

Materials Image Informatics Using Deep Learning

Ankit Agrawal, Kasthurirangan Gopalakrishnan, Alok Choudhary
*2145 Sheridan Rd, TECH building, EECS Department,
Northwestern University, Evanston, IL 60201
ankitag@eecs.northwestern.edu*

The growing application of data-driven analytics in materials science has led to the emergence and popularity of the relatively new field of materials informatics. Of the many types of available data in materials science, image data is quite common and heterogeneous in itself, thanks to the advances in various materials imaging techniques. Within the arena of data analytics techniques, deep learning has recently led to groundbreaking advances in numerous fields such as computer vision. In this chapter, we describe the basics of deep learning, its advantages, challenges, and illustrative applications on materials images at different length scales for the purpose of fast and accurate structure characterization. While it is possible to build an accurate deep learning model from scratch when big data is available, transfer learning is used for small datasets. Together, the advances in materials imaging and deep learning provide unprecedented opportunities for such *materials image informatics* to enable better and faster understanding of the structure-property relationships in materials science and engineering.

1. Introduction

We are currently in the midst of the big data revolution, with humongous volumes of a variety of data being generated at a staggering velocity. This has led to the emergence of the fourth paradigm of science, which is data-driven science, and is becoming increasingly popular in practically all fields of science, engineering, and commerce. Figure 1 depicts the four paradigms of science.¹ Building upon the big data created by the first three paradigms of science – which are experiment, theory, and simulation – the fourth paradigm utilizes scalable machine learning and data mining techniques like predictive analytics, data clustering, relationship mining, deep learning, etc. to extract actionable insights from such big data and inform decision making at various levels. For example, in a scientific setting, it can help determine what is the best experiment or simulation to do next; for a healthcare application, it can estimate patient-specific outcomes and make recommendations of the best course of treatment; for a business application, it can help devise advertisement strategies to maximize profit and take proactive measures to

prevent customer attrition, etc. Therefore, it is not surprisingly that scalable data-driven techniques^{2–21} have found numerous applications in several diverse fields such as business, marketing, and eCommerce,^{22–28} healthcare,^{29–40} climate science,^{41–47} bioinformatics,^{48–55} social media,^{56–64} materials science,^{65–79} and cosmology,^{80–86} amongst many others. In particular, over the last few years, deep learning⁸⁷ has become the data analytics technique of choice due to its groundbreaking success in several traditional artificial intelligence applications like computer vision⁸⁸ and speech recognition.⁸⁹

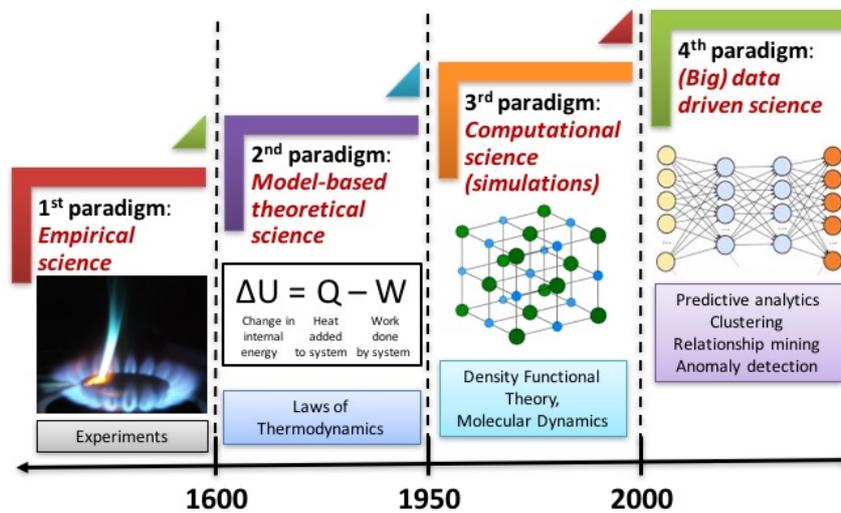


Fig. 1. (reproduced from reference¹ under CC-BY license) The four paradigms of science. For the most part of history, science was purely empirical or observational, which is known today as the experimental branch of science. The advent of calculus in the 17th century made it possible to express natural phenomena as mathematical laws, giving rise to the second paradigm of science, which is model-based theoretical science. The invention of computers in the 20th century allowed solving for much larger and complex theoretical models (system of equations), enabling simulations of several real-world phenomena, which is the third paradigm of science. The 21st century has seen an explosive growth of the data resulting from the first three paradigms, so much so that all the available historical data in itself has become a valuable resource for learning and enhancing our understanding of this world, heralding the arrival of the fourth paradigm of science, which is (big) data-driven science.

The field of materials science and engineering relies on experiments and simulations to better understand the so-called processing-structure-property-performance (PSPP) relationships. It is well-known that almost everything in materials science depends on the understanding of PSPP relationships, where the science relationships of cause and effect go from left to right, and the engineering relationships of goals and means go from right to left. The design of new materials with desired properties requires understanding this complex system of interrelated mechanisms in materials across numerous length and time scales. Figure 2 depicts these PSPP re-

relationships of materials science and engineering. Materials informatics,^{1,72,90} which is the realization of the fourth paradigm of science in materials science, can help generate fast and accurate forward models for predicting materials properties that can serve as cost-effective proxies to experiments and simulations. In turn, such fast forward models can also help realize inverse models for materials discovery and design, which are typically formulated as optimization problems. The importance for such data-driven informatics approaches in materials science has also been emphasized by the Materials Genome Initiative (MGI),^{91,92} which envisions the discovery, development, manufacturing, and deployment of advanced materials twice as fast and half the cost.

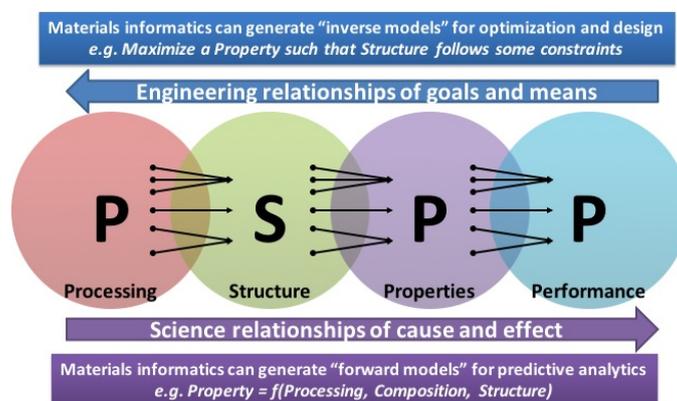


Fig. 2. (reproduced from reference¹ under CC-BY license) The processing-structure-property-performance relationships of materials science and engineering, where the deductive science relationships of cause and effect flow from left to right, and the inductive engineering relationships of goals and means flow from right to left. Further, it is important to note that each relationship from left to right is many-to-one. Materials informatics (data science) approaches can help decipher these relationships via forward and inverse models.

There are numerous kinds of materials data that record the equilibrium and temporal evolution of composition, processing, structure, properties, and application-specific performance metrics for a given material. Advances in materials imaging technologies at different time and length scales such as various types of microscopy, spectroscopy, and photography have made materials image data quite common, which are typically used to infer the materials' structure and subsequently understand structure-property relationships. In that sense, materials structure characterization can be considered as an inverse problem, since structure is the cause and image is the effect, and the characterization problem is to deduce the structure that produced the given image.

In this chapter, we shall take a look at some recent advances in materials image informatics at different length scales using deep learning techniques, after a brief introduction to deep learning and convolutional neural networks (CNNs), which are

a class of deep learning networks specifically used on image datasets. In particular, we will take the example of indexing electron backscatter diffraction (EBSD) images,⁹³ crack detection in pavement images,⁹⁴ and structural health monitoring on infrastructure images taken from an unmanned aerial vehicle (UAV).⁹⁵ The rest of this chapter is organized as follows: Section 2 introduces the basic concepts in deep learning, followed by illustrative materials image informatics in Section 3. Section 4 summarizes and concludes the chapter.

