

DETC2016-59404

DEEP NETWORK-BASED FEATURE EXTRACTION AND RECONSTRUCTION OF COMPLEX MATERIAL MICROSTRUCTURES

Ruijin Cang

Mechanical Engineering
Arizona State University
Tempe, Arizona, 85287
Email: cruijin@asu.edu

Max Yi Ren*

Mechanical Engineering
Arizona State University
Tempe, Arizona, 85287
Email: yiren@asu.edu

ABSTRACT

Computational material design (CMD) aims to accelerate optimal design of complex material systems by integrating material science and design automation. For tractable CMD, it is required that (1) a feature space be identified to allow reconstruction of new designs, and (2) the reconstruction process be property-preserving. Existing solutions rely on the designer's understanding of specific material systems to identify geometric and statistical features, which could be insufficient for reconstructing physically meaningful microstructures of complex material systems. This paper develops a feature learning mechanism that automates a two-way conversion between microstructures and their lower-dimensional feature representations. The proposed model is applied to four material systems: Ti-6Al-4V alloy, Pb-Sn alloy, Fontainebleau sandstone, and spherical colloids, to produce random reconstructions that are visually similar to the samples. This capability is not achieved by existing synthesis methods relying on the Markovian assumption of material systems. For Ti-6Al-4V alloy, we also show that the reconstructions preserve the mean critical fracture force of the system for a fixed processing setting. Source code and datasets are available.

1 Introduction

Computational material design (CMD) has been promoted by the Material Genome Initiative to fundamentally change the strategies for developing and manufacturing advanced material

systems to meet the urging demands in energy, health and national security. Pioneering CMD works based on gradient-based shape and topological optimization and analytical processing-structure-property relations have demonstrated successes in designing a variety of material systems, including functional polymers [1–3], alloys and ceramics [4–6], and polymer-matrix composites [7, 8], to name a few. Nonetheless, existing CMD methods lack scalability and have limited applications to multiscale and high-resolution microstructures. To this end, this paper proposes a computational tool for extraction and reconstruction of microstructure features on multiple length scales. The developed methodology from this paper could lead to better definition of the design space and enable computational design of complex material systems. In the remainder of this section, we introduce the mathematical formulation of the material microstructure design problem and review contemporary feature extraction techniques. Specifications of the proposed model and demonstrations of its performance can be found in Section 3. The advantages and limitations of the proposed algorithm will be discussed in Section 4, followed by conclusions in Section 5.

1.1 Material design as an optimization problem

We start by providing an overview of CMD to justify the necessity of design at the structure level. Computational material design can be mathematically formulated as follows: A material microstructure is represented as a 2D or 3D image $z \in \mathcal{Z}$, where \mathcal{Z} is the image space. Each element of z defines the composition or phase at its location. The image is called *feasible* when some

*Address all correspondence to this author.

processing setting $\theta \in \Theta$ exists, such that $z = p(\theta)$, with $p(\cdot)$ being the process-structure mapping, e.g., a physics-based simulation, and Θ the set of all available processing settings. Consider a material property of interest, e.g., critical fracture force of alloy, derived from a structure-property mapping $f(\cdot)$: $y = f(z)$. Finding the optimal material property y^* through an all-in-one approach, i.e., $\theta^* = \arg \min_{\theta \in \Theta} y = f(p(\theta))$, may not be favorable for the following reasons: (1) Finding the optimal θ^* can be computationally intractable if the problem is combinatorial, e.g., a combination of multiple processing techniques is needed. (2) The mapping $p(\theta)$ is often stochastic, i.e., a certain processing setting can lead to a set of microstructure images. One solution is to perform a nested optimization by first finding a target microstructure z^* that optimizes y , then searching for a processing setting θ^* to match z^* . This approach can be mathematically formulated as the following structure design problem constrained by processing feasibility:

$$\min_{\text{feasible } z} y = f(z). \quad (1)$$

Compared to the all-in-one formulation, Eq. (1) allows parallel searches for the potentially optimal structures and their corresponding processing settings. In addition, due to the high costs of evaluating $p(\cdot)$ and $f(\cdot)$, active learning methods can be leveraged to facilitate efficient search of z^* and θ^* , by iteratively refining predictive models for process-structure and structure-property mappings. The nested problem formulation in Eq. (1) allows these models be refined and validated independently.

1.2 Scientific challenges in optimal material structure design

For complex material systems, solving Eq. (1) will encounter the following issues: The high resolution of material samples required by material property simulations causes the microstructure set \mathcal{Z} to be high dimensional. Further, the nonlinear processing-structure mapping makes \mathcal{Z} costly to be characterized. These issues will be alleviated if the dimensionality of the design space can be reduced. Dimension reduction is possible due to the existence of common structural patterns in the given material system, indicating that there exists a feature representation that concisely encodes the raw material images. Further, since the structural patterns are often related to processing feasibility and material properties, the encoding may lead to easier-to-construct predictive models for the processing-structure-property mappings. Advantages of a feature representation were confirmed by recent studies for homogeneous nanoparticle composite design tasks [7, 8]. However, current practices rely on the designer’s understanding of specific material systems to identify geometric (e.g., particle size and orientation) and statistical descriptors (e.g., correlation functions) that are likely to explain variances in material properties. Generating such quantitative features for complex material systems could be difficult for human designers, or, the manually defined feature set would be insufficient for reconstruction of

physically meaningful microstructures. Feature extraction methods through the use of deep convolutional networks could potentially address this issue. Informally, this line of approaches use sampled microstructures to learn a mapping $x = \Phi(z)$ and its inverse $z = \Phi^{-1}(x)$ (the unsupervised learning step), where x is a learned feature representation of z from a feature space \mathcal{X} , before mapping x to the performance y (the supervised learning step).

