TextureVAE : Learning Interpretable Representations of Material Microstructures Using Variational Autoencoders

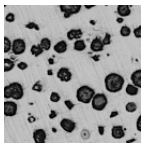
Abstract

We propose a variational autoencoder model based on style loss for learning representations of material microstructure images. We show using latent space traversals that the model captures important attributes of microstructures that are responsible for mechanical properties of materials and is capable of generating microstructures with particular attributes. We discuss how the latent vectors can be used to establish a linkage between structure and properties and enable inverse inference which is crucial for designing materials and products with target properties.

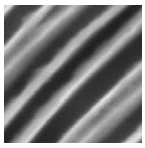
1 Introduction

When a material is put through a manufacturing process, it's internal structure is modified, which in turn affects the properties of the material. Materials scientists seek answers to questions such as what processing is required to achieve the target properties and how do the structure and properties change with the process parameters. It is well known that these mappings are complex, highly non-linear and multiple processing paths can lead to the same property. These mappings can be best described through the space of structures (Kalidindi 2015). The most commonly available description of the structure is in the form of microscopic images, which is known as the *microstructure* (because the length scale is roughly $10^{-6}m$). Obtaining compact representations of microstructure images [1] is therefore crucial for building robust process-structure-property linkages.

The microstructure contains a lot of information such as grain size distribution, volume fractions of different phases and so on. Depending upon the material system under consideration, very different types of features and information is relevant. Traditionally, materials scientists have used statistical methods such as n-point correlation functions and Gaussian random fields for obtaining representations of microstructures. The n-point correlation functions capture the degree of spatial correlation among the locations and constituents in a probabilistic sense (Kalidindi 2015). For example, given a microstructure containing two phases, the 2-point statistics can be used to encode the probability that a random vector of length r has both its ends in the same phase. These are the most widely used correlation functions for formal mathematical characterization of microstructures. However, it has been shown that different microstructures may, under some conditions, lead to similar 2-point correlations (Cang et al. 2018). And beyond n = 2 (i.e., higher-order spatial correlations), n-point statistics quickly become intractable. Hence, these methods are not easily extensible in general.



(a) Cast iron



(b) Ultra-high carbon steel

Figure 1: Example Microstructures - The information of interest varies with the material system under consideration. (Credit: (a) Tewary et al. [to be published], (b) (DeCost et al. 2017)

Another common alternative is using physical descriptors. For example, consider the two microstructures in Figure 1. The first one is a cast iron microstructure containing spherical grains of graphite, and the ferrite phase in background. Here, some physical descriptors of interest are the sizes of the spherical grains, their density and so on. Whereas the second one is an ultra-high carbon steel microstructure containing pearlite phase as an alternating layer with the ferrite phase. Here the important physical descriptors are the orientation of the lamellar pattern, the inter-lamellar spacing and so on. One needs to employ specific image processing to extract physical descriptors from given microstructures. Also, such a representation is specific to the material system and is tightly tied to the expertise of a materials scientist in selecting the right descriptors.

Researchers are looking at the recent methods of representation learning from the deep learning literature as an alternative to the above-mentioned microstructure representation methods. A major challenge in leveraging these techniques is the scarcity of data. Transfer learning can be used to mitigate this up to some extent. However, a further challenge is that it is even more difficult to get microstructures

[1] For simplicity, henceforth referred to as just microstructure

with associated property values, owing to the high cost and time required to do the testing. Deep generative models such as the variational autoencoders aim to learn good latent representations in an unsupervised manner. In absence of a supervisory signal, these models try to learn latent representations from which the original inputs can be reconstructed most accurately. Such a representation is expected to encode all non-redundant information from the image. Further, these models are capable of synthesizing images that are realistic and statistically equivalent to the training images. Synthesis is a common goal to support computational design because the cost and difficulty of experimental characterization is often prohibitively high (Hsu et al. 2020). Inspired by this, we propose a variational autoencoder architecture to learn low-dimensional microstructure representations. We demonstrate that the learned latent representation indeed encodes important features, with a use-case in which such features are known in advance. The representation can be physically interpreted in that individual latent dimensions correspond to different features which are known to be important from the physics knowledge. Such a representation is therefore expected to work well for modeling the structure-property linkages.

