

# Materials Discovery: Understanding Polycrystals from Large-Scale Electron Patterns

Ruoqian Liu, Ankit Agrawal, Wei-keng Liao, Alok Choudhary  
*Electrical Engineering and Computer Science*  
*Northwestern University*  
*Evanston, IL 60208*  
 {rll943, ankitag, wklio, choudhar}@eecs.northwestern.com

Marc De Graef  
*Materials Science and Engineering*  
*Carnegie Mellon University*  
*Pittsburgh, PA 15213*  
 degrae@cmu.edu

**Abstract**—This paper explores the idea of modeling a large image data collection of polycrystal electron patterns, in order to detect insights in understanding materials discovery. There is an emerging interest in applying big data processing, management and modeling methods to scientific images, which often come in a form and with patterns only interpretable to domain experts. While large-scale machine learning approaches have demonstrated certain superiority in analyzing, summarizing, and providing an understandable route to data types like natural images, speeches and texts, scientific images is still a relatively unexplored area. Deep convolutional neural networks, despite their recent triumph in natural image understanding, are still rarely seen adapted to experimental microscopic images, especially in a large scale. To the best of our knowledge, we present the first deep learning solution towards a scientific image indexing problem using a collection of over 300K microscopic images. The result obtained is 54% better than a dictionary lookup method which is state-of-the-art in the materials science society.

**Keywords**—Deep learning; convolutional neural networks; materials design; materials discovery; electronic images; EBSD

## I. INTRODUCTION

With the rapid development of computer and information technology in the last several decades, a prominent trend now seen in various fundamental science researches is the increase of the use of data to drive discoveries. Almost every field in science and engineering has been or is being transformed from data-poor to increasingly data-rich. Enormous amounts of data in science and engineering has been, are being, and will continuously be generated and collected in massive scale, in the order of tera- to petabytes. Moreover, a great amount of the data has been curated, labeled, and made publicly available via the Internet. The trend of data sharing, data openness, collaborative data curation is showing a growing popularity.

This leads to a paradigm change in science, coined the fourth paradigm of scientific discovery by Hey *et al.* [1], and further discussed in the materials discovery context by Agrawal and Choudhary [2]. Generally, scientific discoveries had previously been driven by empirical experiments, theoretical foundations, computer simulations and now, automated or semi-automated data modeling and analyzing

tools. Among those tools, the board application of Machine Learning (ML) and Data Mining (DM) methods have been essential. Recent revolution in Artificial Intelligence (AI), concretized by the success of deep neural networks combined with large scale data and high performance computing technology, has stimulated the adoption of such methodology in various application domains.

In the meantime, the necessity of such a paradigm transition has been most significant in the field of materials science. A number of notable government initiatives and high-profile documents [3], [4] have specifically urged the accelerated materials discovery, to transform the process of identifying and/or synthesizing new materials from a slow-paced, physics-based process to a potentially much faster, informatics-based and data-intensive one. However, despite the growing interests and mounting demand, the use of advanced machine learning methods in understanding the synthesis and/or processing routes of materials has been largely limited. A very recent work by Raccuglia *et.al* [5] has successfully utilized data from failed experiments, with the classifier being support vector machine (SVM). The demonstration of accurate reading of large-scale materials images with deep neural networks are almost never seen.

To the best of our knowledge, this paper presents the first deep learning solution to large-scale, automatic assessment of electron images, as a step towards developing general purpose learning models without much guidance from the domain experts. This is in tune with the theme of Artificial General Intelligence [6].

At the risk of oversimplification, there are two quantities of a material (including, e.g., metals, alloys, composites, ceramics, polymers) to be analyzed in the computational materials science: (a) its internal structure, and (b) its physical/chemical properties and/or engineering performance characteristics (which is closely related to properties). In addition to understanding each of these two quantities, what's more valuable is building the linkage between them. The so-called structure-property relationship, if accurately modeled, provides key information to the discovery of new, alternative material types [7]. While quantity (b) is often

probed by some measurement in experiments or simulation, quantity (*a*), most of the time, exists in the form of images produced by various materials characterization tools.

This reduces materials data management and analysis mostly to image analysis. Scientific images like X-Ray and electron microscopy pose great challenges to general computer vision techniques. Learning from scientific images are very different from the recognizing objects in natural images. While we are aiming at beating average human performance on tasks like object recognition in natural images, the bar is even higher on scientific images, as only professionals with strong domain expertise can probably untangle the visual complexity and reveal the underlying meaning.

In this work, we demonstrate the generalization capability of deep neural networks in the application field of materials discovery. More specifically, we use deep Convolutional Neural Networks (CNNs) to index electron microscopy images, presented in grey scale, with crystalline orientations, represented by three angles. A special loss function is particularly designed to handle the angular periodicity. We also visualize what is learned out of the first convolutional layer to justify the effectiveness of a learned model. The emphasis of the technique is two-fold: (a) the use of a large size of data in building a deep neural network, and (b) the end-to-end modeling with raw pixels of images.