A unique challenge in developing a feature extraction mechanism for (material) design is the demanding quality of decoding (or called reconstruction), i.e., the reconstruction $\Phi(\Phi^{-1}(z))$ shall “match” z , so that any design searched in \mathcal{X} can be correctly validated through a physics-based simulation or an experiment. In Section 3.5, we examine both the visual appearances and the mean property values of $\Phi(\Phi^{-1}(z))$ and z . Visually meaningful decoding is yet challenging for existing deep networks: the deeper the network is, the more details are neglected in the reconstruction [9–11]. To illustrate, photos can be encoded as text labels with a state-of-the-art deep network (e.g., [12, 13]) but image restoration from text labels is hard to be precise [11, 14]. Existing studies have achieved limited success at this challenge by imposing structures onto the image space, e.g., constraining the smoothness of the reconstructed image [9, 10, 14]. This study first investigates a deep network model and postprocessing steps to enable random reconstruction of Ti-6Al-4V alloy samples. We then show that the proposed model can be applied to other material systems, including Pb-Sn (lead-tin) alloy, Pore structure of Fontainebleau sandstone, and 2D suspension of spherical colloids.

1.3 Contributions and limitations

To the authors knowledge, this is the first study that develops a deep network model for both feature extraction and reconstruction of complex material systems. In addition, our method preserves the material property of interest statistically, and thus will ensure the validity of optimal designs identified from the feature space. Nonetheless, while computationally inexpensive, the current reconstruction process is by nature non-scalable. A hybrid approach that combines the proposed feature extraction network and a stochastic material synthesis model will be discussed.

2 Related work

Design at the microstructure level by changing the composition, phases and morphologies could lead to the discovery of unprecedented material systems with advanced performance [15]. The challenges, however, exist in the modeling of interactions among multiple material constituents to predict the properties, and the automation of material design at microstructure level. In the following, we review existing studies that concern the latter challenge, and then introduce background knowledge for the proposed approach based on Convolutional Deep Belief Network (CDBN).

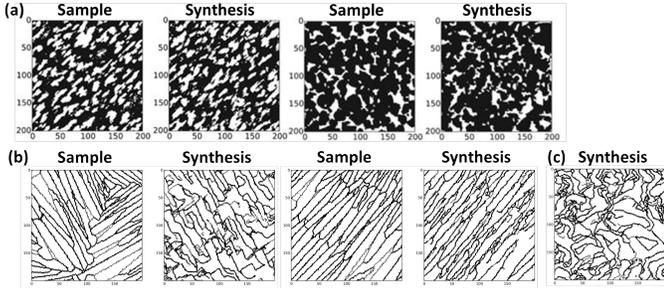


FIGURE 1. (a) Samples and random synthesis results of material systems that are assumed to be Markovian. Images are taken from [35]. (b-c) Random synthesis of the Ti-6Al-4V alloy microstructure following the synthesis algorithm from [35]. The synthesis is based on (b) a single sample and (c) 100 samples from Fig. 10. Image courtesy of Ramin Bostanabad from Northwestern University.

2.1 Microstructure parametrization and reconstruction

Conventional material design relies on choosing different material compositions from material databases [16]. For example, carbon-fibre reinforced epoxy are applied to replace the wooden oars for better performance [17]. This composition-based approach is often limited and cannot be applied in the design of complex materials systems due to the existence of features other than constituents that govern material properties. Such features include morphology of microstructure, i.e., the spatial arrangement of local microstructural features [18], the heterogeneity in microstructure [7] and others. A quantitative representation of material systems is thus needed.

A review from [7] summarizes three categories of microstructure representations: (1) physical descriptors (e.g., composition descriptors such as volume fraction, dispersion descriptors [19–24] such as average distance between fillers, and geometry descriptors [19, 23, 25–30] such as the size of fillers), (2) two-point or N -point correlation functions [20, 25–27, 31], and (3) random fields [25, 32–34]. The last method is proved to be efficient at synthesizing *Markovian* microstructures [35] where the probability distribution of the material composition at each pixel (or voxel) is determined by its *local* surroundings and the conditional probability model can be applied *homogeneously* across a microstructure sample, see Fig. 1a for examples. However, the Markovian assumption may not hold for complex material systems. For example, the algorithm from [35] produces less plausible synthesis results when trained on Ti-6Al-4V alloy samples, see Fig. 1b,c.

2.2 Convolutional deep network

Convolutional networks have long been used for feature extraction, with applications to object and voice recognition [36]

and detection [37], reinforcement learning [38–40], analogy making [41] and many others. By stacking multiple layers, a deep convolutional network can learn from input training data multi-scale features that contribute to the explanation of the corresponding outputs, without necessarily being guided by the outputs. For example, a convolutional network can extract face elements from a large set of human face images, see Fig. 2a. The outputs of the network thus represent the existence of such elements, and is more effective at detecting human faces than using raw image pixel values [42]. The training of deep learning models is made tractable by using stochastic gradient descent [43] combined with back-propagation (for deterministic models) [44] or with contrastive divergence (for probabilistic models such as Restricted Boltzmann Machines (RBM)) [45]. The proposed CDBN network is a par-

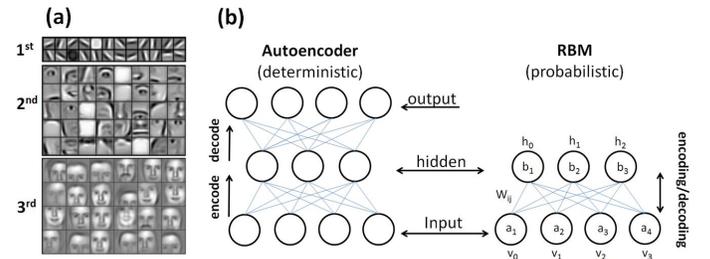


FIGURE 2. (a) Three layers of human face features extracted at increasing length scales [42]. (b) A schematic comparison between an RBM and an autoencoder. a_n in the circle indicates the bias value of input layer and v_n means the value of input layer; b_n in the circle indicates the bias value of hidden layer and h_n means the value of hidden layer; W_{ij} indicates the weights between the input layer and hidden layer

ticular type of deep network that consists of stacked RBM layers. A CDBN is generative, meaning that it can perform both dimension reduction for an input microstructure, and reconstruction of the microstructure with given high-level feature values. It should be noted that other generative models could also be developed for reconstruction purpose, including autoencoder [46] (see Fig. 2b for an illustration and [47] for an example) and its variational version [14, 48]. These models are not investigated in this paper.