Our key contribution is an interpretable microstructure representation method that

- captures physically significant factors of variation which are primarily responsible for the mechanical properties of the material
- can be used to generate different microstructures by varying these factors

2 Related work

With the recent advances in machine learning, there is a renewed interest among materials scientists to leverage these advances for material microstructure modeling. Bostanabad et al. (Bostanabad et al. 2018) provides a detailed review of the state-of-the-art in computational characterization of material microstructure. We discuss some of the more recent works on application of deep learning for this task.

In some recent works, generative adversarial nets (GANs) and variational autoencoders have been used for material microstructure generation. Often the focus is on generation rather than representation learning. For example, (Banko et al. 2020) use a conditional GAN to generate microstructures of thin films conditioned on process parameters and chemical composition. While Hsu et al. (Hsu et al. 2020) use GANs to generate small patches of 3D microstructure of solid oxide fuel cell anodes. They show that the properties computed by numerical simulations on the generated microstructures closely match the experimental observations. Chun et al. (Chun et al. 2020) use a patch-based GAN to generate microstructures of heterogeneous energetic materials (propellants explosives and pyrotechnics). The input to their model consists of a pair of vectors for each grid location (patch). They show that during generation, the two vectors can be used to control overall morphology. However the intuitive meanings of individual dimensions of these vectors are not clear and the authors point to this as possible further work. Liu et al. (Liu et al. 2015) propose a design method for inferring structures with target properties using Bayesian optimization around the GAN generator. The authors talk about the possibility of using the implicitly learned representation from the GAN discriminator for a structure-property model, but do not present any study on this. Probably the closest to our work is Cang et al. (Cang et al. 2018), where a variational autoencoder model with style loss is proposed for generating microstructures of sandstone. The authors show that the generated microstructures are more predictive of the properties (Young's modulus, diffusivity and permeability) than those generated using Gaussian random field method. They add the style loss to the vanilla VAE objective function, retaining the original reconstruction loss. However, the vanilla VAE reconstruction loss is not suitable for microstructure images (see section 3.2 for more details). So we completely replace the reconstruction loss with the style loss. We also show that physically significant factors of variation are explicitly encoded in the learned representation. To the best of our knowledge, ours is the first work on a variational autoencoder model for microstructure generation and interpretable representation learning.

3 Methodology

3.1 Variational Autoencoders

Variational Autoencoders (Kingma and Welling 2013) are typically used to learn latent representations of input samples in an unsupervised manner. The underlying graphical model is $p_\theta(x, z) = p_\theta(z)p_\theta(x|z)$, where x is the observed variable (input sample) and z is latent variable (the representation). Given an input sample, the latent variables can be inferred from the posterior $p(z|x)$. Computing this distribution is a hard problem due to the intractable partition function required in applying Bayes' theorem. In variational inference, an approximate posterior distribution $q_\phi(z|x)$ from a known family is found by minimizing the KL divergence from the true posterior. That is, find q^* such that

$$q^* = argmin_{q_\phi} D_{KL}(q_\phi(z|x)\|p_\theta(z|x))$$

However, computing the KL divergence also involves the same intractable integrals as in posterior computation. So instead of minimizing it, another tractable quantity called the Evidence Lower Bound (ELBO) derived from the above equation is maximized and it is shown that maximizing ELBO is equivalent to minimizing the KL divergence. The ELBO is defined as:

$$\mathcal{L} = -D_{KL}(q_\phi(z|x)\|p_\theta(z)) + \mathbb{E}_{q_\phi(z|x)}\left[log\ p_\theta(x|z)\right] \quad (1)$$

The above loss function can be roughly understood as follows : The second term is the expected log-likelihood of getting back the same x starting with the inferred z from the approximate posterior $q_\phi(z|x)$. It is often called the *reconstruction loss*. The first term is a regularizer that penalizes posteriors very different from the prior.