2. Deep Learning

Deep learning is a recent revolutionary breakthrough in machine learning (ML), arisen from a rediscovery of deep neural networks, and fueled by the availability of two key ingredients – big data and big compute – that are becoming increasingly available and affordable over the past few years. It is considered a very powerful method to exploit the information locked in big data. Deep learning has enabled ground-breaking advances in various fields, such as computer vision,^{96,97} natural language processing,^{98,99} and speech recognition.^{89,100} It is well-known in the field of ML that the way data is represented for input to a ML algorithm has a huge influence on the success of the model. Deep learning enables automated learning of multiple levels of representation, discovering more abstract features at higher levels.

2.1. Artificial Neural Networks

An artificial neural network (ANN) is a computing system inspired by the biological neural networks in animal brains. The fundamental unit of ANNs is called a neuron, which can take in multiple inputs and output a non-linear function (called the activation function) of the weighted sum of inputs. Commonly used activation functions are the sigmoid function, tanh function, and the rectified linear unit (ReLU). An ANN consists of multiple layers of such neurons with interconnections amongst them, such that the output of a neuron becomes one of the inputs to the neurons in the next layer. The inputs to the network constituting the input layer therefore go through a series of hidden layers before giving out the output(s) in the output layer. The number and depth of hidden layers and the way the neurons are connected to each other determines the architecture of the network. Figure 3 depicts a fully-connected ANN, also known as multilayer perceptron (MLP). For a given ANN architecture, everything depends on the strength or weights of the interconnections, which are adjusted or learned during the ANN training process by minimizing the disagreement (technically called the loss function) between ANN-predicted outputs and the ground truth values from a labeled training dataset. A deep learning network is essentially an ANN with more than one hidden layer.

2.2. Advantages

One of the primary advantages of deep learning is that it is feature engineering free, i.e., instead of providing the model with appropriate features (which are usually domain dependent and often require manual effort and intuition), deep learning is capable of automatically extracting relevant features from the data in the first hidden layer, and features-of-features in the second hidden layer, and so on, thereby resulting in an automated hierarchical representation of features. This is exactly why we often need “deep” networks, i.e., consisting of multiple hidden layers. More number of hidden layers means more learning capability at the expense of higher computational cost for training the model.

As one can imagine, more data is always helpful to build a more accurate ML model, and that is indeed the general trend with all supervised ML algorithms. This is where the other advantage of deep learning models comes in. While the accuracy performance of most traditional ML algorithms saturate at a certain point with increasing data sizes, deep learning models, although less accurate than traditional ML algorithms for small data, do not saturate that early and continue to become more and more accurate with increasing data. Therefore, there is usually a cross-over point in terms of data size at which the performance of deep learning models overtakes that of traditional ML algorithms, which can be different for different problems. In other words, deep learning is more powerful and suitable to build highly accurate models on big data. The question is, whether or not enough data is available for a given problem to build more accurate deep learning models.

2.3. Challenges

If deep learning has all these advantages, why is it not commonplace yet? Well, it is increasingly being used in many different fields, but there are still some challenges that need to be addressed and are important open research problems. One of the

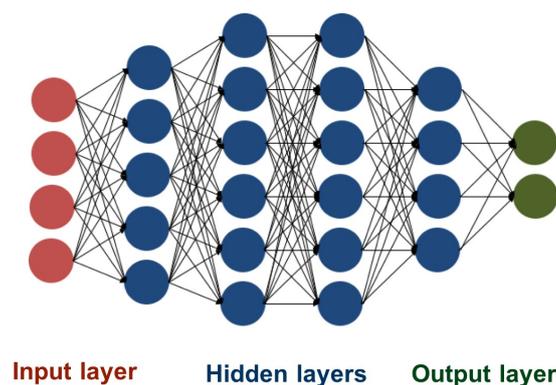


Fig. 3. The basic architecture of a fully-connected deep artificial neural network with four hidden layers.

biggest obstacle is the (non-)availability of enough data in many cases. While there exists big, curated, and labeled datasets to build deep learning networks for several problems like image classification and speech recognition, it is far less common for most of the scientific and engineering problems. But with a concerted effort to standardize the data collection and sharing protocols in many fields of science and engineering, this challenge is expected to be addressed in time. There is also a recent surge in the use of transfer learning in cases where big data is not available (more on that later).

Another big challenge in deep learning is the huge computational cost for training these deep neural networks. Even shallow neural networks are relatively slow to train compared to traditional ML techniques, but with deeper networks and big data, the training time can become too large for some problems, even on the latest computing hardware. An active field of research is parallelization of neural network training algorithms, and while some works in that direction have recently sprung up,^{101,102} it is an actively pursued research problem in the field of high performance computing.

The third challenge relates to the fact that there are significantly more parameters in a deep learning model compared to traditional ML algorithms, beginning with the architecture of the network (number of layers, number of neurons in every layer, and structure of the interconnections), and of course the weights of the interconnections which are learned during training. Therefore, the space of possible network architectures is practically infinite, and there is no systematic procedure to determine the best architecture for a given problem, although there are general guidelines based on prior architectures that have worked. Designing such a systematic methodology for optimal network architecture identification is very much an open research problem.

2.4. Convolutional Neural Networks

Convolutional neural network (CNN) is one of the popular deep learning architectures used for image data. As we know, an image is a 2D matrix of pixels representing intensity or color. Each pixel can thus potentially be treated as a feature, making the number of input features per image tremendously large. If a fully connected ANN were to be used for this, the network would grow too large and its training prohibitively expensive. Moreover, such an approach would also ignore the ordering of the pixels, thereby in-effect ignoring the spatial correlation information in the images, which is undesirable.

CNNs are specifically designed to capture locally correlated features present in images using convolution layers, which consist of multiple kernels or filters consisting of a small set of parameters. Each filter is applied to the input image as a sliding window to get more abstract features (called as feature maps) for next layer computation. In this way, the number of weights/parameters needed is significantly less compared to a fully connected ANN. Further in order for the feature maps to

not grow too large, pooling layers are used after one or several convolution layers to reduce the dimensionality of the feature maps, in-effect also reducing the required number of parameters. Finally, the outputs of stacks of convolution layers and pooling layers are flattened to a long one-dimensional vector, which is fed into one or more fully connected layers to produce the final prediction. Depending on whether it is a classification or regression problem, the last layer generates a probability distribution or a single numerical value, controlled by the number of neurons in the last layer and the activation function (softmax and linear respectively for classification and regression problems). Figure 4 depicts a CNN with the three types of layers.

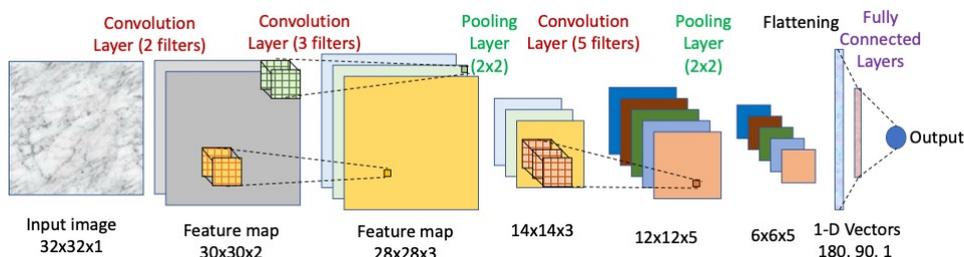


Fig. 4. An example architecture of a convolutional neural network (CNN). It takes a 32×32 grey scale image (i.e., one channel only, in contrast with a RGB image with three channels) as input. It then has two convolution layers with 2 and 3 filters respectively, thereby producing feature maps with 2 and 3 channels respectively. Two of the three filters in the second convolution layer are represented by orange and green (in this case, each of size $3 \times 3 \times 2$, i.e. 18 weights), and each filter produces one channel each for input to the next layer, as depicted by the color of the channel. The two convolution layers are followed by a 2×2 pooling layer, which reduces the dimensionality of the feature maps by applying a non-linear function (such as max, min, avg) to every 2×2 pixel matrix in the source layer and producing a single value for the destination layer, thereby reducing the dimensionality of the feature maps from 28×28 to 14×14 . This is followed by another convolution layer of 5 filters (each of size $3 \times 3 \times 3$, i.e. 27 weights) producing a feature map of 5 channels (one of the five filters and corresponding channel produced depicted in red), and another pooling layer to reduce feature map dimension from 12×12 to 6×6 . Subsequently, these feature maps are flattened to produce a one-dimensional vector of $6 \times 6 \times 5 = 180$ values, which is fed into two fully connected layers of 90 neurons and 1 neuron, which finally produces the output value. Note that each convolution layer produces a feature map with the size reduced by 2 pixels each in length and width compared to the source feature map, since all filters are of size 3×3 and chosen not to operate on boundary rows and columns. Appropriate zero padding can be used around the input image and feature maps before applying convolution if the size of resulting feature maps is intended to increase or remain the same.

