starting loss, presumably owing to the strong filters learned in pre-training, also aids in rapid model convergence. Including the 15 epochs used

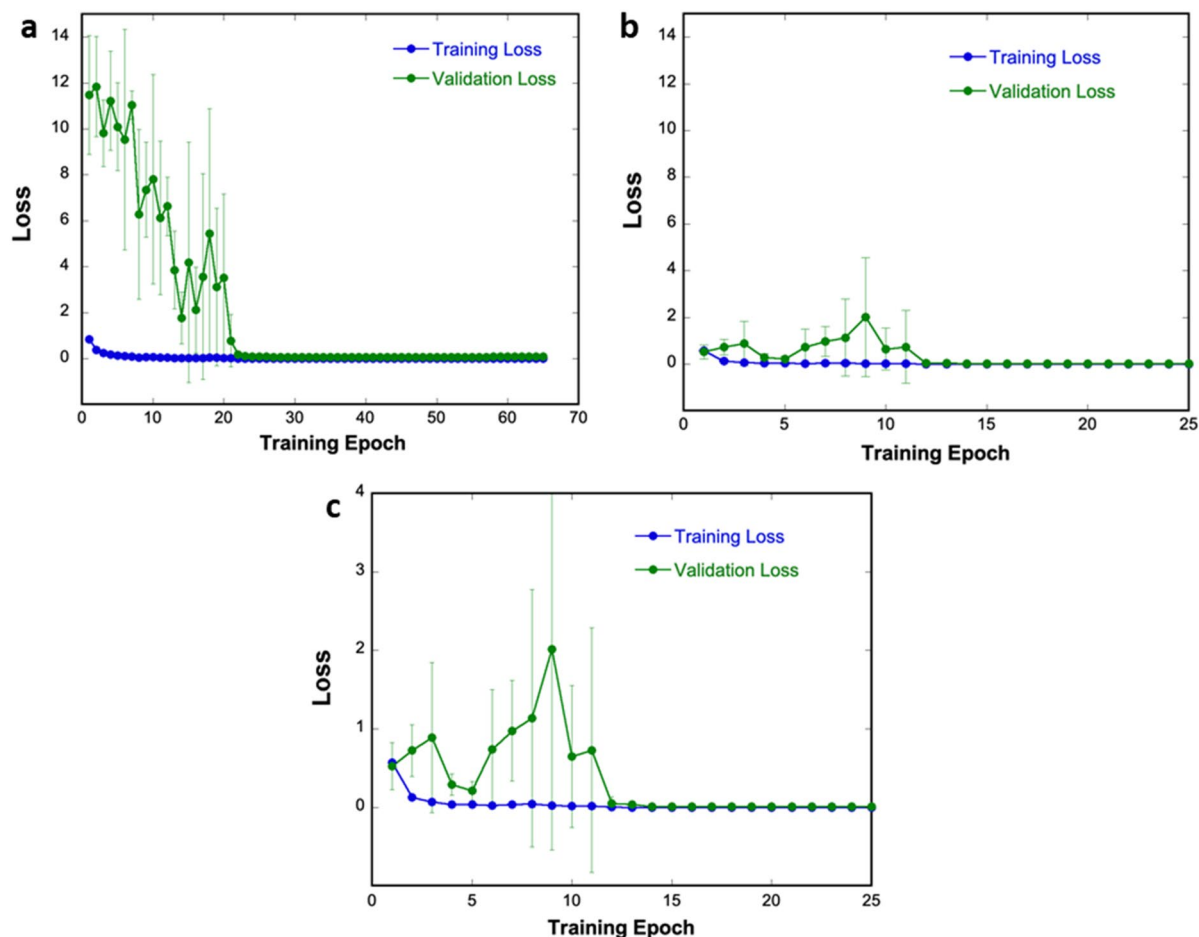


Figure 1. Training and validation history statistics for the two models. (a) The model trained from scratch using grayscale images. (b) The model trained starting from ImageNet weights using 3-channel stacked images. Data is plotted on the same y-axis as in (b). (c) The data in (b) for the transfer learning model plotted on a different scale. Error bars are one standard deviation from five trials per approach.

	Precision	Recall
Trained from scratch	0.93	0.92
Transfer learning	0.97	0.96

Table 1. Classification metrics. The class-weighted average *Precision* and *Recall* on the test data for each model. Both metrics are improved using the transfer learning approach to training the model. The same test data was used to benchmark each model.

to establish convergence, the model trained from scratch required 50 ± 12 epochs, while the transfer learning model required only 26 ± 3 epochs; this represents a twofold reduction in the average number of passes through the training set. Since each epoch with 2400 diffraction patterns requires two minutes on this hardware, a time savings of nearly an hour per training event is gained. With the amount of training data available to the CNN expected to grow, and therefore the time per epoch expected to increase, the time savings will become more pronounced.

Performance on holdout data. The time saving advantages garnered by fine-tuning the weights of an existing model are only valuable if the model performs similarly well or exceeds the model trained from scratch. Table 1 shows the class-weighted *Precision* and *Recall* for the best performing model from each approach, for which further analysis into the internal operations will be studied. See Supplementary Tables S1 and S2 for a breakdown of *Precision* and *Recall* by space group for the transfer learned and trained from scratch models, respectively. Both metrics are improved for the model using transfer learning even though the training, validation, and test sets were held constant. This implies that the feature extractors learned from ImageNet are not only relevant to this new domain, but also at least as valuable as those learned from scratch. This likely results from

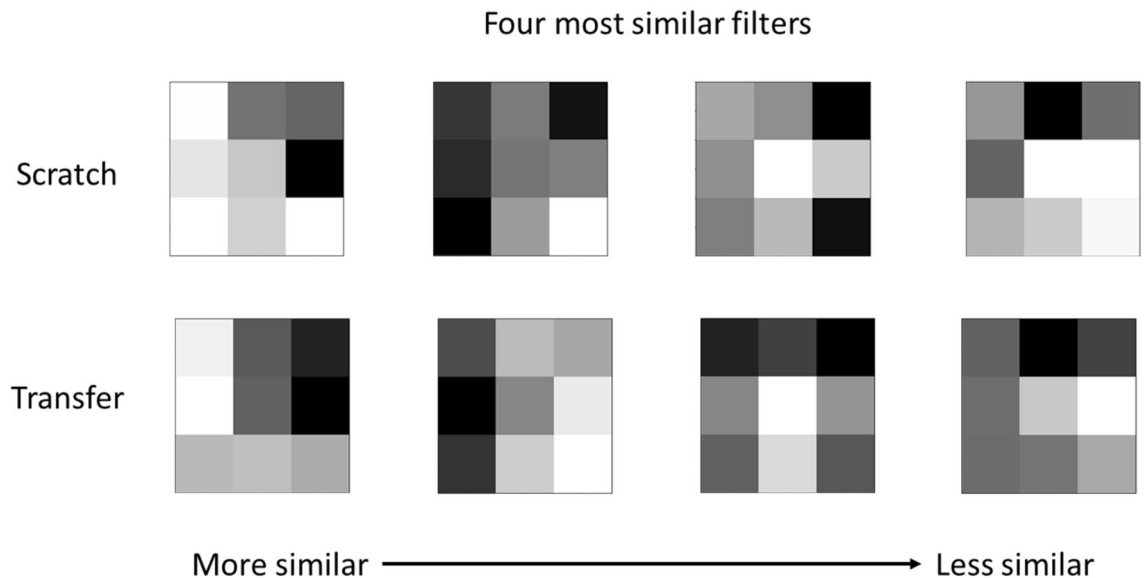


Figure 2. Kernel pairs with lowest Euclidean distance. Kernel pairs are grouped by column between the two models. The leftmost pair has the lowest Euclidean distance, with the kernels becoming less similar moving to the right.