The prior $p_\theta(z)$ and the posterior $q_\phi(z|x)$ are generally assumed to be Gaussian. The distribution q_ϕ is parameterized by an inference network and resembles an encoder. It outputs the μ and σ of the posterior for a given input sample (i.e. $q_\phi(z|x_i) = N(z; \mu_{x_i}, \sigma_{x_i})$). The distribution p_θ is

parameterized by a generator network and resembles a decoder. It outputs a sample from the distribution p_θ given a latent vector z.

3.2 Texture-VAE

However, the reconstruction loss in the original VAE objective is not suitable to model the mismatch between a microstructure image and its reconstruction. We discuss the reasons as follows. In the decoder network, the reconstruction $\hat{x}$ is a deterministic function of z. Now consider the reconstruction loss : $\mathbb{E}_{q_\phi(z|x)}\left[log\, p_\theta(x|z)\right]$. With continuous valued outputs such as images, the generator distribution is generally assumed to be Gaussian whose mean $\mu = f(z;\theta)$ is computed by the generator network. Thus, $\mu = \hat{x}$. Since the reconstruction loss is an expectation of log of a Gaussian density, it is equivalent to $(x-\mu)^2$ (i.e. $(x-\hat{x})^2$) up to some constants. Hence the reconstruction loss is equivalent to pixel-wise mean squared error. It has been argued in many works that pixel-wise comparison is not capable of capturing perceptual image similarity (see for example (Ding et al. 2020), (Dosovitskiy and Brox 2016) or (Larsen et al. 2016)). This is especially true in case of microstructures, which are a type of texture images (they contain randomly repeating patterns such as spheres, lines and so on). Imagine a stripes pattern and another one, shifted one stripe right. A human instantly understands that they are essentially "the same texture", but the pixel-wise difference could be huge. To summarize, the reconstruction loss term in equation 1 is not suitable for texture images since it leads to a pixel-by-pixel comparison between input and reconstructed image. This motivates replacing the reconstruction loss with a better suited measure for textures.

Since textures are different from natural images that generally contain objects, special considerations are needed for representing textures. In texture synthesis literature, Gatys et al. (Gatys, Ecker, and Bethge 2015) proposed to use the feature correlations computed from different layers of a pre-trained network (e.g. VGG19) to represent textures. The feature correlations at layer l are encoded by the Gram matrix $G^{(l)}$, whose elements are inner products between feature maps at that layer. If layer l has C_l feature maps of size $W_l \times H_l$ then:

$$G^l_{ij} = \Sigma_k F^l_{ik} F^l_{jk}$$

Where, $M_l = W_l * H_l$ and F^l is the $C_l \times M_l$ matrix with the flattened feature maps as rows. A texture can then be represented by the concatenation of all Gram matrices. Given an input texture image, the authors propose a method to generate similar textures by starting with a random white noise image and minimizing the squared difference between the representations. Note that the optimization is with respect to the image pixels; the weights of the pre-trained network are not changed. They call the squared difference between Gram matrix concatenations as the "style loss":

$$\mathcal{L}_{style}(x,\hat{x}) = \sum_{l=0}^{L} w_l \left[\frac{1}{4C_l^2 M_l^2} \sum_{i,j} (G^l_{ij} - \hat{G}^l_{ij})^2 \right] \quad (2)$$

While this approach of "generation-by-optimization" was a major improvement over state-of-the-art in texture synthesis, it required an optimization step for each generation. This method has been extended to feed-forward approaches that learn a separate generator network to minimize the style loss, for example (Johnson, Alahi, and Fei-Fei 2016), (Ulyanov, Vedaldi, and Lempitsky 2017) and (Li et al. 2017). The generator transforms a random input vector (typically a standard Gaussian) into texture images. However, these methods are not well suited for representation learning because they focus on generation. Most of these methods require learning one network per texture (or per style). The difficulty in using the same network for multiple styles seems to stem from the fact that the Gram matrices of different styles have very different scales and the generator adapts itself to a particular style. There have been some works on methods capable of learning multiple styles/textures with a single model and they seem to focus on normalizing the styles and forcing a correlation between the random input and generated image. Variational autoencoders naturally address this issue with the encoder learning different representations for different styles. Lastly, while the concatenation of Gram matrices characterizes a texture well, it may not be of much use as a representation vector by itself for other downstream tasks because it is very high dimensional (often, more dimensions than the image itself).