2.5. Transfer Learning

As we saw earlier, deep learning only works well with big data. It has however been shown that it is possible to circumvent that problem in many cases by using a transfer learning approach.¹⁰³ As the name suggests, for solving a given problem, it tries to reuse the knowledge from a previous model that was developed for a

different but related problem, thereby transferring the knowledge from the previous model to the new model. In the context of deep learning, the way it is typically used is that the previous model (neural network) is for a problem for which big data is available, and therefore that network is used as a starting point for developing the new model for the given problem with small data. This can be done either by using the weights of the previous model as initial weights for the new model, or using the previous model to derive feature representations of the small data which in turn are used as features for any traditional ML or deep learning algorithms.

One of the popular class of pre-trained deep learning network is called VGG, developed by the Visual Geometry Group at the University of Oxford. A specific VGG model architecture called VGG16 consists of 5 convolution blocks consisting of 2, 2, 3, 3, and 3 convolution layers respectively with a small filter size (3x3), and 5 maxpooling layers of size 2x2 for spatial pooling (one pooling layer after each convolution block). The output of the last pooling layer is flattened and connected to 3 fully-connected layers, with the final layer as the soft-max layer to do classification. Therefore, it has 16 weight layers (13 convolution layers and 3 fully-connected layers), hence the name VGG16. Rectified Linear Unit (ReLU) activation is applied to all hidden layers. The model also uses dropout regularization in the fully-connected layers. In total, VGG16 has about 144 million parameters. There are several popular CNN architectures¹⁰⁴ such as VGG, AlexNet, GoogLeNet, ResNet,⁹⁷ and pre-training these on large-scale annotated natural image datasets (such as ImageNet) have been shown to be very useful for solving cross domain image classification problems through the concept of transfer learning and fine-tuning.¹⁰⁵

3. Illustrative Materials Image Informatics

In this section, we shall see examples of deep learning networks (CNNs) on materials images. The first example is an illustration of training a CNN from scratch, and the other two are examples of using the transfer learning approach.

3.1. Indexing EBSD Patterns

Electron backscatter diffraction (EBSD) is one of the materials imaging techniques to study the microstructure characteristics, specifically crystal orientation. A high energy electron beam is fired towards a specimen, and the electrons get reflected by the surface at different depths and crystal orientations. The diffraction pattern is recorded at a phosphor screen detector, which are usually in the form of bright parallel and intersecting grey scale bands. As the specimen is tilted or moved, the pattern of bands changes since the orientation of crystal lattice changes. Each EBSD image is produced by a specific crystal orientation, which can be represented by the Euler angles triplet $\langle \phi_1, \Phi, \phi_2 \rangle$. The inverse problem of determining the orientation angles that must have produced a given image is called EBSD indexing, which is the key to performing quantitative microstructure analysis for polycrystals

such as commercial metals, alloys and ceramic, etc.

The most common approach for EBSD indexing is the Hough transform based method, which works by computing the angles between linear features extracted from the diffraction pattern using Hough transform. The main problem with this method is that its performance quickly deteriorates with noise. A more recent method called dictionary based indexing adopts a nearest neighbor search (1-NN) approach¹⁰⁶ where the output angles correspond to the orientation angles of the closest EBSD pattern present in the dictionary. The distance function between two EBSD patterns is simply the dot product of the corresponding one-dimensional pixel vectors. It has been found to be very robust to noise and outperforms the line feature based Hough transform approach. However, it is computationally very expensive, since one needs to compare the given EBSD pattern against every EBSD pattern present in the dictionary.

Liu et al.⁹³ presented the first deep learning solution for EBSD indexing problem, with the aim of providing an end-to-end solution that does not require any domain-specific knowledge or much image processing steps. The developed model takes raw EBSD patterns and produces three numerical angles as output through multiple convolution layers that are intended to automatically learn the spatial dependencies among the pixels and extract relevant features, pooling layers to reduce dimensionality of the intermediate feature maps, and fully connected layers at the end for multi-layer regression. The architecture of the CNN proposed in that work is depicted in Fig. 5.

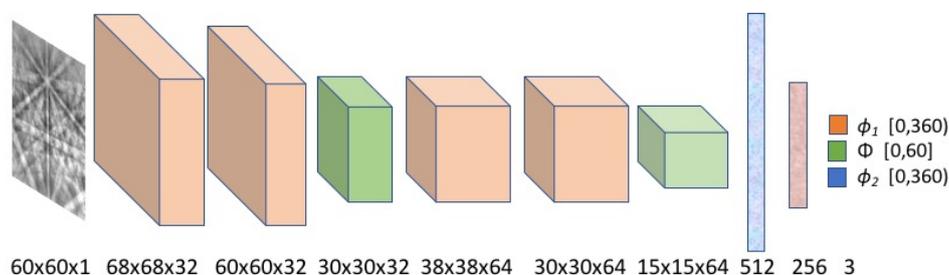


Fig. 5. CNN architecture for EBSD indexing. It has four convolution layers (resulting feature maps shown in brown), two pooling layers (resulting feature maps shown in green), and three fully connected layers including the last (output) layer. A 9x9 filter was used for the convolution layers with different thickness of zero padding around the input image or feature maps, therefore the dimensions of the resulting feature maps change by +8 or -8 accordingly.

The CNN model developed by Liu et al.⁹³ was trained using EBSD patterns (simulation data) of polycrystalline Nickel, with 333,227 simulated patterns, where each pattern is a 60x60 gray scale image. This data was randomly split into training and testing datasets, with 300,000 images used for training and rest for testing. Since the target output are the angles which are numbers, this is a regression problem. Usually, the loss function used for regression is the mean square error between

the predicted value and actual value. However, in this case, the numbers to be predicted are angles, which are periodic, i.e., 1° is very close to 359° . Therefore, a special loss function was designed to take into account this periodicity:

$$L(\phi, \hat{\phi}) = \arccos(\cos(|\phi - \hat{\phi}|))$$

where ϕ is the actual angle and $\hat{\phi}$ is the predicted angle.

Table 1 shows the accuracy and timing comparison of the deep learning (CNN) model⁹³ and dictionary-based approach.¹⁰⁶ For all the three Euler angles, deep learning gives significantly more accurate predictions (lower mean absolute error). The deep learning model takes a week to train, whereas the dictionary or 1-nearest neighbor approach requires no training, as the dictionary itself is the model. However, it takes longer for testing since each test EBSD pattern needs to be compared against all the patterns in the dictionary to find the most similar pattern. The deep learning model runs about an order of magnitude faster than the dictionary-based approach to make predictions of the three Euler angles. On average the deep learning model gave 56% more accurate and 86% faster predictions compared to state-of-the-art dictionary-based approach.