2.3 Restricted Boltzmann Machine

We briefly introduce RBM and its training algorithm. A fully-connected RBM layer consists of visible (input) and hidden (output) nodes, see Fig. 2b. With weighted edges connecting these two sets of nodes, it is a complete bipartite graph. The set of edges connecting from all visible nodes to one hidden node acts as a convolution filter applied to the input image. The output of the convolution is transformed by a sigmoid function, and treated as the *activation* of the corresponding hidden nodes. The *state*

of each hidden node is a binary number drawn from a Bernoulli distribution parameterized by the activation. For K filters, the hidden layer forms a binary image with K channels, served as the input to the next RBM in the CDBN. In a Convolutional RBM (or CRBM), the filter size is typically smaller than the image size, and the set of hidden nodes forms a binary image once the input image is scanned by the filter, see Fig. 3a. Thus, a CRBM only has a small subset of the visible nodes connecting to each of the hidden nodes. Network weights representing the same filter pixels have shared values. In the rest of the paper, the term “filter” and “feature” are used interchangeably. The training of an RBM (and CRBM) is as follows. Consider a model with n visible and m hidden nodes, and let v_i and h_j be the states of the i th visible and the j th hidden nodes, respectively. When visible units are binary-valued, the total energy of the layer is defined as:

$$E(v, h|\theta) = -\sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j - \sum_{i=1}^n \sum_{j=1}^m v_i W_{ij} h_j, \quad (2)$$

or when they are real-valued, we have

$$E(v, h|\theta) = \frac{1}{2} \sum_{i=1}^n v_i^2 - \sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j - \sum_{i=1}^n \sum_{j=1}^m v_i W_{ij} h_j, \quad (3)$$

In the above equations, $\theta = \{W_{ij}, a_i, b_j\}$ is a set of model parameters to be estimated, within which W_{ij} are the network weights, and a_i and b_i represent the biases in the visible and hidden nodes, respectively. Based on this energy function, the joint probability of (v, h) can be derived as:

$$P(v, h|\theta) = \frac{\exp(-E(v, h|\theta))}{Z(\theta)}, \quad (4)$$

where $Z(\theta) = \sum_{v, h} \exp(-E(v, h|\theta))$ is called a partition function.

The probability of the state v in the visible layer is defined as the marginal probability:

$$P(v|\theta) = \frac{1}{Z(\theta)} \sum e^{-E(v, h|\theta)}. \quad (5)$$

Thus, the log-likelihood of the training data can be written as $\phi(\theta) = \log(P(v|\theta)) = \phi^+ - \phi^-$, where $\phi^+ = \log \sum_h \exp(-E(v, h))$ and $\phi^- = \log Z = \log \sum_{v, h} \exp(-E(v, h))$ are called the positive and negative parts. The gradient of the positive part can be simply calculated as $\frac{\partial \phi^+}{\partial w_{ij}} = v_i P(h_j = 1|v)$, while the negative gradient $\frac{\partial \phi^-}{\partial w_{ij}} = P(v_i = 1, h_j = 1)$ is usually computationally expensive due to the calculation of the partition function. This issue, however, is resolved by using Contrastive Divergence [45] to provide an asymptotic approximation of $\frac{\partial \phi^-}{\partial w_{ij}}$.

2.4 Probabilistic max-pooling

In a deep network, a max-pooling layer is often inserted after a convolution layer to reduce the number of hidden nodes. This allows fewer network weights to be computed in the follow-up layer and also facilitates extraction of feature at a larger scale. Specific to CDBN, a probabilistic max-pooling layer segments the output image of a hidden RBM layer into blocks and assigns the associated pooling node a binary value a by drawing from a Bernoulli distribution defined by the activations of the hidden nodes: Without loss of generality, we assume each block is 2×2 . The activations for the four hidden nodes are denoted by I_i for $i = 1, 2, 3, 4$. The probability for the pooling node to be activated is $P(a = 1) = 1 - (1 + \sum_{i=1}^4 I_i)^{-1}$. Inversely, during the reconstruction, we assign ones to all four nodes in the block if $a = 1$ or otherwise zeros. Fig. 3b summarizes the pooling procedure.

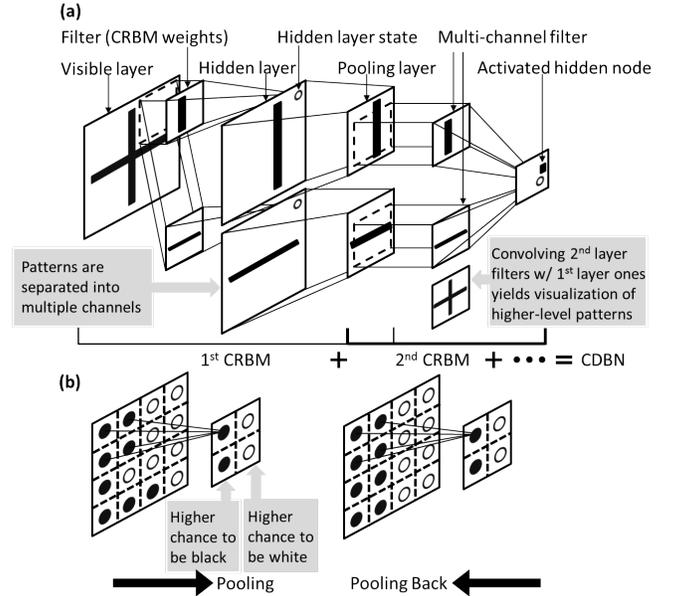


FIGURE 3. (a) CRBM and pooling layers (b) probabilistic max-pooling for 2×2 blocks

3 Proposed CDBN for feature extraction and material reconstruction

The proposed network has three convolutional and two fully-connected RBM layers. Each of the first two CRBM layers is followed by a 2×2 probabilistic max-pooling layer. The convolutional filters for the three layers are 6×6 , 9×9 , and 9×9 in size and with 24, 40, and 288 channels, respectively. The sizes of the layers are $200 \times 200 \times 1$, $97 \times 97 \times 24$, $44 \times 44 \times 40$,

$36 \times 36 \times 288$ and 30×1 for the input and the five hidden layers, respectively. Table 1 summarizes the model and training parameters. For the first four layers, an average sparsity of 0.1 is imposed on the hidden layer activations. The p_λ value specifies the weight on the gradient of the sparsity penalty during the training, and is manually tuned.