Combining these ideas, we propose using the style loss to train a variational autoencoder. In particular we replace the reconstruction loss (i.e. the second term) in equation 1 with the style loss (given in equation 2). We keep the first term as it is - this regularizes the latent representation by forcing all posteriors to be not far from the prior. With a standard Gaussian prior with diagonal covariance matrix, this term encourages a representation with statistically independent dimensions, which is expected to be interpretable. This is discussed in more detail in section 4.3. We call our model the texture-VAE. We show that a single VAE model trained with style loss can be used for multiple textures.

3.3 Model architecture

The architecture of our VAE model is shown in Figure 2. We use the pre-trained VGG19 network (Simonyan and Zisserman 2014) for computing the style loss and as the encoder. The last two fully connected layers of the encoder that compute μ and σ of the posterior are trained from scratch. For the remaining layers, the pre-trained weights are used without fine-tuning. The decoder contains four blocks of deconvolution followed by nearest-neighbor upsampling and LeakyReLU nonlinearity with slope 0.3. It has been previously observed in literature that for generation, explicit upsampling works better than using fractional strided convolutions (Odena, Dumoulin, and Olah 2016). We use a filter size of 3×3 throughout.

For computing the style loss, we use the pooling layer at each scale, i.e. pool_1 to pool_4 and conv_1_1. Gatys et al. (Gatys, Ecker, and Bethge 2015) recommend using the convolutional layers instead of pooling. But we found that in our case, pooling layers worked better.

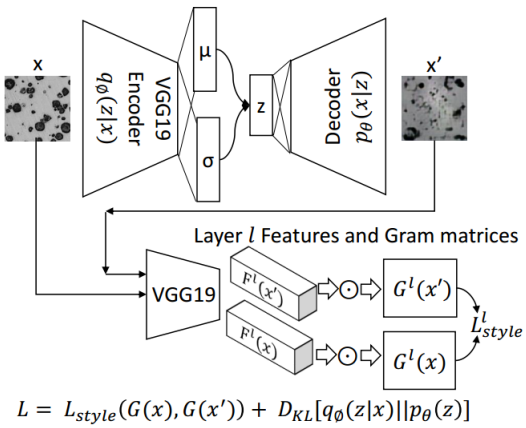


Figure 2: The variational autoencoder model with style loss. L_{style} is the sum of squared differences between Gram matrices at all layers and $\odot$ denotes the inner product of feature maps $(F^l)^T F^l$

4 Experiments

4.1 Dataset

We use microstructures of cast iron to demonstrate the capabilities of our texture-VAE model. These microstructures were produced in a separate study by Ujjal Tewary et al. (to be published) of microstructure and mechanical properties of cast iron produced using sand casting. In the process of sand casting, molten iron ore mixed with C, Si, Mg etc. is poured into molds of desired size and kept in sand for cooling. In case of cylindrical molds, the sample cools from the surface to its core, so the radius governs the cooling rate. In the study, cylindrical castings of various radii (resulting in different cooling rates) were made using sand casting. Microstructures of these cylindrical castings were then captured using a scanning electron microscope. The original study consisted of many experiments but we describe here only those that correspond to the microstructures we used.

We use the microstructures of 12 samples resulting from combination of four cooling rates corresponding to cylinders of radius 12, 24, 36 and 48mm and three compositions (mainly varying Magnesium - 0, 0.025 and 0.045% weight). The images were captured at $100\mu m$ length scale, without any etching. The microstructures mainly contain ferrite and graphite phases. The cooling rate and initial composition affect the grain size, density and morphology (i.e. appearance of the graphite - spherical, flaky or both) of the resulting microstructure. Figure 3 shows small 128x128 patches from a few microstructures. We have chosen these so that the variations in grain size (small to large), density (low to high) and morphology (spherical, flakes and intermediate) can be clearly seen.