Table 1. Accuracy and timing comparison of deep learning model and dictionary approach for EBSD indexing

Predictor	Mean Absolute Error (degrees)	Training Time	Testing Time
Dictionary (1-NN)	5.7, 5.7, 7.7	0	375s
Deep Learning (CNN)	2.5, 1.8, 4.8	7 days	50s

3.2. Pavement Crack Detection

Roads or pavements are complex material composites designed to sustain repeated vehicular traffic. They typically comprise a surface layer, a base and/or subbase layer and a subgrade layer. The surface layer of a pavement structure is typically a Portland Cement Concrete (PCC) composite or Bituminous/Asphalt Concrete (AC), a particular pitch-matrix composite. This leads to two broad pavement categories: AC-surfaced and PCC-surfaced pavement composite with sizes of material constituents ranging from nano to micro. The base/subbase layer, which provides mechanical stability in addition to drainage, is typically comprised of crushed granular aggregates. The subgrade layer is generally the roadbed or natural soil. Figure 6 shows an example of PCC and AC surfaced pavement images. Apart from repeated traffic loading, diurnal temperature changes, seasonal variations, and other climatic factors combine forces to gradually weaken the pavement structural load-carrying capacity. This typically manifests in the form of surface distresses like cracking, rutting, potholes, etc. Early detection of these distresses by transportation agencies

can help prioritize maintenance and repair activities resulting in significant cost savings when one considers the life-cycle maintenance costs. Over the years, significant progress has been made towards automated pavement distress evaluation.¹⁰⁷

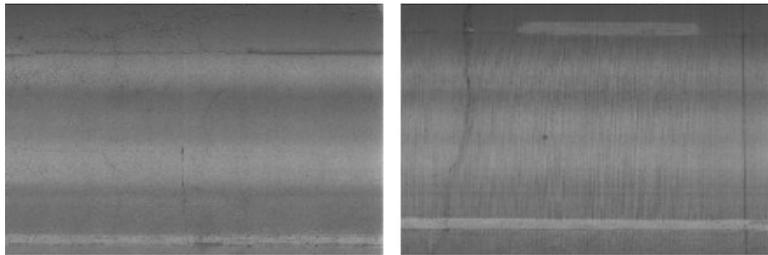


Fig. 6. Sample digital pavement images used in automated pavement distress detection: AC-surfaced pavement image (left); PCC-surfaced pavement image (right) (Source: FHWA/LTPP public database)

Since cracks are the most common distresses appearing on pavement surfaces and are relatively easier to detect and quantify using 2-D pavement images, a number of studies over the past three decades have focused on developing automated vision-based pavement crack detection algorithms and methods with an attempt to detect, classify (longitudinal/transverse/alligator), and characterize (length, width, and severity) the cracks. These methods could broadly be classified as: intensity-thresholding,¹⁰⁸ edge detection,¹⁰⁹ wavelet transforms,¹¹⁰ texture-analysis, and computer vision techniques.¹¹¹ This topic continues to be actively researched by both pavement engineers and computer vision researchers owing to the complex challenges associated with digital pavement image analysis such as inhomogeneity of cracks, diversity of surface texture (eg., tining), background complexity, presence of non-crack features such as joints, lane markings, patches, etc.

With the growing popularity of deep learning, several recent works have employed Convolutional Neural Networks (CNNs) for automated pavement distress detection. In what appears to be one of the first works on the direct application of CNNs to road crack detection, reference¹¹² classified square image patches with or without cracks based on 500 pavement images of size 3264 x 2448 collected using a low-cost smart phone. CrackNet¹¹³ presents an efficient CNN for pixel-perfect crack detection on 3-D asphalt pavement surfaces. Most recently, reference¹¹⁴ employed an end-to-end DL-based state-of-the-art object detection method for detecting and distinguishing objects in road images, acquired using a smartphone installed in a moving car under realistic weather and lighting conditions, into eight different output categories.

Inspired by the success of transfer learning applications in medical image analysis, Gopalakrishnan et al.⁹⁴ proposed the use of a pre-trained CNN model with transfer learning for automated pavement crack detection. A subset of the pavement distress images dataset from the Federal Highway Administration's (FHWA) Long-

Term Pavement Performance (LTPP) program was used, which contains research quality pavement performance information for over 2,500 test sections on in-service highway pavements located throughout the United States and Canada.¹¹⁵ The selected images were manually labeled to fall into two separate categories: “crack” or “no crack”. Both AC- and PCC-surfaced images were combined to introduce higher order complexity. Among the 1,056 pavement images, 337 images had low, medium, or high-severity cracks and 719 images had no cracks. Both longitudinal and transverse cracks were considered although transverse cracks were more common.

The Keras deep learning framework¹¹⁶ was used by Gopalakrishnan et al.,⁹⁴ that includes pre-trained deep learning models. Specifically, the Keras implementation of the VGG16 model with weights pre-trained on ImageNet database was used. Upon removing the last block of fully-connected layers from the VGG16 network, it was used as a deep feature generator for producing semantic image vectors for pavement images. These vectors were then used with several classifiers like Neural Networks (NN), Support Vector Machine (SVM), Random Forest (RF), etc. for predicting the labels. Figure 7 illustrates the workflow of the transfer learning approach used by Gopalakrishnan et al.⁹⁴

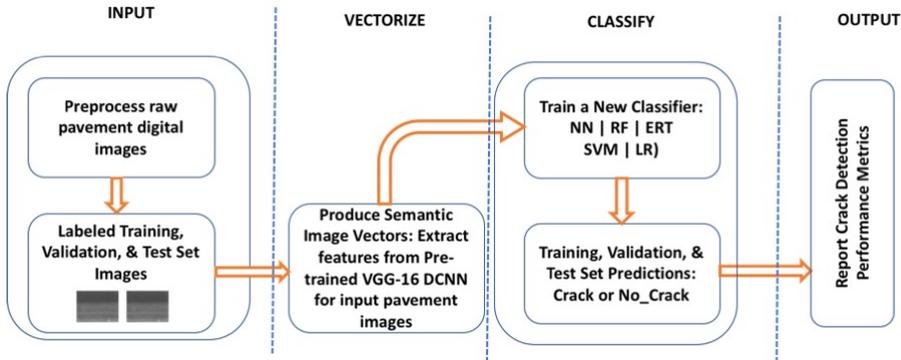


Fig. 7. Overall framework of the deep transfer learning approach for automated pavement crack detection

A comparison of pavement crack detection results on test set images (20%) using various deep transfer learning models is presented in Table 2. Since the image features for all the models were generated using truncated VGG16 model, the final classifier specifies the different models. F1-score is the harmonic mean of precision and recall. AUC is the area under the ROC curve. Among the different classifiers tested, a single-layer neural network classifier (with ‘adam’ optimizer) trained on ImageNet pre-trained VGG16 CNN features yielded the best performance (90% accuracy, precision, recall, F1-score, and 0.87 AUC). Thus, deep transfer learning can be a cost-effective and efficient approach for rapid and automated image classification tasks, especially when the available data is small.

Table 2. Comparison of pavement crack detection results using various deep transfer learning models

Transfer Learning	Final Classifier	Accuracy	F1-score	AUC
Truncated VGG16	Single-layer Neural Network (NN)	0.90	0.90	0.87
Truncated VGG16	Random Forest (RF)	0.86	0.85	0.78
Truncated VGG16	Extremely Randomized Trees (ERT)	0.87	0.86	0.78
Truncated VGG16	Support Vector Machine (SVM)	0.87	0.87	0.80
Truncated VGG16	Logistic Regression (LR)	0.88	0.87	0.79

3.3. Structural Health Monitoring

In recent years, Unmanned Aerial Vehicles (UAVs) or Unmanned Aerial Systems (UAS), commonly referred to as drones, are fast emerging as easy-to-deploy remote sensing and monitoring systems with a variety of applications ranging from disaster management and construction surveying to civil infrastructure inspection.¹¹⁷ Owing to their ability to accommodate a range of payloads with varying sizes and weights, including high-definition (HD) camera, multi-spectral cameras, laser scanners, sensors, etc., they are already widely used for search-and-rescue operations, emergency operations, remote sensing, photogrammetry, precision agriculture, surveillance, marketing, real estate, meteorology, etc.