The rest of paper is organized as follows. In Section II, we review the related work on the recent success of deep neural networks in understanding natural images, as well as the use of neural networks in materials problems. In Section III, we present the problem of Electron backscatter diffraction (EBSD) indexing, which involves reading electron images to determine orientation of surfaces of polycrystals. In Section IV the proposed deep learning solution is discussed. And in Section V the data, implementation and experimental results are presented. We finally conclude the paper and discuss future work in Section VI.

## II. BACKGROUND

The past few years have witnessed a renewed interest in neural networks and backpropagation as a learning framework [8]. With the availability of large scale data and fast computational tools, the practice of training deep neural networks has been favored. Most significantly, convolutional neural networks (CNNs), a class of deep learning models designed to simulate the visual signal processing in central nervous systems, have gained much attention in the field of image understanding. These models usually consist of alternating combination of convolutional layers with trainable filters and local pooling layers, resulting in a complex hierarchical representations of inputs. When trained end-to-end from raw pixel values to classifier outputs, with millions of labeled images, they have achieved superior performance on many image-related tasks including object recognition, segmentation, detection and retrieval [9]–[11].

Deep learning [12] has undergone a fast development. Almost every a new extension to the conventional algorithm emerges (e.g. on arxiv.org) and a new application field becomes interested. Recent examples include the field of drug discovery making use of a multitask deep learning framework [13]. Similarly to drug discovery, the problem of materials discovery is about deciding on certain composition, formulation, and processing steps of materials with a desired target property [14]–[17].

Neural networks as a tool have been used in materials science applications such as spectroscopy classification and structural identification [18], characterizing constitutive relationship for alloys [19], the evaluation of descriptors [20], etc. However, neither the size of data or complexity of networks in these works have gone large enough.

CNNs used on scientific images from other fields have been seen. In the work of brain image segmentation [9], CNNs are used on electron microscopy image as a pixel classifier, outputting for each pixel, the probability of it being a membrane. A number of 10-layer CNNs are trained on a set of  $30\ 512 \times 512$  images, and the final system is an ensemble of them. Ranzato et al. [21] studied the recognition of biological particles from microscopy images. The dataset contains 27 classes each with 500 images, each with a dimension of  $52 \times 52$  pixels. Both works have CNN used for classification. However our work involves making regression inferences, posing more challenges to the deep network.

Another body of work targets at improving the interpretability of networks. The easiest way to probe what's learned by a network is to visualize the weights, activation, gradients from different parts of the network and at different times. More advanced tools are developed for visualizing the activations produced on each layer of a trained CNN, in a live streaming fashion, as well as features extracted at different layers throughout the network [22].

To this date, the actual collaboration of deep learning and materials discovery has been scarce, mainly due to three reasons:

- **The lack of big materials data.** More specifically, the lack of large-scale, high quality, cleaned and labeled data. Materials data simply were never big enough. Failed experimental data got tossed away easily, and those that were saved were not properly curated for future use. The collection of large-scale materials data and the popularization of accessible database projects have just started to play an impact, with couple of example sites including the Materials Project [23]<sup>1</sup>, and Open Quantum Materials Database (OQMD) [24]<sup>2</sup>.
- **The nontransparency of neural networks.** The difficulty of understanding the millions of weights that

<sup>1</sup><https://www.materialsproject.org>

<sup>2</sup><http://www.oqmd.org>

a normal-sized network can have prevents people in scientific discovery domain from adopting it.

- **The proper induction of domain knowledge.** ML by itself is largely agnostic. The supervised modeling from input to output often takes only the sheer closeness of predictions towards targets as the objective and attempts to achieve a lower and lower loss by optimization (e.g. stochastic gradient descent). This practice largely ignores the problem fundamentals and hence easily produces overfitted results.

The objective of this work is to bridge the gap between the two fields, by taking a first effort at addressing these impediments. Firstly, to the best of our knowledge, this work establishes the first microscopic image regressional modeling utilizing a dataset as large as 300K. Secondly, in addition to demonstrating the prediction accuracy, we make a first effort to visualize the process and result of learning. Thirdly, we attempt to induce domain knowledge into the loss function and regularization of learning, making the prediction result less prone to overfitting.

### III. PROBLEM: EBSD INDEXING

The problem involves the study of a ubiquitous materials type: polycrystals. Common examples including commercial metals, alloys and ceramic, polycrystals are solids composed of small crystallites forming an aggregate that appears homogeneous at the macroscopic scale (millimeter length scale). However, at the microscale (millimeter to a nanometer scale, observed using microscopy techniques), a polycrystal exhibits various structural characteristics, including grain orientations, local textures, phase distributions, etc. The study of those microstructures is essential to understand properties of a polycrystal.