the more general nature of the feature extractors necessary for optimal performance on ImageNet⁷⁴. It is possible that increasing the size of the dataset used to train the model from scratch could increase its performance to near that of the *few-shot* transfer learning model; however, the cost to training time would be notable. For this study, it was also important to keep the dataset and its partitions fixed for more deterministic comparisons.

Visualization. Since deep neural networks are used in this study, meaningful information about the internal mathematical operations performed necessitates studying several aspects of the model's inner workings. The first study involves visualization of the filters (or kernels) and corresponding feature maps. Filters and feature maps from the earliest layers in the model are most useful since deeper layers become more abstract. To identify corresponding filters in the two models, the Euclidean distance between all possible pairs of filters in a selected layer from the grayscale and pseudo RGB models were calculated and the similarity was ranked. The four most similar kernels between the two models in the first convolution layer are shown in Fig. 2. The pairs are grouped in columns between the model trained from scratch and using transfer learning. The independent co-learning of these filters suggest they are very valuable for feature tuning and selection on EBSPs. At this low-level in the model, the filters will predominantly be designed to identify edges at various orientations. The next layer will likely combine those edges into corners and small points, and the subsequent layers will figure out larger and larger shapes/features, such as the number of and relative angles between Kikuchi lines. Two of these four filters are recognizable as classical edge detection operators. The second filter has converged close to the x-direction Sobel edge detection operator⁷⁵ and the third filter resembles the Gabor filters with $\theta = 135^\circ$. Visualization of the feature maps resulting from these four filters can provide insight into their function.

Figure 3 shows the result of each filter from Fig. 2 being individually convolved over an input image from the six space groups studied. All feature maps shown are from the grayscale model. The first filter, and therefore the most similar filter between the two models, is primarily observed to perform an inversion on the input image. As a result, the band edges in each image are activated (white regions) and become more distinct. It is reasonable to speculate that deeper parts of the network extract the angle and relative locations of these intersections, a function that was also suggested in Ding et al.'s recent work⁷⁰. The fourth filter is observed to have normalized the contrast of the input image, perhaps to reduce the effects of atomic scattering factors (Z-contrast) observed in prior work⁶⁸. Further analysis of the class-specific feature importance can further improve understanding of the neural network's methodology.

Feature importance. Measurement and visualization of feature importance is another method to increase understanding of the deep neural network. This type of analysis is performed to help determine whether one should trust a prediction and why. There are a number of techniques available for CNNs including Gradient-weighted Class Activation Mapping (Grad-CAM)⁷⁶, activation maximization⁷⁷, LIME⁷⁸, and Shapley values⁷⁹. Several of those listed have been effectively demonstrated in other works involving EBSD patterns^{16,17,68,70}; however, this is the first to use Shapley values. In game theory, Shapley values are a solution to fairly distributing the gains and costs of several actors working in coalition. By definition, each Shapley value is the average expected marginal contribution of one actor after all possible combinations have been considered. In this case, the actors are the features in the images. Essentially, the Shapley value is the average expected marginal contribution of one actor after all possible combinations have been considered. While not perfect, this has proven a fair approach to allocating value in a variety of fields^{80–82}. The results of this analysis, further described in the Methods sec-