Out of the 12, 6 samples corresponding to the lowest and highest cooling rates were also subjected to uni-axial compression to obtain their stress-strain behavior. We used all 12 microstructures to train the texture-VAE model while the 6 with property values were used for property prediction task described in section 4.4.

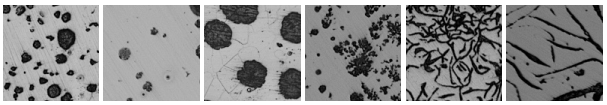


Figure 3: Example patch from each microstructure. (a) small dense spheres, (b) small sparse spheres, (c) large spheres, (d) intermediate (e) fine flakes and (f) thick flakes

Table 1: Average similarity between reconstructed and original patches

Latent Dims	Score	
	DISTS	STSIM-2
64	0.7031	0.7592
32	0.7257	0.7640
16	0.6995	0.7609
8	0.6962	0.7591

The original microstructures were 2048x1532. We use patches of 128x128 cropped by sliding a window with stride 50 for our training.

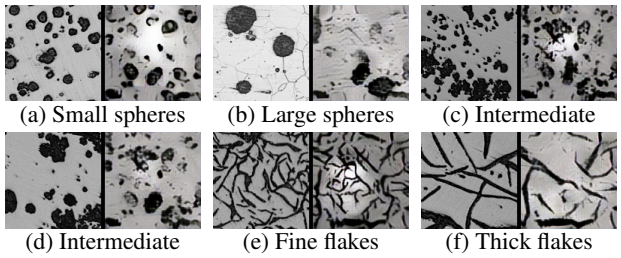
4.2 Evaluation

While there are many metrics of perceptual image similarity, very few of them are focused on texture images. Two recently proposed metrics of texture similarity that seem to be best suited for our evaluation are - i) Deep Image Structure and Texture Similarity - DISTS (Ding et al. 2020) and ii) Structural Texture Similarity - STSIM (Ehmann, Pappas, and Neuhoff 2013). The DISTS score consists of two terms - first one compares the means of features maps (from a variant of pre-trained VGG) and the second one computes cross covariance between them. The two terms are combined using weights tuned to match human judgments and be invariant to re-sampled patches from the same texture. The STSIM metric is based on a similar modification of Structural Similarity Metric (SSIM) to completely avoid pixel-by-pixel comparison, but is computed in the Fourier spectrum. We used the authors' implementation of DISTS[2]. STSIM has several configurations and we compute the STSIM-2 metric using a publicly available implementation[3]. Table 1 shows the average similarity over 100 instances between original patches and reconstructions from texture-VAE models with number of latent vector dimensions 8, 16, 32 and 64. The similarity scores are on a scale of 0 to 1, 1 being the highest. The similarity seems to slightly increase with increasing number of latent dimensions, but beyond 64 it didn't increase, so we stopped there.

One way to qualitatively evaluate a VAE model is to look at the reconstructed and newly generated samples. Figure 4 shows some example reconstructions while Figure 5 shows some randomly generated samples from texture-VAE model with 64 latent dimensions. Although the latent-32 model had slightly higher similarity scores for reconstructions than

[2]https://github.com/dingkeyan93/DISTS

[3]https://github.com/andreydung/Steerable-filter



(a) Small spheres (b) Large spheres (c) Intermediate

(d) Intermediate (e) Fine flakes (f) Thick flakes

Figure 4: Example reconstructions

(a) (b) (c) (d) (e)