Electron backscatter diffraction (EBSD) is a standard technique detecting certain microstructure characteristics on the surfaces of polycrystals. An example of EBSD experiment setup is depicted in Fig. 1(a). A carefully calculated volume of high-energy, high-speed electrons are discharged from a stationary beam towards a specimen. Due to the backscattering nature, the electrons would be reflected by the surface (more specifically, at a range of different depths below the surface) of a crystalline material and travel towards a phosphor screen detector. A diffracting scattering pattern is then captured from the screen. It is often exhibited as a collection of parallel and intersecting bright bands. Figure. 2 displays several examples of such pattern.

If we change the tilting or rotation of specimen and hence the orientation of the crystal lattice, the arrangement patterns would also change. In another word, each EBSD image is produced by a particular crystal orientation, commonly described by three Euler angles, denoted as  $(\varphi_1, \Phi, \varphi_2)$ . Their definition is illustrated in Fig. 1(b). They together represent how the specimen is tilted with respect to the three dimensional axes. The inverse problem that determines

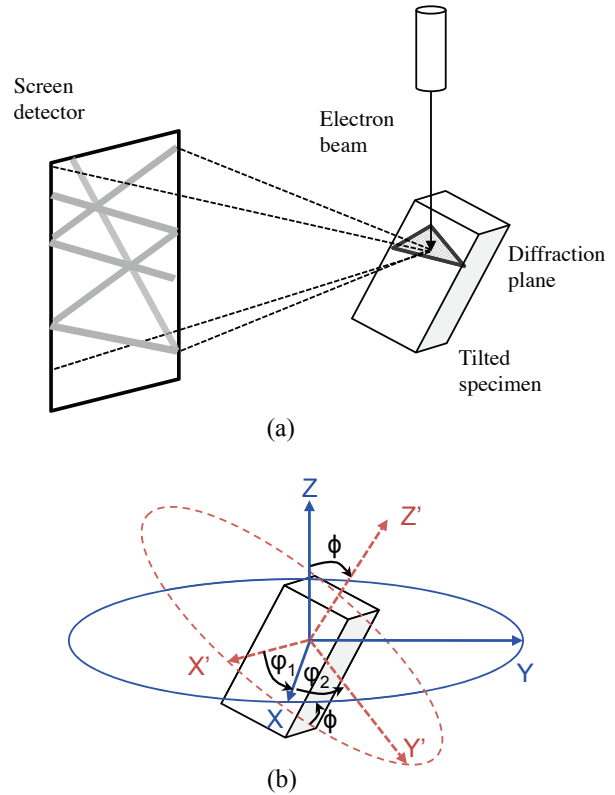


Figure 1: (a) Simplified schematic of EBSD generation: a beam of electrons gets reflected by the diffraction plane in the specimen, and captured on a screen. (b) Definition of Euler angles  $(\varphi_1, \Phi, \varphi_2)$ . Specimen is tilted so that the original axes  $(X, Y, Z)$  become  $(X', Y', Z')$ .

the orientation angles from examining an image, is called automatic EBSD indexing. Such an indexing is key to perform quantitative microstructure analysis for polycrystalline materials.

Traditional approach to EBSD indexing is pattern matching. It requires precomputing a database of (pattern, orientation) pairs, which stores the idealized patterns generated by distinct orientations. When a test pattern is observed, it is compared to each member in the database and the orientation of its 1-Nearest Neighbor (1-NN) is returned. The comparison between two patterns can be made either directly, that is, pixel-by-pixel, or after certain image processing procedures, for example, Hough transform, butterfly convolution, Gaussian filtering, binning, peak detection, etc. [25]. The preprocessing of EBSD images may be complicated but traditional indexing is just based on the idea of nearest neighbor lookup, or instance-based learning.

A well-known drawback of 1-NN is its high computational cost at prediction time, for distance calculation between the test sample and every training sample. Besides, inductive bias is easily introduced if similar images are assumed to exhibit similar crystalline orientations, without

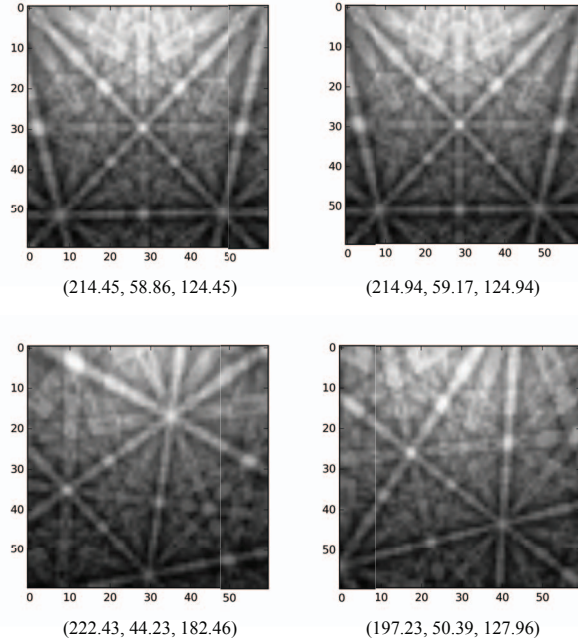


Figure 2: Examples of four EBSD patterns, each denoted with its corresponding Euler angles  $(\varphi_1, \Phi, \varphi_2)$ , used as regression target in deep net training. The upper left and upper right patterns are very similar, and also have a small difference in target angles.

proper regards to the uncertainty introduced by the quality of image, visibility of bands, lighting conditions, and so on.