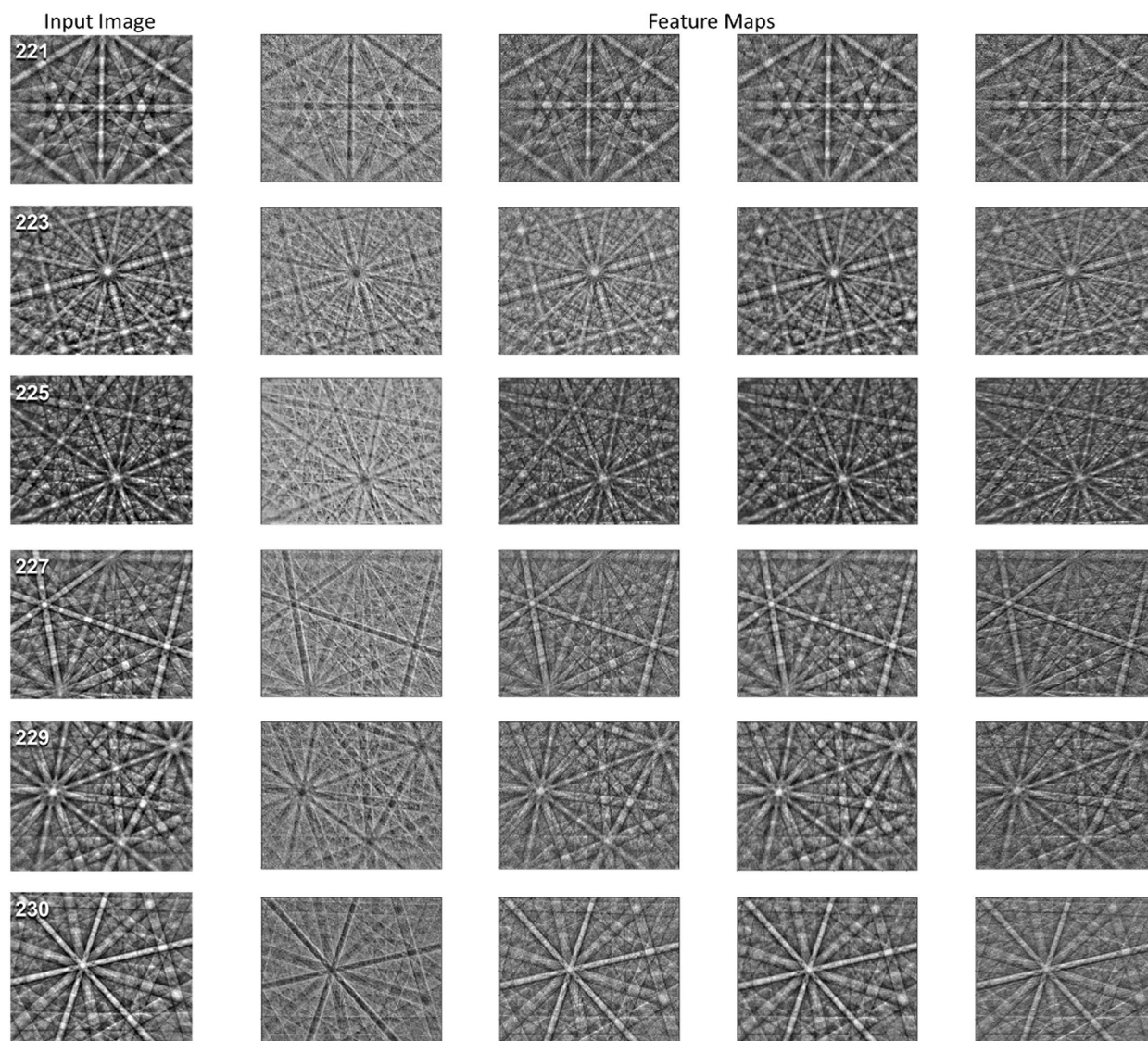


Figure 3. Selected feature maps from the first convolution layer. Feature maps extracted from the grayscale model corresponding to the filters shown previously in Fig. 2. One input image per space group is shown. From top to bottom, the six space groups are $Pm\bar{3}m$, $Pm\bar{3}n$, $Fm\bar{3}m$, $Fd\bar{3}m$, $Im\bar{3}m$, and $Ia\bar{3}d$.

tion, are shown in Fig. 4. Refer to Supplementary Fig. S2 for a demonstration of SHAP analysis on handwritten numbers.

The previously unseen input images are shown in the top row, and as semi-transparent grayscale backings behind each of the explanations. After random selection of the input image, it was verified that the model correctly identified the space group (thus pseudo-random selection). The middle row in Fig. 4 corresponds to the explanations for the correct prediction, while the bottom row displays the explanations for the next most likely (i.e. incorrect) space group. All 6 Shapley explanations ordered by most to least likely space group for each input are displayed in Supplementary Fig. S3. The positive (red) and negative (blue) contributions to each prediction are primarily at diffraction maxima (e.g. zone axes), band intersections, and outlining band edges. This lends credence that the model is indeed utilizing information grounded in the physics of EBSD. The clustering of Shapley values near zone axes is further reaffirming given the abundance of information and their role in classical diffraction pattern indexing⁸³.

Performance in practice. It is also of importance to demonstrate and compare the efficacy of both models in a real-world context, not only the patterns set aside for testing. In this case, the EBSD mapping of a dual phase sample serves to demonstrate that both model training strategies are capable in situations where the diffraction patterns are not collected from an ideal, single-phase material. Figure 5 top-left shows a backscattered electron image of a $Ni_{90}Al_{10}$ (wt%) sample containing a Ni-rich matrix (space group 225) along with Ni_3Al precipitates (space group 221) appearing raised from the surface. It is of importance to note that the model has not yet encountered a solid-solution phase such as the Ni with Al matrix present in this sample. The Al content in the Ni matrix was determined to be $19.7\% \pm 0.53\%$ (at%). The Al chemistry of the Ni_3Al precipitates was determined to

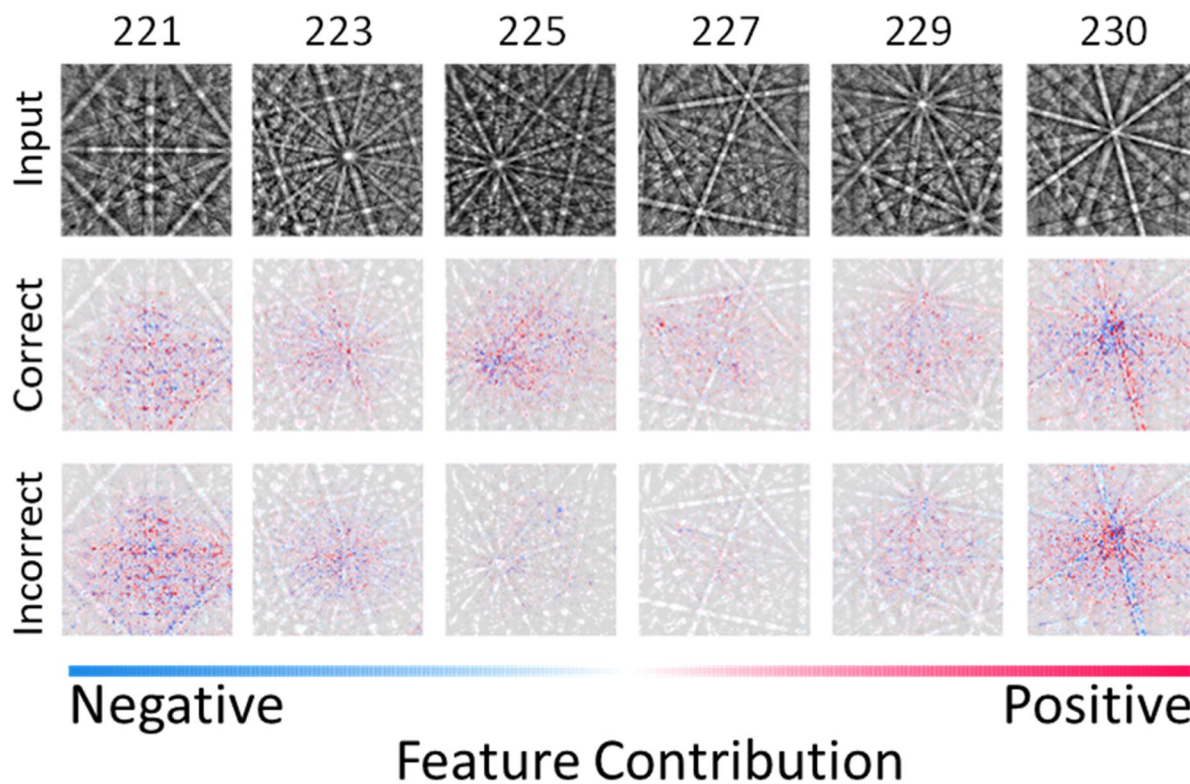


Figure 4. Visual explanation of feature contributions. Shapley values are computed for each input image to gauge the importance of features in the EBSPs. The first row is the raw input image. The second row corresponds to the Shapley values for the correct prediction. Row three corresponds to the first incorrect classification as ranked by softmax probability. From left to right, the six space groups are $Pm\bar{3}m$, $Pm\bar{3}n$, $Fm\bar{3}m$, $Fd\bar{3}m$, $Im\bar{3}m$, and $Ia\bar{3}d$.