An attractive feature of using UAVs for infrastructure inspection is that they provide the accessibility and flexibility to navigate around complex structures and locations that are hard to reach otherwise and collect data with equal or higher quality compared to manual data collection by inspectors. This makes it significantly easier, safer, and cost-effective compared to traditional inspection methods in the context of structural health monitoring (SHM) of civil infrastructure systems such as bridges, buildings, roads, power lines, pipelines, wind turbines, etc. Also, UAV-based visual inspection methods can be used for monitoring both the microscale and macroscale defects of infrastructures resulting in detailed inventory, survey, full field mapping, and condition assessment of civil and transportation infrastructure systems.¹¹⁸

The valuable data such as HD images obtained using UAVs can be used to train machine learning and deep learning algorithms to make the process of identifying defects on the infrastructure from images efficient. Although machine learning based techniques have been successfully applied to object detection in UAV images in the context of SHM, the use of DL for prognostics and damage assessment of civil infrastructure systems is still an emerging area of research, mainly due to the difficulties in generating good quality labeled training image datasets.¹¹⁹ A comprehensive review on the reported uses and applications of DL for UAVs, including the major challenges for the application of DL for UAV-based solutions by Carrio et al.¹²⁰ found that the most successful applications of CNNs to UAVs were with respect to visual data (images) owing to the low-cost, lightweight, and low power consumption of image sensors.

Gopalakrishnan et al.⁹⁵ recently developed a simplified crack detection model from UAV images of civil infrastructure through the application of deep transfer learning approach using a similar approach that was used for automated pavement crack detection.⁹⁴ A Hexacopter UAV (see Figure 8) was used to collect close-up images of few common civil infrastructure systems (like storage silos, local roadways, etc.). More specifically, the HD visual inspection was carried out using the Hexacopter-I UAV with a 30 MP high definition Canon EOS 5D Mk IV DSLR camera mounted on a 3-axis rotatable gimbal with live video transmission, which allows the inspector to change the direction of the camera and focus to get better pictures of the defects on the structure. Some sample HD images of civil infrastructure components with and without cracks captured by Hexacopter are shown in Figure 9.



Fig. 8. A snapshot of Hexacopter-I UAV used for civil infrastructure inspection (Source: InfraDrone LLC)



Fig. 9. Sample UAV images of civil infrastructure systems: with cracks (left) and without cracks (right) (Source: InfraDrone LLC)

For conducting DL experiments, the Keras DL framework was used and the Keras implementation of the VGG-16 model with weights pre-trained on ImageNet

database was used in this study. The dataset consisted of 130 UAV images, of which 80 were labeled as “crack” and 50 as “no crack”. 80% data was used for training and the rest 20% for testing. Among the 80% of the images dataset used for training and validation, 10% was used for validation alone. The early stopping criteria was used with the final model being the one with low validation loss. The transfer learning workflow employed for this problem was similar to the previous example, hence the details are not repeated here. A comparison of crack detection results on UAV images of civil infrastructure components using various deep transfer learning models is presented in Table 3. The results show that the proposed UAV vision-based deep transfer learning crack detection method can achieve about 90% accuracy in finding cracks in realistic situations without any augmentation and preprocessing. Most importantly, it shows that the transfer learning approach can work well even with very small datasets in some cases.

Table 3. Crack detection performance on UAV test set images of civil infrastructure using various deep transfer learning models

Transfer Learning	Final Classifier	Accuracy	F1-score	AUC
Truncated VGG16	Single-layer Neural Network (NN)	0.89	0.89	0.90
Truncated VGG16	Random Forest (RF)	0.86	0.85	0.77
Truncated VGG16	Extremely Randomized Trees (ERT)	0.82	0.82	0.83
Truncated VGG16	Support Vector Machine (SVM)	0.75	0.74	0.83
Truncated VGG16	Logistic Regression (LR)	0.89	0.89	0.90

4. Summary

Materials informatics is an emerging field at the intersection of materials science and computer science. In particular, data-driven techniques in computer science are increasing being applied on a variety of materials data with great success, and in this chapter we discussed a few recent advances in the application of deep learning techniques on materials image data at different length scales (materials image informatics). The fundamental concepts of deep learning, its advantages, challenges, and some ways to address those challenges were also discussed. The increasingly availability of materials databases (both images and other data types) along with groundbreaking advances in data science approaches (in particular deep learning based techniques) offer a lot of promise to successfully realize the goals of the Materials Genome Initiative, and aid in the discovery, design, and deployment of next-generation materials.

Acknowledgments

The authors would like to acknowledge support from the following grants: AFOSR award FA9550-12-1-0458; NIST awards 70NANB14H012, 70NANB19H005; NSF

award CCF-1409601; DOE awards DE-SC0019358, DE-SC0014330. The authors would also like to thank Prof. Marc De Graef from Carnegie Mellon University for providing EBSD images, the Federal Highway Administration's (FHWA's) Long-Term Pavement Performance (LTPP) program for providing pavement images, and Infradrone LLC for providing UAV images of civil infrastructure, used for the works described in this chapter.

References

1. A. Agrawal and A. Choudhary, Perspective: Materials informatics and big data: Realization of the "fourth paradigm" of science in materials science, *APL Materials*. 4(053208), 1–10, (2016).
2. C. Bohm and F. Krebs. High performance data mining using the nearest neighbor join. In *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, pp. 43–50, (2002). doi: 10.1109/ICDM.2002.1183884.
3. V. Kumar, M. V. Joshi, E.-H. S. Han, P.-N. Tan, and M. Steinbach. High performance data mining. In *High Performance Computing for Computational Science?VECPAR 2002*, pp. 111–125. Springer, (2003).
4. R. Grossman and Y. Gu. Data mining using high performance data clouds: Experimental studies using sector and sphere. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '08*, pp. 920–927, New York, NY, USA, (2008). ACM. ISBN 978-1-60558-193-4. doi: 10.1145/1401890.1402000. URL <http://doi.acm.org/10.1145/1401890.1402000>.
5. Y. Bengio, Learning deep architectures for ai, *Foundations and trends® in Machine Learning*. 2(1), 1–127, (2009).
6. W. Hendrix, M. M. A. Patwary, A. Agrawal, W.-k. Liao, and A. Choudhary. Parallel hierarchical clustering on shared memory platforms. In *Proceedings of the 19th International Conference on High Performance Computing (HiPC)*, pp. 1–9, (2012).
7. M. Patwary, D. Palsetia, A. Agrawal, W.-k. Liao, F. Manne, and A. Choudhary. A new scalable parallel dbSCAN algorithm using the disjoint-set data structure. In *ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*, pp. 1–11. IEEE, (2012). Article No. 62.
8. W. Fan and A. Bifet, Mining big data: current status, and forecast to the future, *ACM SIGKDD Explorations Newsletter*. 14(2), 1–5, (2013).
9. A. Agrawal, M. Patwary, W. Hendrix, W.-k. Liao, and A. Choudhary, *High performance big data clustering*, In ed. L. Grandinetti, *Advances in Parallel Computing, Volume 23: Cloud Computing and Big Data*, pp. 192–211. IOS Press, (2013).
10. M. Patwary, D. Palsetia, A. Agrawal, W.-k. Liao, F. Manne, and A. Choudhary. Scalable parallel optics data clustering using graph algorithmic techniques. In *Proceedings of 25th International Conference on High Performance Computing, Networking, Storage and Analysis (Supercomputing, SC'13)*, pp. 1–12, (2013). Article No. 49.
11. W. Hendrix, D. Palsetia, M. M. A. Patwary, A. Agrawal, W.-k. Liao, and A. Choudhary. A scalable algorithm for single-linkage hierarchical clustering on distributed memory architectures. In *Proceedings of 3rd IEEE Symposium on Large-Scale Data Analysis and Visualization (LDAV 2013), Atlanta GA, USA, Oct. 2013*, pp. 7–13, (2013).
12. Z. Chen, S. W. Son, W. Hendrix, A. Agrawal, W.-k. Liao, and A. Choudhary. Numarck: Machine learning algorithm for resiliency and checkpointing. In *Proceedings of*