Moreover, in light of a lightweight, straightforward learning algorithm as the 1-NN, the key to success relies largely on the extraction of features that describe the unique geometry of each EBSD pattern. Much effort is seen invested in devising templates [26] and other key descriptors for use in the matching of patterns. They often tend to examine the number, positions and widths of certain bands in the pattern, and create attributes that relate to known physics of electron scattering phenomena in the image formation process [27].

In contrast, we attempt to make use of deep learning models to address the problem of EBSD image indexing in an end-to-end fashion. We are interested in building learning models that do not require domain-specific knowledge or much image processing steps. That said, we feed raw pixel values as the input to a deep CNN architecture and design a special loss function to attend to the peculiar characteristic of orientation angles.

#### IV. DEEP LEARNING SOLUTION

Our solution is to construct a deep CNN that takes EBSD images as input, and produce three real-valued angles as output, through multiple convolutional layers that are used to take into consideration the spatial dependencies among image pixels, and fully connected layers for multi-layer regression.

The data is a set of grey scale images with input pixel values between 0 (black) and 255 (white). Each image is associated with three target values. The problem setup at a glance is quite similar to the MNIST digit classification [28]. The MNIST handwritten digits classification has been well-studied and used as a benchmark for the CNN development [29], [30]. The database contains 70,000  $28 \times 28$  pixel images divided into training and test sets. The similarity of our problem with MNIST is that it also processes grey scale images with one channel of input, whose pixel values are between 0 (black) and 255 (white), and associates each image with a one-valued target. However there are a number of aspects that make this problem distinctive, and much more difficult:

- The image size is much larger than MNIST, at  $60 \times 60$  (vs. MNIST of  $28 \times 28$ ).
- The data volume is much larger, of 300K (vs. MNIST of 70K).
- The digit patterns in MNIST have a roughly bilevel representation - white being the digits, black the background. The grey scaled values can be binarized without much loss of information. However, a backscatter pattern is the regular arrangement of parallel bright bands on a steep continuous background, leading to a much larger variation of inputs.
- The target is a real number (actually, three real numbers, but we choose to model them one at a time), making it a regression problem.
- The target variable is an angle; the periodicity of angular data has to be addressed.

##### A. Loss function

We separately model each Euler angle, resulting in three tasks. For each task, we use the same training and testing inputs but different target outputs. A special loss function is designed to account for the periodicity of angular data (the fact that  $0^\circ$  is close to  $359^\circ$ ) when measuring the difference between predicted outputs and the ground truth. The definition is as follows.

Suppose a training set of  $m$  samples is given as  $\{X_i, y_i^j\}_{i=1}^m$ , where  $X_i \in R^n$  denotes the  $i$ -th training sample, and  $y_i^j$  denotes the  $j$ -th output,  $j = 1, 2, 3$ , of the same training sample. For  $j = 1, 3$  the output  $y^j$  is and orientation angle between  $0^\circ$  to  $360^\circ$ . For  $j = 2$ ,  $y^j$  is bounded by a smaller range between  $0^\circ$  to  $60^\circ$ .

To quantitatively measure the difference between the predicted angle  $\hat{y}_i$  and ground truth  $y_i$ , while taking care of the periodicity of angular expressions, the loss function is designed in the following form. The  $j$  index in  $y$  is omitted as we use the same form across all three tasks. For each training sample  $i$ , the loss  $L_i$  given the predicted angle  $\hat{y}_i$  and ground truth  $y_i$  is:

$$L_i(y_i, \hat{y}_i) = \arccos(\cos(\|y_i - \hat{y}_i\|)) \quad (1)$$

Such a loss function converts any angular difference between  $[-360^\circ, 360^\circ]$  to  $[0^\circ, 180^\circ]$ , or  $[0, \pi]$  in radians. In CNN training, this customized loss function is used instead of the cross-entropy in classification.