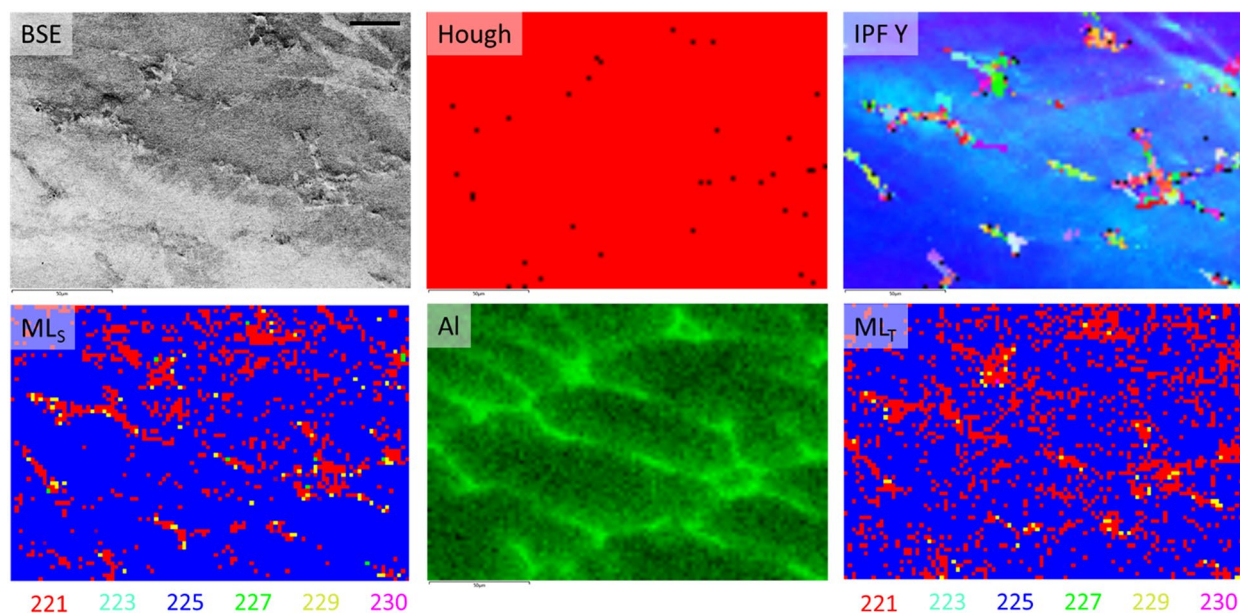


Figure 5. Phase mapping a dual-phase sample. Top left shows a backscattered electron image of the area to be mapped. The Ni_3Al precipitates appear raised in the Ni-rich matrix. Top middle shows the Hough transform-based phase map. Red pixels are identified as Ni or Ni_3Al with equal certainty, while black pixels were not solved. Top right is the inverse pole figure map in the Y-direction. Bottom left is the phase map produced using the predictions from the grayscale CNN. Bottom middle shows the aluminum EDS map. Bottom right is the phase map produced using the predictions from the *few-shot* transfer learning approach. There are 6900 total EBSPs (pixels). Scale bar = 25 μm .

	221	223	225	227	229	230
Scratch	984	0	5822	10	72	12
Transfer	1501	0	5342	0	56	1
Difference	517	0	(480)	(10)	(16)	(11)

Table 2. Tabulated predictions for the dual-phase sample. The number of patterns classified to each space group by the respective model. The difference is calculated by subtracting the number predicted by the model trained from scratch from the number predicted by the *few-shot* transfer learning model. Parentheses denotes the transfer learning model predicted fewer EBSPs belonging to the respective class. From left to right, the six space groups are $Pm\bar{3}m$, $Pm\bar{3}n$, $Fm\bar{3}m$, $Fd\bar{3}m$, $Im\bar{3}m$, and $Ia\bar{3}d$.

be $25.7\% \pm 0.74\%$ (at%). While partially selected for this reason, the material was primarily chosen since space groups 221 and 225 have two of the lowest F1-scores in each model and can readily be produced in this singular sample. Furthermore, it was important that the phases could be differentiated visually and easily to the reader, such as with EDS maps. While this means the phases could potentially be differentiated if the EBSD operator assigned reference chemistries to the phases in advance of collecting the EBSPs, the purpose of this demonstration is really to compare the transfer learning and trained from scratch models' abilities to identify the space group without further information. For the most complete phase identification in EBSD (i.e. lattice parameters), multiple analysis methods (e.g. XRD and EDS) may need to be employed.

The Hough-based phase map in Fig. 5 is a representative image of the expected results from commercial systems. The phase map consists of entirely one phase (shown in red) and Oxford Aztec software is observed to predict Ni and Ni₃Al with an equal number of bands and mean angular deviation (MAD), effectively a combined measure of certainty, for each diffraction pattern. It is the order of phase selection by the user in Oxford Aztec software that ultimately determines whether Ni or Ni₃Al is selected in this case. Example diffraction patterns from each phase are shown in Supplementary Fig. S4. Black pixels remain unindexed by the Hough method, typically a result of poor diffraction pattern quality. The inverse pole figure (IPF) in the Y-direction is provided along with the EDS map for aluminum to elucidate where the Ni₃Al is expected to be found within the Ni matrix.

The last two images in Fig. 5 show the phase maps produced by the scratch (ML_s) and *few-shot* transfer learned (ML_t) models. Over a statistical number of diffraction patterns, the two models are expected to produce similar answers with some variance, although the test set would suggest that the transfer learning model will generally outperform. Comparing the results with the Al EDS map and IPF Y, both models perform well at identifying the space group of each EBSP with a low false positive rate for the four space groups known not to be present. There are almost certainly some errors with regard to the EBSPs classified as space group 221 ($Pm\bar{3}m$) or 255 ($Fm\bar{3}m$), the *Precision* and *Recall* of each model alert the user of this in advance; however, the results over these 6900 EBSPs demonstrate the improvement over the Hough-based approach and, more importantly for this study, the robustness of a *few-shot* transfer learning approach.

The number of EBSPs each model classifies to the available space groups is tallied in Table 2. A total of 6900 diffraction patterns were individually identified by each neural network without any other information provided. The transfer learning model is observed to have a reduced misclassification rate to the space groups 227 ($Fd\bar{3}m$), 229 ($Im\bar{3}m$), and 230 ($Ia\bar{3}d$) in this phase map. The largest difference between the two models is for the diffraction patterns classified to space group 221; the class with the lowest *Precision* for each of the two models. The total difference of 517 diffraction patterns only equates to 7% of the total diffraction patterns; well within reason and the expected margin of error between the two models, particularly between these two space groups for the current models. Of those 517, only 480 of these predictions differ between space groups 221 and 225, the other 37 differences are due to false positives (i.e. 227, 229, and 230) in the model trained from scratch. The phase fraction of Ni₃Al likely lies somewhere between what is predicted by these two models. The results of this comparison also suggests that there is future opportunity to construct an approach leveraging Bayesian deep learning or an ensemble of (i.e. at least two) individually trained models making individual classifications combined with model averaging (e.g. voting) to reduce variance, provide insight into overall uncertainty, and identify when “no solution” is an appropriate answer^{84–86}.