- 26th International Conference on High Performance Computing, Networking, Storage and Analysis (Supercomputing, SC'14)*, pp. 733–744, (2014).
13. Y. Low, D. Bickson, J. Gonzalez, C. Guestrin, A. Kyrola, and J. M. Hellerstein. Distributed graphlab: a framework for machine learning and data mining in the cloud, *Proceedings of the VLDB Endowment*. **5**(8), 716–727, (2012).
 14. Y. Xie, D. Palsetia, G. Trajcevski, A. Agrawal, and A. Choudhary. Silverback: Scalable association mining for temporal data in columnar probabilistic databases. In *Proceedings of 30th IEEE International Conference on Data Engineering (ICDE), Industrial and Applications Track*, pp. 1072–1083, (2014).
 15. S. Jha, J. Qiu, A. Luckow, P. Mantha, and G. Fox. A tale of two data-intensive paradigms: Applications, abstractions, and architectures. In *Big Data (BigData Congress), 2014 IEEE International Congress on*, pp. 645–652 (June, 2014). doi: 10.1109/BigData.Congress.2014.137.
 16. Y. Xie, P. Daga, Y. Cheng, K. Zhang, A. Agrawal, and A. Choudhary. Reducing infrequent-token perplexity via variational corpora. In *Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistics (ACL) and the 7th International Joint Conference on Natural Language Processing*, pp. 609–615, (2015).
 17. D. Palsetia, W. Hendrix, S. Lee, A. Agrawal, W.-k. Liao, and A. Choudhary. Parallel community detection algorithm using a data partitioning strategy with pairwise sub-domain duplication. In *High Performance Computing, 31st International Conference, ISC High Performance 2016, Frankfurt, Germany, June 19-23, 2016, Proceedings*, pp. 98–115, (2016).
 18. E. Rangel, W. Hendrix, A. Agrawal, W.-k. Liao, and A. Choudhary. Agoras: A fast algorithm for estimating medoids in large datasets, *Procedia Computer Science*. **80**, 1159–1169, (2016).
 19. Q. Kang, W.-k. Liao, A. Agrawal, and A. Choudhary. A filtering-based clustering algorithm for improving spatio-temporal kriging interpolation accuracy. In *Proceedings of 25th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 2209–2214, (2016).
 20. Y. Xie, Z. Chen, D. Palsetia, G. Trajcevski, A. Agrawal, and A. Choudhary. Silverback+: Scalable association mining via fast list intersection for columnar social data, *Knowledge and Information Systems (KAIS)*. **50**(3), 969–997, (2017).
 21. Y. Xie, Z. Chen, A. Agrawal, and A. Choudhary. Distinguish polarity in bag-of-words visualization. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, pp. 3344–3350, (2017).
 22. R. Kohavi. Mining e-commerce data: the good, the bad, and the ugly. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 8–13. ACM, (2001).
 23. G. Linden, B. Smith, and J. York, Amazon. com recommendations: Item-to-item collaborative filtering, *Internet Computing, IEEE*. **7**(1), 76–80, (2003).
 24. D. L. Olson and Y. Shi, *Introduction to business data mining*. vol. 10, (McGraw-Hill/Irwin Englewood Cliffs, 2007).
 25. Y. Zhou, D. Wilkinson, R. Schreiber, and R. Pan. Large-scale parallel collaborative filtering for the netflix prize. In *Algorithmic Aspects in Information and Management*, pp. 337–348. Springer, (2008).
 26. Y. Xie, D. Honbo, A. Choudhary, K. Zhang, Y. Cheng, and A. Agrawal. Voxsup: a social engagement framework. pp. 1556–1559. ACM, (2012). Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD).
 27. H. Chen, R. H. Chiang, and V. C. Storey, Business intelligence and analytics: From

- big data to big impact., *MIS quarterly*. **36**(4), 1165–1188, (2012).
28. R. Liu, A. Agrawal, W.-k. Liao, and A. Choudhary. Enhancing financial decision-making using social behavior modeling. In *Proceedings of 8th KDD Workshop on Social Network Mining and Analysis for Business, Consumer and Social Insights (SNAKDD)*, pp. 13:1–13:5, (2014). Article No. 13.
 29. A. Agrawal, S. Misra, R. Narayanan, L. Polepeddi, and A. Choudhary. A lung cancer outcome calculator using ensemble data mining on seer data. In *Proceedings of the Tenth International Workshop on Data Mining in Bioinformatics, BIOKDD '11*, pp. 5:1–5:9, New York, NY, USA, (2011). ACM. ISBN 978-1-4503-0839-7. doi: <http://doi.acm.org/10.1145/2003351.2003356>. URL <http://doi.acm.org/10.1145/2003351.2003356>.
 30. A. Agrawal and A. Choudhary, Association rule mining based hotspot analysis on seer lung cancer data, *International Journal of Knowledge Discovery in Bioinformatics*. **2**(2), 34–54, (2011).
 31. H. C. Koh, G. Tan, et al., Data mining applications in healthcare, *Journal of health-care information management*. **19**(2), 65, (2011).
 32. A. Agrawal, S. Misra, R. Narayanan, L. Polepeddi, and A. Choudhary, Lung cancer survival prediction using ensemble data mining on seer data, *Scientific Programming*. **20**(1), 29–42, (2012).
 33. J. S. Mathias, A. Agrawal, J. Feinglass, A. J. Cooper, D. W. Baker, and A. Choudhary, Development of a 5 year life expectancy index in older adults using predictive mining of electronic health record data, *Journal of the American Medical Informatics Association*. **20**, e118–e124, (2013).
 34. L. Liu, J. Tang, Y. Cheng, A. Agrawal, W.-k. Liao, and A. Choudhary. Mining diabetes complication and treatment patterns for clinical decision support. In *Proceedings of 22th ACM International Conference on Information and Knowledge Management (CIKM 2013), San Francisco, USA, Oct. 2013*, pp. 279–288, (2013).
 35. A. Agrawal and A. Choudhary. An analysis of variation in hospital billing using medicare data. In *Proceedings of the KDD Workshop on Data Mining for Healthcare (DMH)*, pp. 1–6, (2013).
 36. C. K. Reddy and C. C. Aggarwal, *Healthcare data analytics*. vol. 36, (CRC Press, 2015).
 37. A. Agrawal, J. Mathias, D. Baker, and A. Choudhary. Identifying hotspots in five year survival electronic health records of older adults. In *Proceedings of 6th IEEE International Conference on Computational Advances in Bio and Medical Sciences (ICCABS)*, pp. 1–6, (2016).
 38. A. Agrawal, J. Mathias, D. Baker, and A. Choudhary. Five year life expectancy calculator for older adults. In *Proceedings of IEEE International Conference on Data Mining (ICDM)*, pp. 1280–1283, (2016).
 39. R. Al-Bahrani, A. Agrawal, and A. Choudhary, Survivability prediction of colon cancer patients using neural networks, *Health Informatics Journal*. p. 1460458217720395, (2017).
 40. K. Lee, S. A. Hasan, O. Farri, A. Choudhary, and A. Agrawal. Medical concept normalization for online user-generated texts. In *Proceedings of the Fifth International Conference on Healthcare Informatics (ICHI)*, pp. 462–469, (2017).
 41. W. Hendrix, I. Tetteh, A. Agrawal, F. Semazzi, W.-k. Liao, and A. Choudhary. Community dynamics and analysis of decadal trends in climate data. In *3rd IEEE ICDM Workshop on Knowledge Discovery from Climate Data, (ClimKD)*, pp. 9–14, (2011).
 42. F. Olaiya and A. B. Adeyemo, Application of data mining techniques in weather pre-