### B. Architecture

**Convolutional layer.** A convolutional layer is parametrized by the number of channels, kernel size, stride factor, border mode, and the connection table. Each layer has  $M$  channels of equal size  $(M_x, M_y)$ . A kernel of size  $(K_x, K_y)$  is shifted over the effective region of the input image. The effective region is determined by the border mode factor: at “full” mode, at least one pixel row or column of the kernel has to be inside the image, and while outside the image is padded with 0s; at “valid” mode, the kernel has to be completely inside the image. The stride factors  $S_x$  and  $S_y$  define how many pixels the kernel skips in  $x$ - and  $y$ -direction between subsequent convolutions. This factor is normally 1 unless noted otherwise. The size of the output channel at layer  $l$ , indicated at the superscript, is then determined as:

$$\begin{aligned} \text{full: } M_*^l &= \frac{M_*^{l-1} + K_*^l}{S_*^l} - 1 \\ \text{valid: } M_*^l &= \frac{M_*^{l-1} - K_*^l}{S_*^l} + 1 \end{aligned} \quad (2)$$

The subscript  $*$  can be either  $x$  or  $y$ . Kernels of a given channel share their weights.

**Max-pooling layer.** Max-pooling has shown to lead to faster convergence, a selection of superior invariant features, better generalization, and enable position invariance [31]. The output of the max-pooling layer is given by the maximum activation over non-overlapping rectangular regions of size  $(P_x, P_y)$ , so that the image is down sampled by a factor of  $P_x$  and  $P_y$  along each direction.

**Regression layer.** It is common to use multiple fully-connected (FC) layers after several rounds of convolution. The resulting structure of the last convolutional layer is flattened before connecting to the following FC layer. For a regression network, the last layer is always fully connected, with one output unit for the regression target.

### C. Network configuration

Out of many network configurations that we experimented, we present the best one that is a 7-layer CNN, with 4 convolutional layers and 3 fully connected layers (including the last output layer). Table I lists the detailed configurations for each layer. Conv stands for convolutional layer and FC for fully-connected layer. Two Conv layers with the same number of channels are placed together, one with “full” mode and the other with “valid”, so that after the pair of layers the image size remains unchanged. Then a pooling is added and the size is halved.

Table I: Details of each layer in our CNN for EBSD indexing.

Layer	Type	Channel	Feature	Kernel	Pooling
0	Input	1	$60 \times 60$	N/A	N/A
1	Conv	32	$68 \times 68$	$9 \times 9$	N/A
2	Conv	32	$60 \times 60$	$9 \times 9$	$2 \times 2$
3	Conv	64	$30 \times 30$	$9 \times 9$	N/A
4	Conv	64	$30 \times 30$	$9 \times 9$	$2 \times 2$
5	FC	N/A	512	$1 \times 1$	N/A
6	FC	N/A	256	$1 \times 1$	N/A
7	Output	N/A	1	$1 \times 1$	N/A

We also insert 4 Dropout [32] modules at the following places to regularize: (1) between layer 2 and layer 3, with a dropout probability of 0.25; (2) between layer 4 and layer 5, probability 0.25; (2) between layer 5 and layer 6, probability 0.5; (2) between layer 6 and layer 7, probability 0.5.

In convolutional layers, kernels all convolute with a stride 1. Pooling layers are all max-pooling and are non-overlapping. The weights are initialized using a Gaussian distribution, with 0 mean and standard deviation depending on the number of inputs and outputs of its layer, a concept from [33]. In particular, in a layer where its number of inputs is  $f_{in}$  and number of outputs  $f_{out}$ , the standard deviation of the Gaussian is:  $\sqrt{\frac{2}{f_{in}+f_{out}}}$ .

All layers use ReLu [34] as the activation function, except the output layer uses linear, for regression purposes. The optimization algorithm used for CNN training is the stochastic gradient descent (SGD), with a minibatch of size 30, using momentum [35] of 0.9 and initial step size 0.001 which is halved every 30 epochs for about 10 times. Each epoch takes a fixed number of random training samples uniformly sampled across classes.

## V. EXPERIMENT RESULTS

The data are generated by a forward model that simulate the physical EBSD experiment, as described in [36], from Carnegie Mellon University. It is roughly the same data used as the “dictionary” in [37]. Each image contains  $60 \times 60$  pixels, and the total number of images in the dictionary is 333,227. A random 300,000 samples are used for training, and 30,000 for testing. The three target orientation angles are in a range of  $[0, 360]$ ,  $[0, 60]$ , and  $[0, 360]$ . To the best of our knowledge, the work conducted here is the first deep learning solution towards EBSD indexing using a dataset this large. Experiments are implemented using Theano, carried out on a single NVIDIA TITAN X GPU with 12GB of memory. There are over 8 million free parameters in our CNN. The training takes about a week for 300 epochs.

### A. Prediction results

The best network trained for the first angle has a normalized error rate of 0.007 on test data, after 300 epochs. That is a Mean Absolute Error (MAE) of  $2.5^\circ$  when predicting an angle between 0 and  $360^\circ$ . For the second angle the



difference is  $1.8^\circ$ , and for the third  $4.8^\circ$ . The three curves of test error movement over training epochs is shown in Fig. 3.