Thus, a *few-shot* transfer learning approach to classifying electron backscatter diffraction patterns is an attractive method for leveraging the knowledge a deep neural network has attained in a previous context. The convolutional neural network-based approach to diffraction pattern classification is advantageous in that it requires little or no a-priori knowledge of the phases in a new sample and can readily be improved or expanded to new classes with the inclusion of new data. The similarity of EBSD patterns to those from techniques such as CBED suggests the *few-shot* transfer learning approach could also apply and potentially be more beneficial given the slower rate of data collection with other electron diffraction methods. Limitations of the current models exist in the number of space groups currently differentiable and the “black box” nature of neural networks. The number of space groups the model can learn to differentiate can be continuously expanded as more data becomes available for training. One of the goals of this work is to discern whether the *few-shot* transfer learning approach can be used to reduce the amount of data necessary for robust expansion to all 230 space groups or other diffraction pattern classification tasks. Indeed, we find that this approach does not hinder the model's performance on holdout or entirely new data and offers accelerated training time. While it can be difficult to precisely determine how the CNN performs this task, recent advances in eXplainable AI (e.g. SHAP) provides tools for developing insight and trust in the model's predictions. The combination of ease of scaling, flexibility of the framework, and ability

to assess aspects of a model's decision process support the utilization of CNNs and *few-shot* transfer learning as another tool for phase differentiation and symmetry identification in electron diffraction.

Materials and methods

Materials. Eighteen different single-phase materials, comprising 6 of the 10 space groups within the ($4/m\bar{3}2/m$) point group, were selected for training the space group classification CNN. Suitable samples for the remaining 4 space groups could not be obtained. The six space groups are $Pm\bar{3}m$, $Pm\bar{3}n$, $Fm\bar{3}m$, $Fd\bar{3}m$, $Im\bar{3}m$, and $Ia\bar{3}d$. Numerically, these are space groups 221, 223, 225, 227, 229, and 230. Space groups 221 and 223 are primitive cubic, 225 and 227 are face centered cubic, and 229 and 230 are body centered cubic. Each of the six space groups share the threefold rotary inversion necessary for inclusion in the ($4/m\bar{3}2/m$) point group. Supplementary Table S3 details the similarities and differences between the symmetry operations of the six space groups. The materials were FeAl, NiAl, Ni₃Al, Fe₃Ni, Cr₃Si, Mo₃Si, Ni, Al, NbC, TaC, TiC, Si, Ge, W, Ta, Fe, Al₄CoNi₂, and Al₄Ni₃. These materials were of low texture, typically less than 2 times random in any direction. Refer to Kaufmann et al. for the distributions of orientation, band contrast, and mean angular deviation for these samples⁶⁹.

A dual-phase material known to challenge Hough-based EBSD was fabricated to demonstrate and compare the capabilities of each CNN training approach. An additional constraint for the material selected was that the two space groups be identifiable within an EDS map, even though this meant an operator could force the Hough-based method to differentiate the phases by chemistry if they knew the phases in advance. An ingot of Ni₉₀Al₁₀ (wt%) was arc melted and processed via hot rolling at 600 °C to 45% reduction in thickness followed by aging at 600 °C for 4 h and air cooled. X-ray diffraction (XRD) using a Rigaku Miniflex X-ray Diffractometer with a 1D detector, a step size of 0.02°, 5° per minute scan rate, and Cu K α radiation (wavelength $\lambda = 1.54059$ Å) was performed to confirm the existence of phases belonging to space groups 221 ($Pm\bar{3}m$) and 225 ($Fm\bar{3}m$) (Supplementary Fig. S5).

Electron backscatter diffraction pattern collection. EBSD patterns (EBSPs) were collected as previously described in Kaufmann et al.⁶⁸. Diffraction patterns were collected using a Thermo Scientific (formerly FEI) Apreo scanning electron microscope (SEM) equipped with an Oxford Symmetry EBSD detector utilized in high resolution (1244 × 1024) mode. The geometry of the setup was held constant as follows. The working distance was 18.1 mm ± 0.1 mm. Oxford Aztec software was used to set the detector insertion distance to 160.2 and the detector tilt to -3.1. The imaging parameters were 20 kV accelerating voltage, 51 nA beam current, 0.8 ms ± 0.1 ms dwell time, and 30-pattern averaging. The Hough indexing parameters were 12 Kikuchi bands, a Hough resolution of 250, and band center indexing.

After collecting high resolution EBSPs from each material, all patterns collected were exported as tiff images. The images were resized for the CNN using the `resize` function in `scikit-image`. All collected data for each material was individually assessed by the neural network, and the collection of images for each sample may contain partial or low-quality diffraction patterns, which could decrease the accuracy of their identification. The test data was not filtered to better assess the model as it would be applied in practice.

Neural network architecture. The well-studied convolutional neural network architecture Xception⁸⁷ was selected as the basis architecture for fitting a model that determines which space group a diffraction pattern originated from. The Xception architecture was used without modifications for training the model from scratch and the transfer learning process to facilitate comparison of the training metrics, performance, and internal workings. Selection of this network was partially based on Xception or derivatives of Xception being used previously in the EBSD community^{16,68–70}. Xception is also a standard model with ImageNet weights readily available in deep learning APIs such as Keras⁸⁸. A schematic of the convolutional neural network operating on an EBSP is provided in Fig. S6. Due to space constraints, only the resultant feature maps from selected convolutional layers are shown after image input and before the 2048-dimensional vector. For a complete description of the Xception architecture, please refer to Fig. 5 in Xception: Deep Learning with Depthwise Separable Convolutions⁸⁷.

Neural network training. For both the transfer learning and from scratch approaches, training was performed using 400 diffraction patterns per space group. The diffraction patterns supplied at training were evenly divided between the number of materials per space group that the model had access to during training. For example, if the model was given two materials of the same space group during training, 200 diffraction patterns per material were made available. The validation set contained 100 diffraction patterns per space group, equivalent to the standard 80:20 train/validation split. The validation set was only used to monitor the training progress and model convergence. The test set contains the rest of the patterns (a total of 145,453 images; refer to Supplementary Table S1 for class distribution) that were not used for training or validation. The images selected for training, testing, and validation were the exact same for transfer learning and for from scratch learning.

Model hyperparameters were selected or tuned as follows. Adam (adaptive moment estimation) optimization with a learning rate of 0.001⁸⁹, and a minimum delta of 0.001 as the validation loss were employed for convergence criteria. Adam is chosen for its ability to work well with little hyperparameter tuning, relatively low memory requirements, and its ability to smooth the steps of gradient descent using momentum. Monitoring of validation loss, i.e. early stopping criteria, was employed instead of a fixed number of epochs to allow both models the necessary epochs to converge while keeping the risk of overfitting to the training data low. The patience criteria for validation loss convergence was set to 15 epochs to allow for sufficient certainty that the model had converged and was unlikely to meaningfully improve. The weight decay was set to $1e^{-5}$ following previous optimization work⁸⁷. The CNNs were implemented with TensorFlow⁹⁰ and the Keras API⁸⁸ and model training was performed using an NVIDIA Titan V.

Diffraction pattern classification. Each diffraction pattern collected, but not used in training (> 140,000 images), was evaluated in a random order by the corresponding trained CNN model without further information. The output classification of each diffraction pattern was recorded, saved in a (.csv) file, and tabulated. *Precision* and *Recall* were calculated for each material and each space group using Scikit-learn⁹¹. *Precision* (Eq. 1) for each class (e.g. 230) is defined as the number of correctly predicted images out of all photos predicted to belong to that class (e.g. 230). *Recall* (Eq. 2) is the number of correctly predicted images for each class divided by the actual number of images for the class. F1-score is the weighted harmonic of the *Precision* and *Recall* and is particularly valuable in situations where the number of test images per class is variable. A high F1-score means the model has low false positives and low false negatives.