TABLE 1. A list of algorithmic parameters that can be tuned.

Layer	#filter	filter size	sparsity	p_λ	rotation
1 st	2	6×6	0.1	10	12
2 nd	40	9×9	0.1	10	N/A
3 rd	288	9×9	0.1	10	N/A
4 th	1000	36×36	0.1	10	N/A
5 th	30	1000×1	0	0	N/A

3.1 Orientation-invariant filters

One common trait of microstructure images is the existence of the same low-level features with different linear transformations across the image. For example, the grain boundaries of alloy samples shown in Fig. 1b are similar with only orientation differences. This indicates that a set of filters of the CRBM layers can be parametrized by linear transformations, thus significantly reducing the number of network weights to be trained. Specifically, our model incorporates $K = 2$ orientation-invariant filters in the first CRBM layer, and each filter has $S = 12$ pre-defined orientations with even intervals of 15 degrees. Let the transformation matrices be $\{T_s\}_{s=1}^S$, the RBM energy function can be updated accordingly to incorporate the transformation [49]:

$$E(v, H) = - \sum_{j=1}^K \sum_{s=1}^S (T_s v)^T w_j h_{j,s} - \sum_{j=1}^K \sum_{s=1}^S b_{j,s} h_{j,s} - c^T v. \quad (6)$$

In addition, among all S hidden nodes $h_{j,s}$ for the j th filter, at most one can be activated, i.e., $\sum_{s=1}^S h_{j,s} \leq 1, h_{j,s} \in 0, 1, j = 1, \dots, K$. This is achieved by considering $h_{j,s}$ as an outcome of a softmax classifier, with

$$p(h_{j,s} = 1|v) = \frac{\exp((T_s v)^T w_j + b_{j,s})}{1 + \sum_{s'=1}^S \exp((T_{s'} v)^T w_j + b_{j,s'})}. \quad (7)$$

For the visible layer, we still have:

$$p(v_i = 1|h) = \text{sigmoid} \left(\sum_{j,s} (T_s^T w_j)_i h_{j,s} + c_i \right). \quad (8)$$

This modification to the training problem allows a single filter to be extracted for multiple image patterns that are only different by rotation, or equivalently, it allows the hidden layer to have explicit orientation awareness.

3.2 Explanation of the network model

The configuration of the network is fine-tuned manually. Here we explain the rationale behind the use of the five RBM layers. The first three layers are used to extract local features at an increasing length scale. To demonstrate, Fig. 4 visualizes these filters learned from 100 Ti-6Al-4V alloy samples that are produced from the same laser sintering process. A full set of these samples can be found in Appendix B. The 4th layer, a fully-connected RBM, then extracts global features from the third layer. Examples of 4th-layer filters can be found in Fig. 7a. The last layer is used to reduce the dimensionality by learning the correlations among these global features. It should be noted that the last layer is also necessary to avoid dense sampling of similar microstructures in the resultant design space. To illustrate, Fig. 5 compares random reconstructions uniformly sampled from the 4th and the 5th layers (see Section 3.3 for details on reconstruction). One can see that samples from the 4th layer are more similar to each other.

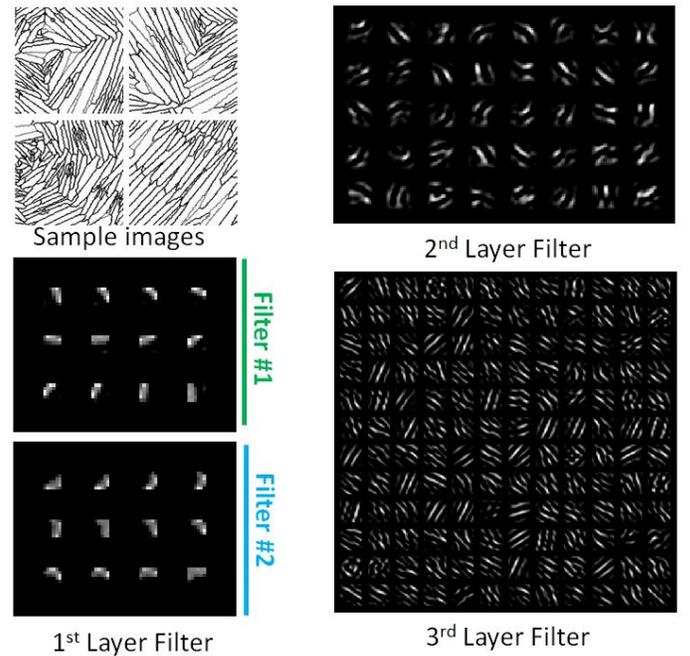


FIGURE 4. Visualizations of filters from the first three layers: The 1st layer has two filters, each with 12 orientations (see Section 3.1). Only the first 144 filters from the 3rd layer are shown due to limited space.

3.3 Reconstruction and postprocessing

To reconstruct random microstructures, one starts by randomly assigning binary values to the 5th layer of the trained

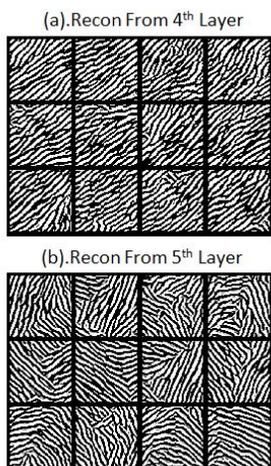


FIGURE 5. Comparison between random reconstructions from (a) the 4th and (b) the 5th layers, all uniformly sampled from the corresponding design spaces ($\{0, 1\}^{1000}$ in (a) and $\{0, 1\}^{30}$ in (b)).