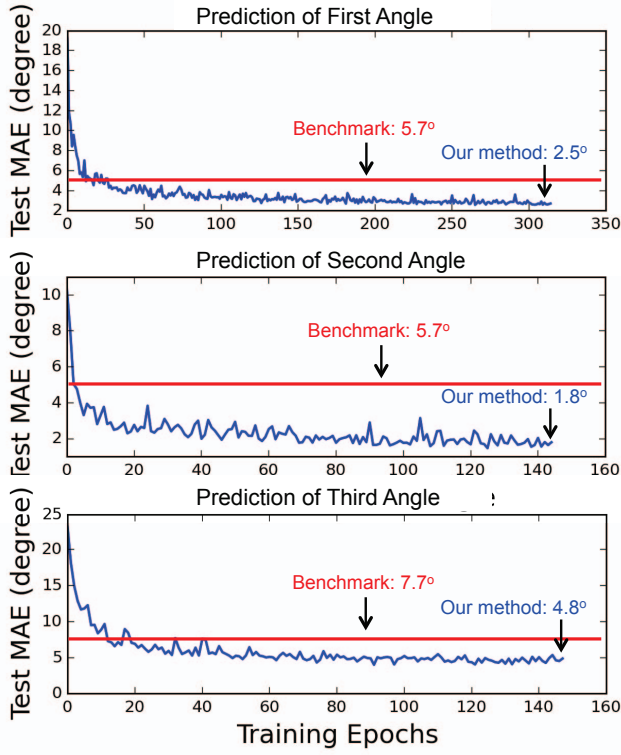


Figure 3: EBSD application: test MAE (unit: degrees) over training epochs, for each of the orientation angles in prediction.

The result can be compared with the state-of-the-art benchmark of a 1-NN method presented in [37]. We replicate the method using the same split of training and test as used in deep net training, and obtained MAEs of  $5.7^\circ$ ,  $5.7^\circ$ , and  $7.7^\circ$ , respectively for each angle. On an average we are 54% better than the benchmark method.

### B. Weight and activation visualization

In addition to their superior classification and regression performance, the interpretability of deep networks are becoming an appealing feature. Visualizing first-layer weights of a trained network has been a widely adopted practice to understand what has been learned by the network. Figure 4 shows the filters learned by the first Conv layer, which is directly looking at raw pixels.

The cleanness of features learned by the first Conv layer is an important indication of how well the network is trained. We can see all of the 32 filters in Fig. 4 are able to portray a clean feature. For example the first filter is clearly trying to capture a diagonal beam from upper left to lower right. The second filter is capturing a similar beam but of a opposite direction and a lower placement in the filter.

Another commonly practiced visualization technique is plotting and viewing activations of the network during the

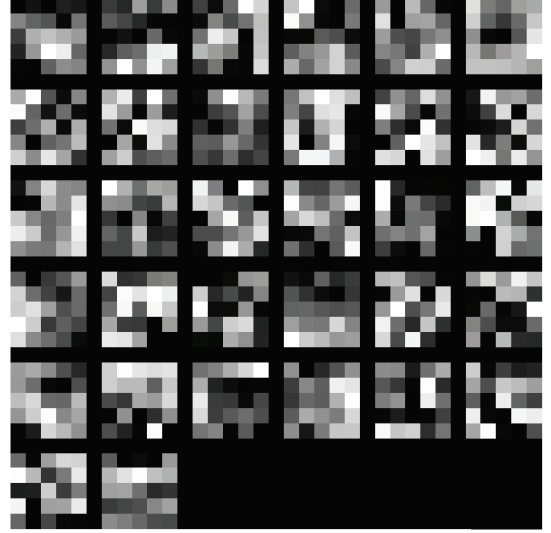


Figure 4: Filters learned by the first Conv layer in our CNN from EBSD images. A total of 32 square filters are shown due to the 32 channel size in Layer 1. We can see that different filters are capturing different features of the image. For example, in the first row, filter 1-3 are all displaying an edge with different angles and at different positions; filter 6 displays a centered knot.

forward pass for some given images. Figure 5 displays 16 original images with their first Conv layer activations. We can see how the first layer acts like a filter, removing most noisy information and extracting some part of the beams in image. Once beams are detected, later layers can use the information to figure out the calculation of orientation angles. Deeper layer activations can be visualized as well. Generally a deeper layer would produce a sparser activation pattern.

## VI. CONCLUSION

In this paper we demonstrate the use of deep learning as a step towards automated and accelerated materials discovery. Deep convolutional neural networks are used for the characterization of microstructures in electron microscope images, utilizing a dataset as large as 300,000 and generating crystalline orientations with a higher prediction accuracy than present state-of-the-art. We further visualize the weights and activations learned by the network, as a way to break the nontransparency that exists in most ML models, impeding domain scientists from widely adopting them.