$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}, \quad (1)$$

$$\text{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}. \quad (2)$$

Neural network insight. Comparisons between the resultant models from pseudo RGB transfer learning and training the model from scratch offer an opportunity to understand how the CNNs go about their given task. Visualization techniques are implemented using the keras-vis package v0.4.1⁷⁷. Filters from the first layer of each model are extracted and plotted as 3×3 matrices with matplotlib⁹². The first layer was targeted since the earliest filters represent lower level features such as colors and edges. The Euclidean distance between the individual filter arrays in each model was computed using the NumPy linear algebra toolbox to compute L2 norms⁹³ and the four most similar learned filters between the two models were identified. The outputs (a.k.a. feature maps) of the first layer corresponding to these four filters are also extracted to examine the activations of the two approaches. The feature maps from earlier convolution layers are more useful since deeper layers operate in feature space and are therefore more difficult to understand^{29,70}. Lastly, Shapley values, deeply rooted in game theory^{94,95}, are estimated using the DeepSHAP tools in SHAP⁹⁶. SHAP uses a distribution of background samples, approximates the model with a linear function between each background data sample and the current input to be explained, and assumes the input features are independent to compute approximate SHAP values. The sum of the SHAP values equals the difference between the expected model output (averaged over the background dataset) and the current model output. One hundred images per space group were used as background samples in conjunction with one new input image per space group. The total number of diffraction patterns used as a background follows the SHAP software protocols. When plotted as an overlay, red pixels represent positive SHAP values that increase the probability of the class, while blue pixels represent negative SHAP values that reduce the probability of the class. An example of SHAP analysis on handwritten digits from the MNIST database⁹⁷ is shown in Supplementary Fig. S2. For a given image, the presence and absence of features that positively correlate with a class are shown in red, while negative correlations are shown in blue. As an example, in the image of a ‘four’ the lack of a connection on top makes it a four instead of a nine. Combined, these insights into the operations of the neural network can further substantiate the validity of the transfer learning approach, increase trust by better understanding the model’s methods, and provide indications in cases where the model is incorrect about future predictions.

Data availability

The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request. The code will be available in the GitHub repository https://github.com/krkau/fma/EBSD_transfer_learning and in the online Zenodo repository (<https://doi.org/10.5281/zenodo.3564937>)⁹⁸.

Received: 28 November 2020; Accepted: 31 March 2021

Published online: 14 April 2021

References

- O'Mara, J., Meredig, B. & Michel, K. Materials data infrastructure: A case study of the citrination platform to examine data import, storage, and access. *JOM* **68**, 2031–2034 (2016).
- Ward, L. *et al.* Matminer: An open source toolkit for materials data mining. *Comput. Mater. Sci.* **152**, 60–69 (2018).
- Mrdjenovich, D. *et al.* propnet: A knowledge graph for materials science. *Matter* **2**, 464–480 (2020).
- Tabor, D. P. *et al.* Accelerating the discovery of materials for clean energy in the era of smart automation. *Nat. Rev. Mater.* **3**, 5–20 (2018).
- Ong, S. P. *et al.* Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Comput. Mater. Sci.* **68**, 314–319 (2013).
- Saal, J. E., Kirklin, S., Aykol, M., Meredig, B. & Wolverton, C. Materials design and discovery with high-throughput density functional theory: The open quantum materials database (OQMD). *JOM* **65**, 1501–1509 (2013).
- Jha, D. *et al.* ElemNet: Deep learning the chemistry of materials from only elemental composition. *Sci. Rep.* **8**, 17593 (2018).
- Oviedo, F. *et al.* Fast and interpretable classification of small X-ray diffraction datasets using data augmentation and deep neural networks. *NPJ Comput. Mater.* **5**, 1–9 (2019).
- Kaufmann, K. *et al.* Discovery of high-entropy ceramics via machine learning. *NPJ Comput. Mater.* **6**, 42 (2020).
- DeCost, B. L. & Holm, E. A. A computer vision approach for automated analysis and classification of microstructural image data. *Comput. Mater. Sci.* **110**, 126–133 (2015).
- McAuliffe, T. P. *et al.* Advancing characterisation with statistics from correlative electron diffraction and X-ray spectroscopy, in the scanning electron microscope. *Ultramicroscopy* **211**, 112944 (2020).
- Kaufmann, K. & Vecchio, K. S. Searching for high entropy alloys: A machine learning approach. *Acta Mater.* **198**, 178–222 (2020).