- diction and climate change studies, *International Journal of Information Engineering and Electronic Business (IJIEEB)*. **4**(1), 51, (2012).
43. Z. Chen, Y. Xie, Y. Cheng, K. Zhang, A. Agrawal, W.-k. Liao, N. Samatova, and A. Choudhary. Forecast oriented classification of spatio-temporal extreme events. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2952–2954, (2013).
 44. J. H. Faghmous and V. Kumar. Spatio-temporal data mining for climate data: Advances, challenges, and opportunities. In *Data Mining and Knowledge Discovery for Big Data*, pp. 83–116. Springer, (2014).
 45. A. R. Ganguly, E. Kodra, A. Agrawal, A. Banerjee, S. Boriah, S. Chatterjee, S. Chatterjee, A. Choudhary, D. Das, J. Faghmous, P. Ganguli, S. Ghosh, K. Hayhoe, C. Hays, W. Hendrix, Q. Fu, J. Kawale, D. Kumar, V. Kumar, W.-k. Liao, S. Liess, R. Mawalagedara, V. Mithal, R. Oglesby, K. Salvi, P. K. Snyder, K. Steinhaeuser, D. Wang, and D. Wuebbles. Toward enhanced understanding and projections of climate extremes using physics-guided data mining techniques, *Nonlinear Processes in Geophysics*. **21**, 777–795, (2014).
 46. C. Jin, Q. Fu, H. Wang, W. Hendrix, Z. Chen, A. Agrawal, A. Banerjee, and A. Choudhary. Running map inference on million node graphical models: A high performance computing perspective. In *Proceedings of the 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, pp. 565–575, (2015).
 47. V. Lakshmanan, E. Gilleland, A. McGovern, and M. Tingley, *Machine Learning and Data Mining Approaches to Climate Science*. (Springer, 2015).
 48. S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, Gapped BLAST and PSI-BLAST: A New Generation of Protein Database Search Programs., *Nucleic Acids Research*. **25**(17), 3389–3402, (1997). ISSN 0305-1048. doi: 10.1093/nar/25.17.3389. URL <http://dx.doi.org/10.1093/nar/25.17.3389>.
 49. A. Agrawal and X. Huang, PSIBLAST_PairwiseStatSig: reordering PSI-BLAST hits using pairwise statistical significance, *Bioinformatics*. **25**(8), 1082–1083, (2009).
 50. A. Agrawal, A. Mittal, R. Jain, and R. Takkar, Fuzzy-adaptive-thresholding-based exon prediction, *International Journal of Computational Biology and Drug Design*. **3**(4), 311–333, (2010).
 51. A. Agrawal and X. Huang, Pairwise statistical significance of local sequence alignment using sequence-specific and position-specific substitution matrices, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. **8**(1), 194–205, (2011).
 52. S. Misra, A. Agrawal, W.-k. Liao, and A. Choudhary, Anatomy of a hash-based long read sequence mapping algorithm for next generation dna sequencing, *Bioinformatics*. **27**(2), 189–195, (2011).
 53. A. Agrawal, S. Misra, D. Honbo, and A. Choudhary, Parallel pairwise statistical significance estimation of local sequence alignment using message passing interface library, *Concurrency and Computation: Practice and Experience*. **23**(17), 2269–2279, (2011).
 54. Y. Zhang, S. Misra, A. Agrawal, M. M. A. Patwary, W.-k. Liao, Z. Qin, and A. Choudhary, Accelerating pairwise statistical significance estimation for local alignment by harvesting gpu’s power, *BMC Bioinformatics*. **13**(Suppl 5), S3, (2012).
 55. A. O’Driscoll, J. Daugelaite, and R. D. Sleator, ‘big data’, hadoop and cloud computing in genomics, *Journal of biomedical informatics*. **46**(5), 774–781, (2013).
 56. Y. Xie, Z. Chen, K. Zhang, Y. Cheng, D. K. Honbo, A. Agrawal, and A. Choudhary, Muses: a multilingual sentiment elicitation system for social media data, *IEEE*

- Intelligent Systems*. **99**, 1541–1672, (2013).
57. K. Lee, A. Agrawal, and A. Choudhary. Real-time disease surveillance using twitter data: Demonstration on flu and cancer. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pp. 1474–1477, New York, NY, USA, (2013). ACM. ISBN 978-1-4503-2174-7. doi: 10.1145/2487575.2487709. URL <http://doi.acm.org/10.1145/2487575.2487709>.
 58. Y. Xie, Z. Chen, Y. Cheng, K. Zhang, A. Agrawal, W.-k. Liao, and A. Choudhary. Detecting and tracking disease outbreaks by mining social media data. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2958–2960, (2013).
 59. D. Palsetia, M. M. A. Patwary, A. Agrawal, and A. Choudhary, Excavating social circles via user-interests, *Social Network Analysis and Mining*. **4**(1), 1–12, (2014).
 60. D. Palsetia, M. Patwary, W. Hendrix, A. Agrawal, and A. Choudhary. Clique guided community detection. In *Proceedings of 2014 IEEE International Conference on Big Data (BigData)*, pp. 500–509, (2014).
 61. Y. Cheng, A. Agrawal, H. Liu, and A. Choudhary. Social role identification via dual uncertainty minimization regularization. In *Proceedings of International Conference on Data Mining (ICDM)*, pp. 767–772, (2014).
 62. R. Zafarani, M. A. Abbasi, and H. Liu, *Social media mining: an introduction*. (Cambridge University Press, 2014).
 63. K. Lee, A. Agrawal, and A. Choudhary. Mining social media streams to improve public health allergy surveillance. In *Proceedings of IEEE/ACM International Conference on Social Networks Analysis and Mining (ASONAM)*, pp. 815–822, (2015).
 64. K. Lee, A. Agrawal, and A. Choudhary. Forecasting influenza levels using real-time social media streams. In *Proceedings of the Fifth International Conference on Healthcare Informatics (ICHI)*, pp. 409–414, (2017).
 65. C. C. Fischer, K. J. Tibbetts, D. Morgan, and G. Ceder, Predicting crystal structure by merging data mining with quantum mechanics, *Nature materials*. **5**(8), 641–646, (2006).
 66. K. Gopalakrishnan, A. Agrawal, H. Ceylan, S. Kim, and A. Choudhary, Knowledge discovery and data mining in pavement inverse analysis, *Transport*. **28**(1), 1–10, (2013).
 67. P. Deshpande, B. P. Gautham, A. Cecen, S. Kalidindi, A. Agrawal, and A. Choudhary, *Application of Statistical and Machine Learning Techniques for Correlating Properties to Composition and Manufacturing Processes of Steels*, In *2nd World Congress on Integrated Computational Materials Engineering*, pp. 155–160. John Wiley & Sons, Inc., (2013). ISBN 9781118767061.
 68. 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*. **3**(8), 1–19, (2014).
 69. B. Meredig, A. Agrawal, S. Kirklin, J. E. Saal, J. W. 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*. **89** (094104), 1–7, (2014).
 70. 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*. **4**(13), 1–17, (2015).
 71. R. Liu, A. Kumar, Z. Chen, A. Agrawal, V. Sundararaghavan, and A. Choudhary, A predictive machine learning approach for microstructure optimization and materials