The efforts made towards bridging the gap between materials science society and deep learning society are demonstrated in three-fold: (1) making use of big data that exist in image form; (2) incorporate domain knowledge into neural network loss function during modeling; and (3) visualizing various network characteristics as a first step towards probing the interpretability of the million scale weight space. We address the big data problem both in the

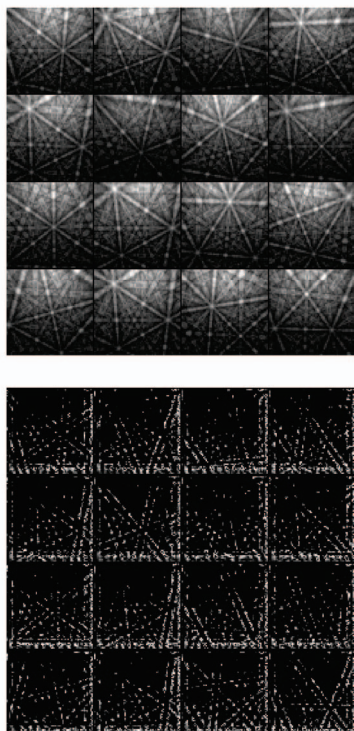


Figure 5: 16 EBSD images (top) with their activations (bottom) after the first Conv layer of our trained CNN. For better visibility the activation plotted are 10 times the actual value.

notion of making use of a large collection of image data, and the visualization of a large scale weight space.

As popular as deep learning has become these days, there is no free lunch. A single universal architecture, parameter set, loss function, training method or initialization cannot work for all kinds of problems. It can hardly be used as an off-the-shelf classifier. We managed to treat the application problems from an agnostic point of view, but there is still extensive data-driven exploration conducted to adapt the learning model to each given problem.

As databases containing various type of materials data are growing, being refined and becoming available, The application of machine learning techniques begins to gain great expectation for faster and smarter materials discovery. Similar development has been successful in biological science, drug discovery and healthcare, and is yet to be seen in the materials domain. Our demonstration in this work could play a role in encouraging the use of large datasets, efficient analytics, and advanced computational models for all kinds of applications in this field.

A more fundamental challenge persists, though, in the next step after an accurate model construction: how to distill information from learned model and extract crucial insights that relate structure to property; how to not only build robust predictive, quantitative models, but also interpret them with

domain knowledge, and systematically integrate them in the discovery, engineering process of materials.

As a future work, learning concepts such as transfer learning can be introduced. The idea is to transfer knowledge from other learning tasks, either from a different materials domain or from entirely outside of scientific images. The power of transfer learning lies in the ability of adopting knowledge acquired from other tasks, either closely related or loosely related, or even unrelated, with a lot of training samples. More and bigger data from diverse fields can therefore be joined to regularized the learning space of a given problem, and thus improve the performance.

#### ACKNOWLEDGEMENT

This work is supported in part by the following grants: AFOSR award FA9550-12-1-0458; NIST award 70NANB14H012; DARPA award N66001-15-C-4036; NSF awards CCF-1029166, IIS-1343639, CCF-1409601; DOE awards DE-SC0007456, DE-SC0014330.

#### REFERENCES

- [1] A. J. Hey, S. Tansley, K. M. Tolle *et al.*, *The fourth paradigm: data-intensive scientific discovery*. Microsoft research Redmond, WA, 2009, vol. 1.
- [2] A. Agrawal and A. Choudhary, "Perspective: Materials informatics and big data: Realization of the fourth paradigm of science in materials science," *APL Materials*, vol. 4, no. 5, p. 053208, 2016.
- [3] "Materials genome initiative for global competitiveness," [www.whitehouse.gov/mgi](http://www.whitehouse.gov/mgi), 2011, [Online; accessed 12-February-2016].
- [4] J. Allison, D. Backman, and L. Christodoulou, "Integrated computational materials engineering: A new paradigm for the global materials profession," *JOM: the journal of the Minerals, Metals & Materials Society*, vol. 58, no. 11, pp. 25–27, 2006.
- [5] P. Raccuglia, K. C. Elbert, P. D. Adler, C. Falk, M. B. Wenny, A. Mollo, M. Zeller, S. A. Friedler, J. Schrier, and A. J. Norquist, "Machine-learning-assisted materials discovery using failed experiments," *Nature*, vol. 533, no. 7601, pp. 73–76, 2016.
- [6] B. Goertzel and C. Pennachin, *Artificial general intelligence*. Springer, 2007, vol. 2.
- [7] S. R. Kalidindi and M. D. Graef, "Materials data science: Current status and future outlook," *Annual Review of Materials Research*, vol. 45, no. 1, pp. 171–193, 2015. [Online]. Available: <http://dx.doi.org/10.1146/annurev-matsci-070214-020844>
- [8] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems 2, NIPS 1989*. Morgan Kaufmann Publishers, 1990, pp. 396–404.