13. Qiao, Z., Welborn, M., Anandkumar, A., Manby, F. R. & Miller, T. F. OrbNet: Deep learning for quantum chemistry using symmetry-adapted atomic-orbital features. *J. Chem. Phys.* **153**, 124111 (2020).
14. Holm, E. A. In defense of the black box. *Science (80-)*. **364**, 26–27 (2019).
15. Adadi, A. & Berrada, M. Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access* **6**, 52138–52160 (2018).
16. Kaufmann, K. *et al.* Crystal symmetry determination in electron diffraction using machine learning. *Science (80-)*. **367**, 564–568 (2020).
17. Foden, A., Previero, A. & Britton, T. B. Advances in electron backscatter diffraction. Preprint at <http://arxiv.org/abs/1908.04860> (2019).
18. Ziletti, A., Kumar, D., Scheffler, M. & Ghiringhelli, L. M. Insightful classification of crystal structures using deep learning. *Nat. Commun.* **9**, 2775 (2018).
19. Modarres, M. H. *et al.* Neural network for nanoscience scanning electron microscope image recognition. *Sci. Rep.* **7**, 1–12 (2017).
20. Stan, T., Thompson, Z. T. & Voorhees, P. W. Optimizing convolutional neural networks to perform semantic segmentation on large materials imaging datasets: X-ray tomography and serial sectioning. *Mater. Charact.* **160**, 110119 (2020).
21. DeCost, B. L., Lei, B., Francis, T. & Holm, E. A. High throughput quantitative metallography for complex microstructures using deep learning: A case study in ultrahigh carbon steel. *Microsc. Microanal.* **25**, 21–29 (2019).
22. Roberts, G. *et al.* Deep learning for semantic segmentation of defects in advanced STEM images of steels. *Sci. Rep.* **9**, 1–12 (2019).
23. de Haan, K., Ballard, Z. S., Rivenson, Y., Wu, Y. & Ozcan, A. Resolution enhancement in scanning electron microscopy using deep learning. *Sci. Rep.* **9**, 1–7 (2019).
24. Xie, T. & Grossman, J. C. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Phys. Rev. Lett.* **120**, 145301 (2018).
25. Spurgeon, S. R. *et al.* Towards data-driven next-generation transmission electron microscopy. *Nat. Mater.* <https://doi.org/10.1038/s41563-020-00833-z> (2020).
26. Rosenstein, M. T., Marx, Z., Kaelbling, L. P. & Dietterich, T. G. To transfer or not to transfer. In *Neural Information Processing Systems (NIPS '05) Workshop Inductive Transfer: 10 Years Later* (2005).
27. Jha, D. *et al.* Enhancing materials property prediction by leveraging computational and experimental data using deep transfer learning. *Nat. Commun.* **10**, 1–12 (2019).
28. Thompson, J. A. F., Schonwiesner, M., Bengio, Y. & Willett, D. How transferable are features in convolutional neural network acoustic models across languages? In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing—Proceedings 2019-May*, 2827–2831 (Institute of Electrical and Electronics Engineers Inc., 2019).
29. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
30. Pan, X. *et al.* Multi-task Deep learning for fine-grained classification/grading in breast cancer histopathological images. In *Studies in Computational Intelligence* **810**, 85–95 (Springer, 2020).
31. Xie, Y. & Richmond, D. Pre-training on grayscale imagenet improves medical image classification. In *The European Conference on Computer Vision (ECCV) Workshops* **11134** (Springer, 2018).
32. Gonzalez, J., Bhowmick, D., Beltran, C., Sankaran, K. & Bengio, Y. Applying knowledge transfer for water body segmentation in Peru. Preprint at <http://arxiv.org/abs/1912.00957> (2019).
33. Pan, S. J. & Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**, 1345–1359 (2010).
34. Deng, J. *et al.* ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition* 248–255 (Institute of Electrical and Electronics Engineers (IEEE), 2010). <https://doi.org/10.1109/cvpr.2009.5206848>
35. Bloom, P. *How Children Learn the Meanings of Words* (MIT Press, 2000).
36. Felzenszwalb, P. F. & Huttenlocher, D. P. Pictorial structures for object recognition. *Int. J. Comput. Vis.* **61**, 55–79 (2005).
37. Fei-Fei, L., Fergus, R. & Perona, P. One-shot learning of object categories. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**, 594–611 (2006).
38. Lake, B. M., Salakhutdinov, R. & Tenenbaum, J. B. Human-level concept learning through probabilistic program induction. *Science (80-)*. **350**, 1332–1338 (2015).
39. Zhang, R., Che, T., Ghahramani, Z., Bengio, Y. & Song, Y. MetaGAN: An Adversarial Approach to Few-Shot Learning. In *NeurIPS 2018* 2365–2374 (2018).
40. Liu M.-Y. *et al.* Few-shot unsupervised image-to-image translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019).
41. Li, H., Kadav, A., Durdanovic, I., Samet, H. & Graf, H. P. Pruning filters for efficient ConvNets. In *5th Int. Conf. Learn. Represent.* 1–13 (2016).
42. Guo, Z. *et al.* Single path one-shot neural architecture search with uniform sampling. Preprint at <http://arxiv.org/abs/1904.00420> (2019).
43. Schwartz, A. J., Kumar, M., Adams, B. L. & Field, D. P. *Electron Backscatter Diffraction in Materials Science* (Springer Science+Business Media, LLC, 2009). <https://doi.org/10.1007/978-0-387-88136-2>.
44. Vecchio, K. S. & Williams, D. B. Convergent beam electron diffraction study of Al₃Zr in Al–Zr AND Al–Li–Zr alloys. *Acta Metall.* **35**, 2959–2970 (1987).
45. Vecchio, K. S. & Williams, D. B. Convergent beam electron diffraction analysis of the T 1 (Al₂CuLi) phase in Al–Li–Cu alloys. *Metall. Trans. A* **19**, 2885–2891 (1988).
46. Tong, V. S., Knowles, A. J., Dye, D. & Britton, T. B. Rapid electron backscatter diffraction mapping: Painting by numbers. *Mater. Charact.* **147**, 271–279 (2019).
47. Thomsen, K., Schmidt, N. H., Bewick, A., Larsen, K. & Goulden, J. Improving the accuracy of orientation measurements using EBSD. *Microsc. Microanal.* **19**, 724–725 (2013).
48. Zhu, C., Kaufmann, K. & Vecchio, K. S. Novel remapping approach for HR-EBSD based on demons registration. *Ultramicroscopy* **208**, 112851 (2020).
49. Lassen, N. C. K. *Automated Determination of Crystal Orientations from Electron Backscattering Patterns* (The Technical University of Denmark, 1994).
50. Britton, T. B. *et al.* Factors affecting the accuracy of high resolution electron backscatter diffraction when using simulated patterns. *Ultramicroscopy* **110**, 1443–1453 (2010).
51. Hielscher, R., Bartel, F. & Britton, T. B. Gazing at crystal balls: Electron backscatter diffraction pattern analysis and cross correlation on the sphere. *Ultramicroscopy* **207**, 112836 (2019).
52. Foden, A., Collins, D. M., Wilkinson, A. J. & Britton, T. B. Indexing electron backscatter diffraction patterns with a refined template matching approach. *Ultramicroscopy* **207**, 112845 (2019).
53. Karthikeyan, T., Dash, M. K., Saroja, S. & Vijayalakshmi, M. Evaluation of misindexing of EBSD patterns in a ferritic steel. *J. Microsc.* **249**, 26–35 (2013).
54. Chen, C. L. & Thomson, R. C. The combined use of EBSD and EDX analyses for the identification of complex intermetallic phases in multicomponent Al–Si piston alloys. *J. Alloys Compd.* **490**, 293–300 (2010).
55. McLaren, S. & Reddy, S. M. Automated mapping of K-feldspar by electron backscatter diffraction and application to 40Ar/39Ar dating. *J. Struct. Geol.* **30**, 1229–1241 (2008).