- design, *Nature Scientific Reports*. **5**(11551), (2015).
72. S. R. Kalidindi and M. D. Graef, Materials data science: Current status and future outlook, *Annual Review of Materials Research*. **45**(1), 171–193, (2015). doi: 10.1146/annurev-matsci-070214-020844. URL <http://dx.doi.org/10.1146/annurev-matsci-070214-020844>.
 73. L. Ward, A. Agrawal, A. Choudhary, and C. Wolverton, A general-purpose machine learning framework for predicting properties of inorganic materials, *npj Computational Materials*. **2**(16028), (2016).
 74. A. Agrawal and A. Choudhary. A fatigue strength predictor for steels using ensemble data mining. In *Proceedings of 25th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 2497–2500, (2016).
 75. A. Furmanchuk, A. Agrawal, and A. Choudhary, Predictive analytics for crystalline materials: Bulk modulus, *RSC Advances*. **6**(97), 95246–95251, (2016).
 76. A. Agrawal, B. Meredig, C. Wolverton, and A. Choudhary. A formation energy predictor for crystalline materials using ensemble data mining. In *Proceedings of IEEE International Conference on Data Mining (ICDM)*, pp. 1276–1279, (2016).
 77. L. Ward, R. Liu, A. Krishna, V. I. Hegde, A. Agrawal, A. Choudhary, and C. Wolverton, Including crystal structure attributes in machine learning models of formation energies via voronoi tessellations, *Physical Review B*. **96**(2), 024104, (2017).
 78. R. Liu, Y. C. Yabansu, Z. Yang, A. N. Choudhary, S. R. Kalidindi, and A. Agrawal, Context aware machine learning approaches for modeling elastic localization in three-dimensional composite microstructures, *Integrating Materials and Manufacturing Innovation*. pp. 1–12, (2017).
 79. A. Furmanchuk, J. E. Saal, J. W. Doak, G. B. Olson, A. Choudhary, and A. Agrawal, Prediction of seebeck coefficient for compounds without restriction to fixed stoichiometry: A machine learning approach, *Journal of Computational Chemistry*. **39**(4), 191–202, (2018). ISSN 1096-987X. doi: 10.1002/jcc.25067. URL <http://dx.doi.org/10.1002/jcc.25067>.
 80. Y. Liu, W.-k. Liao, and A. Choudhary. Design and evaluation of a parallel hop clustering algorithm for cosmological simulation. In *Parallel and Distributed Processing Symposium, 2003. Proceedings. International*, pp. 8–pp. IEEE, (2003).
 81. N. M. Ball and R. J. Brunner, Data mining and machine learning in astronomy, *International Journal of Modern Physics D*. **19**(07), 1049–1106, (2010).
 82. M. Brescia, S. Cavuoti, M. Paolillo, G. Longo, and T. Puzia, The detection of globular clusters in galaxies as a data mining problem, *Monthly Notices of the Royal Astronomical Society*. **421**(2), 1155–1165, (2012).
 83. J. VanderPlas, A. J. Connolly, Ž. Ivezić, and A. Gray. Introduction to astroml: Machine learning for astrophysics. In *Intelligent Data Understanding (CIDU), 2012 Conference on*, pp. 47–54. IEEE, (2012).
 84. A. Agnello, B. C. Kelly, T. Treu, and P. J. Marshall, Data mining for gravitationally lensed quasars, *Monthly Notices of the Royal Astronomical Society*. **448**(2), 1446–1462, (2015).
 85. E. Rangel, N. Li, S. Habib, T. Peterka, A. Agrawal, W.-k. Liao, and A. Choudhary. Parallel dtfe surface density field reconstruction. In *Proceedings of IEEE Cluster*, pp. 30–39, (2016).
 86. E. Rangel, N. Frontiere, S. Habib, K. Heitmann, W.-k. Liao, A. Agrawal, and A. Choudhary. Building halo merger trees from the q continuum simulation. In *Proceedings of 24th IEEE International Conference on High Performance Computing, Data, and Analytics (HiPC'17)*, pp. 398–407, (2017).
 87. Y. LeCun, Y. Bengio, and G. Hinton, Deep learning, *nature*. **521**(7553), 436, (2015).

88. A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, (2012).
89. G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al., Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, *Signal Processing Magazine, IEEE*. **29**(6), 82–97, (2012).
90. K. Rajan, Materials informatics: The materials "gene" and big data, *Annual Review of Materials Research*. **45**(1), 153–169, (2015). doi: 10.1146/annurev-matsci-070214-021132.
91. Materials Genome Initiative for Global Competitiveness, June 2011; OSTP 2011.
92. Materials Genome Initiative Strategic Plan, National Science and Technology Council Committee on Technology Subcommittee on the Materials Genome Initiative, June 2014.
93. R. Liu, A. Agrawal, W.-k. Liao, M. D. Graef, and A. Choudhary. Materials discovery: Understanding polycrystals from large-scale electron patterns. In *Proceedings of IEEE BigData Workshop on Advances in Software and Hardware for Big Data to Knowledge Discovery (ASH)*, pp. 2261–2269, (2016).
94. K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection, *Construction and Building Materials*. **157**, 322–330, (2017).
95. K. Gopalakrishnan, H. Gholami, A. Vidyadharan, A. Choudhary, and A. Agrawal, Crack damage detection in unmanned aerial vehicle images of civil infrastructure using pre-trained deep learning model, *International Journal for Traffic and Transport Engineering*. **8**(1), 1–14, (2018).
96. M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential deep learning for human action recognition. In *International Workshop on Human Behavior Understanding*, pp. 29–39. Springer, (2011).
97. K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, (2016).
98. R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pp. 160–167. ACM, (2008).
99. X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 513–520, (2011).
100. G. E. Dahl, D. Yu, L. Deng, and A. Acero, Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition, *IEEE Transactions on audio, speech, and language processing*. **20**(1), 30–42, (2012).
101. A. Coates, B. Huval, T. Wang, D. Wu, B. Catanzaro, and N. Andrew. Deep learning with cots hpc systems. In *International Conference on Machine Learning*, pp. 1337–1345, (2013).
102. S. Lee, D. Jha, A. Agrawal, A. Choudhary, and W.-k. Liao. Parallel deep convolutional neural network training by exploiting the overlapping of computation and communication. In *Proceedings of 24th IEEE International Conference on High Performance Computing, Data, and Analytics (HiPC'17)*, pp. 183–192, (2017).
103. S. J. Pan and Q. Yang, A survey on transfer learning, *IEEE Transactions on knowledge and data engineering*. **22**(10), 1345–1359, (2010).

104. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, et al. Going deeper with convolutions. *Cvpr*, (2015).
105. H. Shin, H. Roth, M. Chen, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. Summers, Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning, *IEEE transactions on medical imaging*. **35**(5), 1285–1298, (2016).
106. 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*. **21**(3), 739–752, (2015).
107. K. Gopalakrishnan. Advanced pavement health monitoring and management, (2016). IGI Global Videos, <http://www.igi-global.com/video.aspx?ref=advanced-pavement-health-monitoring-management&titleid=137625>, Accessed May 09, 2017.
108. S. Chambon and J. Moliard, Automatic road pavement assessment with image processing: Review and comparison, *International Journal of Geophysics*. **2011**(989354), 20, (2011).
109. A. Ayenu-Prah and N. Attoh-Okine, Evaluating pavement cracks with bidimensional empirical mode decomposition, *EURASIP J. Adv. Signal Process*. **2008**(1), 861701, (2008).
110. L. Ying and E. Salari, Beamlet transform-based technique for pavement crack detection and classification, *Computer-Aided Civil and Infrastructure Engineering*. **25**(8), 572–580, (2010).
111. C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure, *Advanced Engineering Informatics*. **29**(2), 196–210, (2015).
112. L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu. Road crack detection using deep convolutional neural network. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp. 3708–3712, (2016).
113. A. Zhang, K. C. Wang, B. Li, E. Yang, X. Dai, Y. Peng, Y. Fei, Y. Liu, J. Q. Li, and C. Chen, Automated pixel-level pavement crack detection on 3d asphalt surfaces using a deep-learning network, *Computer-Aided Civil and Infrastructure Engineering*. **32**(10), 805–819, (2017).
114. H. Maeda, Y. Sekimoto, T. Seto, T. Kashiya, and H. Omata, Road damage detection using deep neural networks with images captured through a smartphone, *arXiv preprint arXiv:1801.09454*. (2018).
115. G. E. Elkins, T. Thompson, B. Ostrom, A. Simpson, and B. Visintine. Long-term pavement performance information management system: Pavement performance database user reference guide, final report fhwa-rd-03-088 (revision). Technical report, Federal Highway Administration (FHWA), Washington, D.C., USA, (2016).
116. F. Chollet et al. Keras. <https://github.com/fchollet/keras>, (2015).
117. D. Floreano and R. J. Wood, Science, technology and the future of small autonomous drones, *Nature*. **521**(7553), 460–466, (2015).
118. S. Sankarasrinivasan, E. Balasubramanian, K. Karthik, U. Chandrasekar, and R. Gupta, Health monitoring of civil structures with integrated uav and image processing system, *Procedia Computer Science*. **54**, 508–515, (2015).
119. S. Yokoyama and T. Matsumoto, Development of an automatic detector of cracks in concrete using machine learning, *Procedia Engineering*. **171**, 1250–1255, (2017).
120. A. Carrio, C. Sampedro, A. Rodriguez-Ramos, and P. Campoy, A review of deep learning methods and applications for unmanned aerial vehicles, *Journal of Sensors*. **2017**, 1–13, (2017).