- [9] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Deep neural networks segment neuronal membranes in electron microscopy images," in *Advances in neural information processing systems*, 2012, pp. 2843–2851.
- [10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [11] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013.
- [12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [13] B. Ramsundar, S. Kearnes, P. Riley, D. Webster, D. Konerding, and V. Pande, "Massively multitask networks for drug discovery," *arXiv preprint arXiv:1502.02072*, 2015.
- [14] R. Liu, A. Kumar, Z. Chen, A. Agrawal, V. Sundararaghavan, and A. Choudhary, "A predictive machine learning approach for microstructure optimization and materials design," *Scientific reports*, vol. 5, 2015.
- [15] R. Liu, Y. C. Yabansu, A. Agrawal, S. R. Kalidindi, and A. N. Choudhary, "Machine learning approaches for elastic localization linkages in high-contrast composite materials," *Integrating Materials and Manufacturing Innovation*, vol. 4, no. 1, p. 1, 2015.
- [16] A. Agrawal, P. D. Deshpande, A. Cecen, G. P. Basavarsu, A. N. Choudhary, and S. R. Kalidindi, "Exploration of data science techniques to predict fatigue strength of steel from composition and processing parameters," *Integrating Materials and Manufacturing Innovation*, vol. 3, no. 1, pp. 1–19, 2014.
- [17] B. Meredig, A. Agrawal, S. Kirklin, J. E. Saal, J. Doak, A. Thompson, K. Zhang, A. Choudhary, and C. Wolverton, "Combinatorial screening for new materials in unconstrained composition space with machine learning," *Physical Review B*, vol. 89, no. 9, p. 094104, 2014.
- [18] B. G. Sumpter and D. W. Noid, "On the design, analysis, and characterization of materials using computational neural networks," *Annual Review of Materials Science*, vol. 26, no. 1, pp. 223–277, 1996.
- [19] Y. Sun, W. Zeng, Y. Zhao, Y. Qi, X. Ma, and Y. Han, "Development of constitutive relationship model of ti600 alloy using artificial neural network," *Computational Materials Science*, vol. 48, no. 3, pp. 686–691, 2010.
- [20] H. Bhadeshia, R. Dimitriu, S. Forsik, J. Pak, and J. Ryu, "Performance of neural networks in materials science," *Materials Science and Technology*, vol. 25, no. 4, pp. 504–510, 2009.
- [21] M. Ranzato, P. Taylor, J. House, R. Flagan, Y. LeCun, and P. Perona, "Automatic recognition of biological particles in microscopic images," *Pattern Recognition Letters*, vol. 28, no. 1, pp. 31–39, 2007.
- [22] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization," *arXiv preprint arXiv:1506.06579*, 2015.
- [23] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. a. Persson, "The Materials Project: A materials genome approach to accelerating materials innovation," *APL Materials*, vol. 1, no. 1, p. 011002, 2013. [Online]. Available: <http://link.aip.org/link/AMPADS/v1/i1/p011002/s1\&Agg=doi>
- [24] S. Kirklin, J. E. Saal, B. Meredig, A. Thompson, J. W. Doak, M. Aykol, S. Rühl, and C. Wolverton, "The open quantum materials database (oqmd): assessing the accuracy of dft formation energies," *npj Computational Materials*, vol. 1, p. 15010, 2015.
- [25] X. Tao and A. Eades, "Errors, artifacts, and improvements in ebcd processing and mapping," *Microscopy and Microanalysis*, vol. 11, no. 01, pp. 79–87, 2005.
- [26] E. Rauch and L. Dupuy, "Rapid spot diffraction patterns identification through template matching," *Archives of Metallurgy and Materials*, vol. 50, pp. 87–99, 2005.
- [27] A. J. Schwartz, M. Kumar, B. L. Adams, and D. P. Field, *Electron backscatter diffraction in materials science*. Springer, 2009, vol. 2.
- [28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [29] D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3642–3649.
- [30] M. Ranzato, C. Poultney, S. Chopra, and Y. LeCun, "Efficient learning of sparse representations with an energy-based model," in *Advances in neural information processing systems*, 2006, pp. 1137–1144.
- [31] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," in *Artificial Neural Networks–ICANN 2010*. Springer, 2010, pp. 92–101.
- [32] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.
- [33] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International conference on artificial intelligence and statistics*, 2010, pp. 249–256.
- [34] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.



- [35] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, “On the importance of initialization and momentum in deep learning,” in *Proceedings of the 30th international conference on machine learning (ICML-13)*, 2013, pp. 1139–1147.
- [36] P. G. Callahan and M. De Graef, “Dynamical electron backscatter diffraction patterns. part i: Pattern simulations,” *Microscopy and Microanalysis*, vol. 19, no. 05, pp. 1255–1265, 2013.
- [37] Y. H. Chen, S. U. Park, D. Wei, G. Newstadt, M. A. Jackson, J. P. Simmons, M. De Graef, and A. O. Hero, “A dictionary approach to electron backscatter diffraction indexing,” *Microscopy and Microanalysis*, vol. 21, no. 03, pp. 739–752, 2015.