56. Ram, F. & De Graef, M. Phase differentiation by electron backscatter diffraction using the dictionary indexing approach. *Acta Mater.* **144**, 352–364 (2018).
57. Chen, Y. H. *et al.* A dictionary approach to electron backscatter diffraction indexing. *Microsc. Microanal.* <https://doi.org/10.1017/S1431927615000756> (2015).
58. Singh, S. *et al.* High resolution low kV EBSD of heavily deformed and nanocrystalline Aluminium by dictionary-based indexing. *Sci. Rep.* <https://doi.org/10.1038/s41598-018-29315-8> (2018).
59. Ram, F., Wright, S., Singh, S. & De Graef, M. Error analysis of the crystal orientations obtained by the dictionary approach to EBSD indexing. *Ultramicroscopy* **181**, 17–26 (2017).
60. Day, A. P. Spherical EBSD. *J. Microsc.* <https://doi.org/10.1111/j.1365-2818.2008.02011.x> (2008).
61. Lenthe, W. C., Singh, S. & Graef, M. D. A spherical harmonic transform approach to the indexing of electron back-scattered diffraction patterns. *Ultramicroscopy* **207**, 112841 (2019).
62. Zhu, C., Kaufmann, K. & Vecchio, K. Automated reconstruction of spherical Kikuchi maps. *Microsc. Microanal.* <https://doi.org/10.1017/S1431927619000710> (2019).
63. McAuliffe, T. P., Dye, D. & Britton, T. B. Spherical-angular dark field imaging and sensitive microstructural phase clustering with unsupervised machine learning. *Ultramicroscopy* **219**, 113132 (2020).
64. Nowell, M. M. & Wright, S. I. Phase differentiation via combined EBSD and XEDS. *J. Microsc.* **213**, 296–305 (2004).
65. Goehner, R. P. & Michael, J. R. Phase identification in a scanning electron microscope using backscattered electron Kikuchi patterns. *J. Res. Natl. Inst. Stand. Technol.* **101**, 301–308 (1996).
66. Dingley, D. J. & Wright, S. I. Phase identification through symmetry determination in EBSD patterns. In *Electron Backscatter Diffraction in Materials Science* (eds. Schwartz, A., Kumar, M., Adams, B. & Field, D.) 97–107 (Springer, 2009). <https://doi.org/10.1007/978-0-387-88136-2>
67. Li, L. & Han, M. Determining the Bravais lattice using a single electron backscatter diffraction pattern. *J. Appl. Crystallogr.* <https://doi.org/10.1107/S1600576714025989> (2015).
68. Kaufmann, K., Zhu, C., Rosengarten, A. S. & Vecchio, K. S. Deep neural network enabled space group identification in EBSD. *Microsc. Microanal.* **26**, 447–457 (2020).
69. Kaufmann, K. *et al.* Phase Mapping in EBSD using convolutional neural networks. *Microsc. Microanal.* **26**, 458–468 (2020).
70. Ding, Z., Pascal, E. & De Graef, M. Indexing of electron back-scatter diffraction patterns using a convolutional neural network. *Acta Mater.* **199**, 370–382 (2020).
71. Callahan, P. G. & De Graef, M. Dynamical electron backscatter diffraction patterns. Part I: Pattern simulations. *Microsc. Microanal.* **19**, 1255–1265 (2013).
72. *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning* (eds. Samek, W., Montavon, G., & Vedaldi, A.) 11700, (Springer International Publishing, 2019)
73. Carter, S., Armstrong, Z., Schubert, L., Johnson, I. & Olah, C. Activation atlas. *Distill* **4**, e15 (2019).
74. Graff, C. A. & Ellen, J. *Correlating Filter Diversity with Convolutional Neural Network Accuracy* 75–80 (Institute of Electrical and Electronics Engineers (IEEE), 2017). <https://doi.org/10.1109/icmla.2016.0021>.
75. Kanopoulos, N., Vasanthavada, N. & Baker, R. L. Design of an image edge detection filter using the sobel operator. *IEEE J. Solid-State Circuits* **23**, 358–367 (1988).
76. Selvaraju, R. R. *et al.* Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision* **2017**, 618–626 (2017).
77. Kotikalapudi, R. keras-vis. <https://github.com/raghakot/keras-vis> (2017).
78. Ribeiro, M. T., Singh, S. & Guestrin, C. “Why Should I Trust You?” Explaining the Predictions of Any Classifier. (2016). <https://doi.org/10.1145/2939672.2939778>
79. Lundberg, S. M., Allen, P. G. & Lee, S.-I. A Unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems 30* (eds. Guyon, I. *et al.*) 4765–4774 (Curran Associates, Inc., 2017).
80. Chen, H., Janizek, J. D., Lundberg, S. & Lee, S.-I. True to the Model or True to the Data? Preprint at <http://arxiv.org/abs/2006.16234> (2020).
81. Ghorbani, A. & Zou, J. Data shapley: Equitable valuation of data for machine learning. In *36th International Conference on Machine Learning, ICML 2019* **2019**, 4053–4065 (International Machine Learning Society (IMLS), 2019).
82. Wu, M., Wicker, M., Ruan, W., Huang, X. & Kwiatkowska, M. A game-based approximate verification of deep neural networks with provable guarantees. *Theor. Comput. Sci.* **807**, 298–329 (2020).
83. Winkelmann, A., Britton, T. B. & Nolze, G. Constraints on the effective electron energy spectrum in backscatter Kikuchi diffraction. *Phys. Rev. B* **99**, 064115 (2019).
84. Yuanyuan, C. & Zhibin, W. Quantitative analysis modeling of infrared spectroscopy based on ensemble convolutional neural networks. *Chemom. Intell. Lab. Syst.* **181**, 1–10 (2018).
85. Blundell, C., Cornebise, J., Kavukcuoglu, K. & Wierstra, D. Weight uncertainty in neural networks. In *32nd Int. Conf. Mach. Learn. ICML 2015* **2**, 1613–1622 (2015).
86. Hinton, G., Vinyals, O. & Dean, J. Distilling the Knowledge in a Neural Network. Preprint at <http://arxiv.org/abs/1503.02531> (2015).
87. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE International Conference on Computer Vision* 1251–1258 (The Computer Vision Foundation, 2017).
88. Chollet, F. K. (2015). <https://github.com/keras-team/keras>.
89. Kingma, D. P. & Ba, J. L. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015—Conference Track Proceedings* 1–15 (2015).
90. Abadi, M. *et al.* TensorFlow: A system for large-scale machine learning. In *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation* 265–283 (USENIX, 2016).
91. Pedregosa, F. *et al.* Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
92. Hunter, J. D. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.* **9**, 99–104 (2007).
93. Van Der Walt, S., Colbert, S. C. & Varoquaux, G. The NumPy array: A structure for efficient numerical computation. *Comput. Sci. Eng.* **13**, 22–30 (2011).
94. Shapley, L. S. Stochastic games. *Proc. Natl. Acad. Sci. U. S. A.* **39**, 1095–1100 (1953).
95. Shapley, L. S. A value for n-person games. In *Contributions to the Theory of Games (AM-28), Volume II* (eds. Kuhn, H. W. & Tucker, A. W.) 307–317 (Princeton University Press, 1953).
96. Lundberg, S. M. & Lee, S.-I. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems* Volume 30, 4765–4774 (2017).
97. LeCun, Y., Bottou, L., Bengio, Y. & Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **86**, 2278–2323 (1998).
98. krkaufma. krkaufma/Electron-Diffraction-CNN v1.0.1. (2019). Code available at <https://doi.org/10.5281/ZENODO.3564937>.

Acknowledgements

The authors would like to thank William M. Mellor for composing Supplementary Table S3.

Author contributions

K.K. and H.L. co-designed the initial project scope and aims. K.K. performed the bulk of the programming work with guidance from H.L. K.K. drafted the early versions of the manuscript and figures. X.L. performed the experimental work for the Ni₉₀Al₁₀ sample. K.S.V. guided the development of the project and reviewed and revised the early drafts of the manuscript. All authors participated in analyzing and interpreting the final data and contributed to the discussions and revisions of the manuscript.

Funding

Supported by the U.S. Department of Defense (DoD) [through the National Defense Science and Engineering Graduate Fellowship (NDSEG) Program] and the ARCS Foundation, San Diego Chapter (K.K.); and the Oerlikon Group (K.S.V.).

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/s41598-021-87557-5>.

Correspondence and requests for materials should be addressed to K.S.V.

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) 2021