CDBN, and inversely sample the previous layers through deconvolution. To speed up the process, we directly treat activations (real values between 0 and 1) as the states of hidden nodes, to avoid draws from Bernoulli distributions and to restrict the only source of randomness in reconstruction to the 5th layer. We discuss below the necessity of post-processing steps to avoid undesirable reconstructions. First, we observe that with the proposed reconstruction method, the 3rd hidden layer (a $36 \times 36 \times 288$ image) is overly activated, see Fig. 6a. The slightly activated nodes in the 3rd layer across its 288 channels lead to regions in the input layer with overlapped grain boundaries (as black pixels) and such regions are undesirably visualized as voids. Thresholding the activations of the 3rd layer at 0.5 helps to achieve a visually more reasonable reconstruction, as shown in Fig. 6b. This threshold is empirically chosen. Secondly, we notice that the resultant grain boundaries are wider than those from the original samples. This is due to the convolution operation and the pooling. Applying an existing skeletonization method [50] to the reconstruction results in the final outcome in Fig. 6c. Lastly, the convolution operation tends to smear the reconstructed image towards its boundary. The reported reconstruction results in this paper are the cropped areas (143×143) without the blurred margins.

3.4 Random reconstruction results

Here we demonstrate the reconstruction performance of the proposed CDBN on four material systems: Ti-6Al-4V alloy (100 images in Fig. 10), Pb-Sn (lead-tin) (60 images in Fig. 11), Pore structure of Fontainebleau sandstone (60 images in Fig. 12) and 2D suspension of spherical colloids (80 images in Fig. 13). The CDBN specifications follow Section 3. Fig. 7 provides a visual comparison between samples and the corresponding random re-

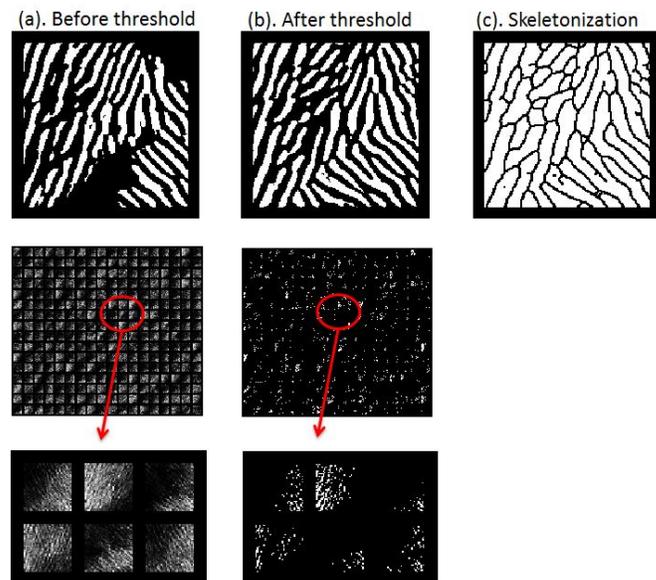


FIGURE 6. Postprocessing steps: Column (a) shows the original reconstruction and its 3rd layer activations (288 channels) with enlarged sample channels; Column (b) shows the reconstruction after thresholding the 3rd layer activations at 0.5; Column (c) shows the further improved reconstruction after skeletonization.

constructions. The first three material systems follow almost the same network setting, with the only exception in Ti-6Al-4V alloy where the first layer is orientation-invariant. The network for spherical colloids is slightly different than the other three, since its local features, the sphere, can be directly extracted from a single CRBM layer. Thus the rest four layers in this case are all fully-connected RBMs. From the results, it is evident that the proposed network has some general applicability across material systems.

3.5 Deeper understanding of the proposed model

To further understand the performance and limitations of the proposed CDBN, we conduct two additional studies focusing on the reconstruction of the Ti-6Al-4V alloy system: The first concerns the restoration of original samples, and the second validates the reconstruction of the material property of interest, i.e., the critical fracture force.

Samples of the original microstructure images and their corresponding reconstructions are shown in Fig. 8. A close examination reveals discrepancies between the reconstructed and the original images. We believe that such discrepancies are caused by the nature of the network: Similar local patterns are learned as a single filter, and are replaced by the filter during reconstruction. To visually improve the restoration, one could introduce more

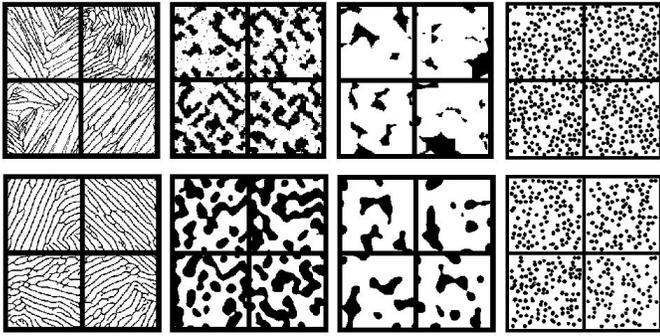


FIGURE 7. A comparison between random reconstructions (bottom) and the original samples (top) for four different material systems.

filters to the network (and fine-tune the sparsity). Yet the true question is: Does such visual (and thus structural) discrepancies induce differences in the material performance?

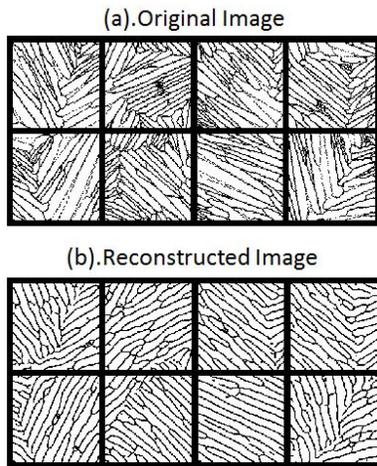


FIGURE 8. Comparison between the samples (top) and their reconstructions (bottom). Details of the two fail to match.

To answer this question, we calculate the critical fracture force of three sets of samples: the original samples, their reconstructions, and 100 random reconstructions, using an existing simulation package [51]. The simulation results are summarized in Fig. 9, with two findings: (1) An ANOVA test shows that the mean forces of the three groups are statistically the same; and (2) discrepancies in the fracture force are visible between the original samples and their reconstructions, due to the aforementioned structural difference. We argue that finding (1) is a desirable result, while individual-wise property matching may

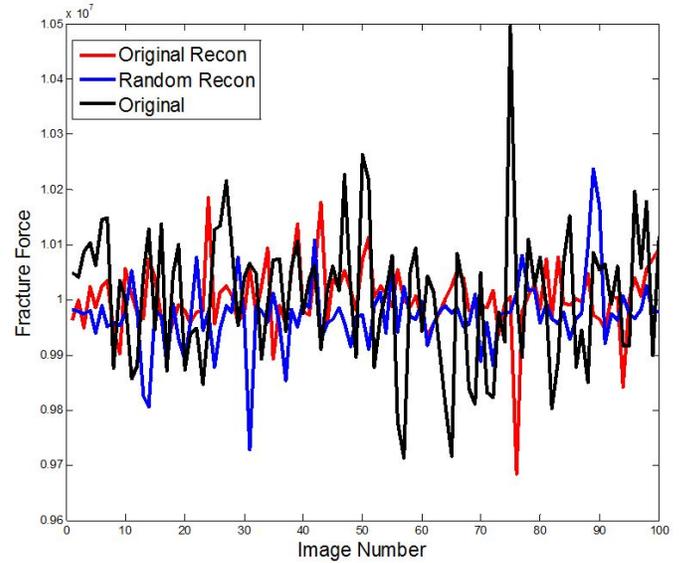


FIGURE 9. Comparison on critical fracture forces between the original Ti-6Al-4V samples, their corresponding reconstructions and random reconstructions. (color online)

not be necessary. This is because each processing setting may produce various structures with small variance in their property values. Thus, during the structural design, it is required that the set of reconstructions of all samples that are derived from the same processing parameters (as is the case in this study) have the same *mean* property value as that of the original samples. To further examine if the proposed model satisfies this requirement, more experiments using samples and their reconstructions from different processing parameters will be needed.

4 Discussion

Here we provide more discussions on the limitations and future directions of the presented work. First, we note that the network is designed to only reconstruct images of a fixed size, which is not desirable when scalable synthesis of material systems is needed. One potential solution is to apply a conditional probability model (e.g., [35]) to the outputs of a hidden layer of the CDBN that represent the activations of key microstructure patterns. The intuition is that while the Markovian assumption may not directly hold at the image level, the activations of local patterns could still be captured by this assumption. Such a hybrid model could be used to generatively produce new microstructure patches given its surroundings, thus achieving synthesis of complex material samples. Secondly, identification of proper network specifications (e.g., the number of layers, the number of filters, and filter sizes) is non-trivial. In fact, an arbitrary choice of these parameters could lead to abysmal feature extraction and recon-

struction performance. Due to the unknown sensitivity of the performance with respect to the modeling parameters, we cannot claim that the presented model is universally applicable to all material systems, despite the fact that the model worked well on the four demonstrated systems. Lastly, it is also worth noting that the proposed network essentially learns a lower-dimensional manifold from the image space using limited samples. However, it does not provide a feasible region defined on this manifold. As a result, random reconstructions are not guaranteed to be physically meaningful, and a validation through processing-structure mapping is needed. It is therefore desirable that certain known physics-based rules (i.e., constraints on features) can be incorporated into the feature extraction process to eliminate or reduce the chance of creating infeasible reconstructions.

5 Conclusions

This paper presented a novel methodology for feature extraction and reconstruction of complex microstructures through a convolutional deep belief network. We showed that the proposed 5-layer CDBN model, along with some postprocessing steps, enables significant dimension reduction and visually plausible random reconstruction for four different material systems. Specific to the Ti-6Al-4V alloy system, we further showed that the proposed model preserves the mean critical fracture force of material samples derived from the same processing setting after reconstructing these samples.

Acknowledgement

This work has been financially supported by the startup funding from the Arizona State University. This support is gratefully acknowledged. We would also like to thank Dr. Honglak Lee, Dr. Kihyuk Sohn and Ye Liu for their support on the CDBN implementation, and Dr. Wei Chen, Dr. Hongyi Xu, Ramin Bostanabad, Dr. Yang Jiao and Dr. Yongming Liu for valuable discussion. All source code and datasets are available at <https://github.com/DesignInformaticsLab/Material-Design>.

REFERENCES

[1] Sharma, V., Wang, C., Lorenzini, R. G., Ma, R., Zhu, Q., Sinkovits, D. W., Pilania, G., Oganov, A. R., Kumar, S., Sotzing, G. A., et al., 2014. "Rational design of all organic polymer dielectrics". *Nature communications*, **5**.

[2] Baldwin, A. F., Huan, T. D., Ma, R., Mannodi-Kanakkithodi, A., Tefferi, M., Katz, N., Cao, Y., Ramprasad, R., and Sotzing, G. A., 2015. "Rational design of organotin polyesters". *Macromolecules*, **48**(8), pp. 2422–2428.

[3] Ma, R., Sharma, V., Baldwin, A. F., Tefferi, M., Offenhach, I., Cakmak, M., Weiss, R., Cao, Y., Ramprasad, R., and Sotzing, G. A., 2015. "Rational design and synthesis of

polythioureas as capacitor dielectrics". *Journal of Materials Chemistry A*, **3**(28), pp. 14845–14852.

[4] Kalidindi, S. R., and De Graef, M., 2015. "Materials data science: current status and future outlook". *Annual Review of Materials Research*, **45**, pp. 171–193.

[5] Kaczmarowski, A., Yang, S., Szlufarska, I., and Morgan, D., 2015. "Genetic algorithm optimization of defect clusters in crystalline materials". *Computational Materials Science*, **98**, pp. 234–244.

[6] Kirklın, S., Saal, J. E., Hegde, V. I., and Wolverton, C., 2016. "High-throughput computational search for strengthening precipitates in alloys". *Acta Materialia*, **102**, pp. 125–135.

[7] Xu, H., Dikin, D. A., Burkhart, C., and Chen, W., 2014. "Descriptor-based methodology for statistical characterization and 3d reconstruction of microstructural materials". *Computational Materials Science*, **85**, pp. 206–216.

[8] Xu, H., Liu, R., Choudhary, A., and Chen, W., 2015. "A machine learning-based design representation method for designing heterogeneous microstructures". *Journal of Mechanical Design*, **137**(5), p. 051403.

[9] Dosovitskiy, A., and Brox, T., 2015. "Inverting convolutional networks with convolutional networks". *arXiv preprint arXiv:1506.02753*.

[10] Mahendran, A., and Vedaldi, A., 2015. "Understanding deep image representations by inverting them". In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, IEEE, pp. 5188–5196.

[11] Nguyen, A., Yosinski, J., and Clune, J., 2015. "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images". In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, IEEE, pp. 427–436.

[12] Krizhevsky, A., Sutskever, I., and Hinton, G. E., 2012. "ImageNet classification with deep convolutional neural networks". In *Advances in neural information processing systems*, pp. 1097–1105.

[13] Simonyan, K., and Zisserman, A., 2014. "Very deep convolutional networks for large-scale image recognition". *arXiv preprint arXiv:1409.1556*.

[14] Yan, X., Yang, J., Sohn, K., and Lee, H., 2015. "Attribute2Image: Conditional image generation from visual attributes". *arXiv preprint arXiv:1512.00570*.

[15] McDowell, D. L., and Olson, G. B., 2008. "Concurrent design of hierarchical materials and structures". *Scientific Modeling and Simulations*, **15**, pp. 207–240.

[16] Broderick, S., Suh, C., Nowers, J., Vogel, B., Mallapragada, S., Narasimhan, B., and Rajan, K., 2008. "Informatics for combinatorial materials science". *JOM*, **60**(3), pp. 56–59.

[17] Ashby, M. F., and Cebon, D., 1993. "Materials selection in mechanical design". *Le Journal de Physique IV*, **3**(C7), pp. C7–1.

[18] Karasek, L., and Sumita, M., 1996. "Characterization of

- dispersion state of filler and polymer-filler interactions in rubber carbon black composites”. *Mater. Sci.*, **31**, pp. 281–289.
- [19] Rollett, A. D., Lee, S.-B., Campman, R., and Rohrer, G., 2007. “Three-dimensional characterization of microstructure by electron back-scatter diffraction”. *Annu. Rev. Mater. Res.*, **37**, pp. 627–658.
- [20] Borbely, A., Csikor, F., Zabler, S., Cloetens, P., and Biermann, H., 2004. “Three-dimensional characterization of the microstructure of a metal–matrix composite by holotomography”. *Materials Science and Engineering: A*, **367**(1), pp. 40–50.
- [21] Tewari, A., and Gokhale, A., 2004. “Nearest-neighbor distances between particles of finite size in three-dimensional uniform random microstructures”. *Materials Science and Engineering: A*, **385**(1), pp. 332–341.
- [22] Pytz, R., 2004. “Microstructure description of composites, statistical methods, mechanics of microstructure materials”. *CISM Courses and Lectures*.
- [23] Steinzig, M., and Harlow, F., 1999. “Probability distribution function evolution for binary alloy solidification”. In *Solidification, Proceedings of the Minerals, Metals, Materials Society Annual Meeting*, San Diego, CA, Citeseer, pp. 197–206.
- [24] Scaloni, J., Fieller, N., Stillman, E., and Atkinson, H., 2003. “Spatial pattern analysis of second-phase particles in composite materials”. *Materials Science and Engineering: A*, **356**(1), pp. 245–257.
- [25] Torquato, S., 2013. *Random heterogeneous materials: microstructure and macroscopic properties*, Vol. 16. Springer Science & Business Media.
- [26] Sundararaghavan, V., and Zabaras, N., 2005. “Classification and reconstruction of three-dimensional microstructures using support vector machines”. *Computational Materials Science*, **32**(2), pp. 223–239.
- [27] Basanta, D., Miodownik, M. A., Holm, E. A., and Bentley, P. J., 2005. “Using genetic algorithms to evolve three-dimensional microstructures from two-dimensional micrographs”. *Metallurgical and Materials Transactions A*, **36**(7), pp. 1643–1652.
- [28] Holotescu, S., and Stoian, F., 2011. “Prediction of particle size distribution effects on thermal conductivity of particulate composites”. *Materialwissenschaft Und Werkstofftechnik*, **42**(5), pp. 379–385.
- [29] Klaysom, C., Moon, S.-H., Ladewig, B. P., Lu, G. M., and Wang, L., 2011. “The effects of aspect ratio of inorganic fillers on the structure and property of composite ion-exchange membranes”. *Journal of colloid and interface science*, **363**(2), pp. 431–439.
- [30] Gruber, J., Rollett, A., and Rohrer, G., 2010. “Misorientation texture development during grain growth. Part II: Theory”. *Acta Materialia*, **58**(1), pp. 14–19.
- [31] Liu, Y., Greene, M. S., Chen, W., Dikin, D. A., and Liu, W. K., 2013. “Computational microstructure characterization and reconstruction for stochastic multiscale material design”. *Computer-Aided Design*, **45**(1), pp. 65–76.
- [32] Quiblier, J. A., 1984. “A new three-dimensional modeling technique for studying porous media”. *Journal of Colloid and Interface Science*, **98**(1), pp. 84–102.
- [33] Jiang, Z., Chen, W., and Burkhart, C., 2013. “Efficient 3D porous microstructure reconstruction via gaussian random field and hybrid optimization”. *Journal of microscopy*, **252**(2), pp. 135–148.
- [34] Grigoriu, M., 2003. “Random field models for two-phase microstructures”. *Journal of Applied Physics*, **94**(6), pp. 3762–3770.
- [35] Bostanabad, R., Bui, A. T., Xie, W., Apley, D. W., and Chen, W., 2016. “Stochastic microstructure characterization and reconstruction via supervised learning”. *Acta Materialia*, **103**, pp. 89–102.
- [36] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al., 2012. “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups”. *Signal Processing Magazine, IEEE*, **29**(6), pp. 82–97.
- [37] Krizhevsky, A., Sutskever, I., and Hinton, G. E., 2012. “ImageNet classification with deep convolutional neural networks”. In *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds. Curran Associates, Inc., pp. 1097–1105.
- [38] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M., 2013. “Playing Atari with deep reinforcement learning”. *arXiv preprint arXiv:1312.5602*.
- [39] Schmidhuber, J., 2015. “Deep learning in neural networks: An overview”. *Neural Networks*, **61**, pp. 85–117.
- [40] Levine, S., Finn, C., Darrell, T., and Abbeel, P., 2015. “End-to-end training of deep visuomotor policies”. *arXiv preprint arXiv:1504.00702*.
- [41] Reed, S. E., Zhang, Y., Zhang, Y., and Lee, H., 2015. “Deep visual analogy-making”. In *Advances in Neural Information Processing Systems*, pp. 1252–1260.
- [42] Lee, H., Grosse, R., Ranganath, R., and Ng, A. Y., 2009. “Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations”. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ACM, pp. 609–616.
- [43] Bousquet, O., and Bottou, L., 2008. “The tradeoffs of large scale learning”. In *Advances in neural information processing systems*, pp. 161–168.
- [44] Rumelhart, D. E., Hinton, G. E., and Williams, R. J., 1988. “Learning representations by back-propagating errors”. *Cognitive modeling*, **5**(3), p. 1.

- [45] Hinton, G. E., 2002. "Training products of experts by minimizing contrastive divergence". *Neural computation*, **14**(8), pp. 1771–1800.
- [46] Bengio, Y., 2009. "Learning deep architectures for AI". *Foundations and trends® in Machine Learning*, **2**(1), pp. 1–127.
- [47] Yumer, M. E., Asente, P., Mech, R., and Kara, L. B., 2015. "Procedural modeling using autoencoder networks". In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, ACM, pp. 109–118.
- [48] Kingma, D. P., and Welling, M., 2013. "Auto-encoding variational Bayes". *arXiv preprint arXiv:1312.6114*.
- [49] Sohn, K., and Lee, H., 2012. "Learning invariant representations with local transformations". *arXiv preprint arXiv:1206.6418*.
- [50] scikit image. scikit-image axis skeletonization.
- [51] Chen, H., Lin, E., Jiao, Y., and Liu, Y., 2014. "A generalized 2d non-local lattice spring model for fracture simulation". *Computational Mechanics*, **54**(6), pp. 1541–1558.

Appendix: Sample Microstructure Images

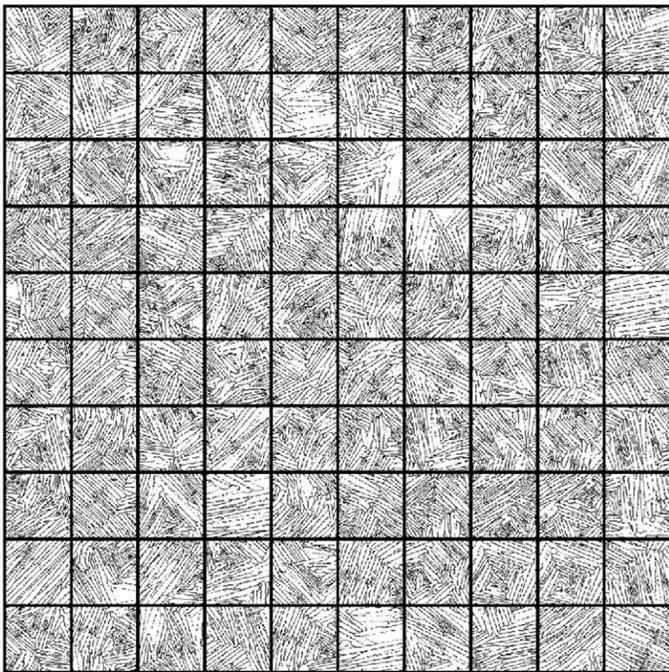


FIGURE 10. Ti-6Al-4V alloy

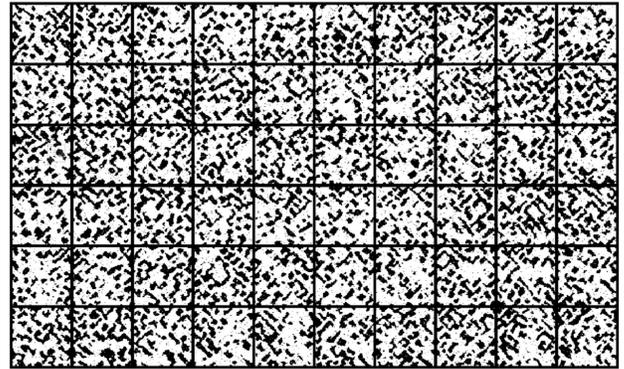


FIGURE 11. Pb-Sn (lead-tin) alloy

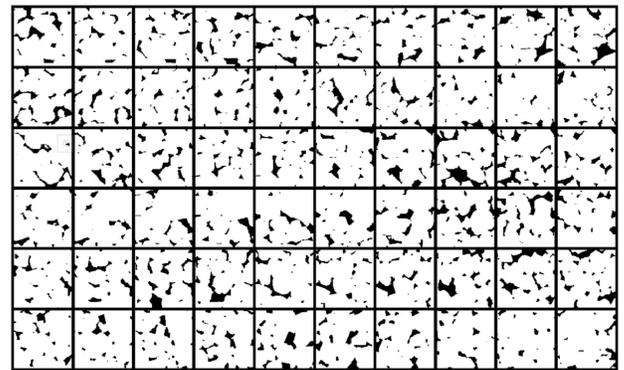


FIGURE 12. Pore structure of Fontainebleau sandstone

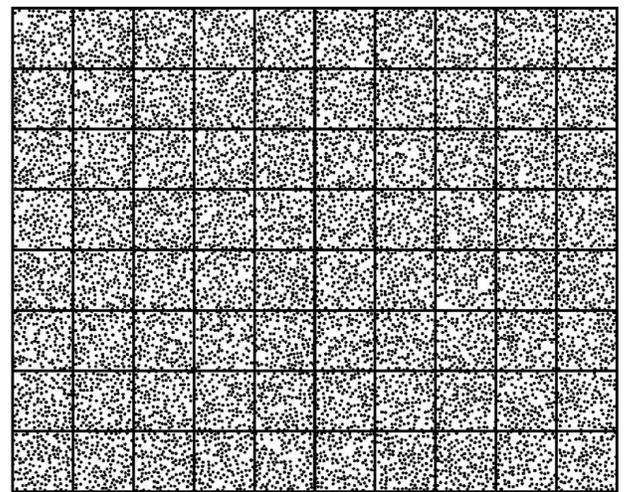


FIGURE 13. 2D suspension of spherical colloids