Figure 5: Randomly generated samples

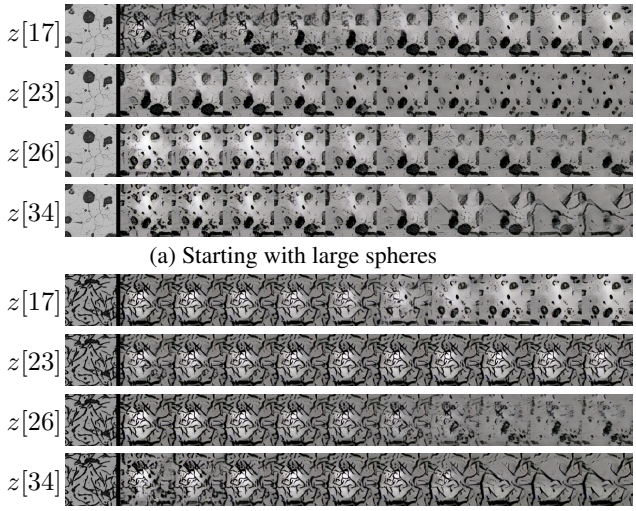
latent-64, the latter recovered finer details better, which are important from the domain perspective (for example the ferrite grain boundaries, explained in the next paragraph). Hence we did all further experimentation with the latent-64 model. The left half of each image in Figure 4 is a patch from the original microstructure - x, while the right half is the reconstructed image - $\hat{x} = Dec(Enc(x))$. We have shown some representative examples from each microstructure. It can be seen that even the minute structural details such as the ferrite grain boundaries (the thin lines in the gray portion) which are faintly visible only in the case of large spherical grains in Figure 4b are reconstructed quite well.

These results show that the texture-VAE model is capable of reconstructing input samples quite well across different textures. The randomly generated samples also span different textures and look structurally similar to the original ones.

4.3 Interpretability

Variational autoencoders have been shown to recover factors of variation in the training data ((Kingma and Welling 2013), (Higgins et al. 2017)). The first term in the learning objective of VAE encourages the posterior $p_\theta(z|x)$ to be like the prior $p(z)$ which is a standard normal distribution with diagonal covariance matrix. That is, this term encourages the latent dimensions to be statistically independent (Higgins et al. 2017). Such representations are easier to interpret and can be more useful in downstream tasks ((Ridgeway 2016) and (Bengio, Courville, and Vincent 2013)). We perform experiments to show that the texture-VAE model recovers physically significant factors of variation.

Starting with an image x, we obtain its latent representation $z = Enc(x)$. Then we choose a dimension i of z and vary it in the range $[-4, 4]$ by choosing 10 equally spaced values, while keeping all other dimensions unchanged. That is, $z'[i] = j,\ \ j \in linspace(-4, 4, 10)$ and $z'[k] = z[k]$ for all other dimensions k. By decoding these z' vectors, we observe the variations in the image space. Figure 6 shows images of two examples obtained by varying dimensions 17, 23, 26 and 34. These dimensions were chosen for illustration since they seem to produce physically significant variations that are visually discernible as well. Figure 6a shows the variations starting with a large-spheres microstructure, whereas Figure 6b shows the variations starting with a fine-flakes microstructure. From the figure, dimension 17 seems to correspond to morphology, with lower values indicating flaky and higher values indicating spherical structures. Dimensions 23 and 26 seem to correspond to the density and size respectively of spherical grains, with their values increasing as we go from left to right. Whereas dimension 34 seems to correspond to the density of flakes. From the physics of Cast iron microstructures, it is known that grain size and density are correlated - when the spherical grains are large (or the flakes are thick), they are more likely to be sparse. This correlation seems to be well-captured in the variations of dimensions 23, 26 and 34.



(a) Starting with large spheres

(b) Starting with fine flakes

Figure 6: Effect of varying latent dimensions, starting with different morphologies. Each row corresponds to one latent dimension. The leftmost image is the original patch and the remaining are variations obtained by varying that particular latent dimension

4.4 Structure-Property linkage

As shown in Figure 6, for cast iron microstructures, some of the latent dimensions seem highly correlated with quantities such as grain size, morphology, grain density and so on. It is known that these factors have a profound impact on the mechanical properties of cast iron. For example, the spherical grains prevent a passing crack from further propagating, so lead to higher strength. Whereas the flakes deflect the crack into a number of other directions, so lead to brittleness. Consequently, the representation is expected to lend itself to a more accurate property prediction model. In the following, we describe some experiments that support this claim.

As stated earlier, the stress-strain curves of 6 microstructures corresponding to the smallest and largest cooling rates were available, from which we obtained ultimate tensile

Table 2: Property prediction accuracy

Method	UTS		Ys	
	R^2	MAPE	R^2	MAPE
LinReg	0.89	20	0.88	14
SVR-RBF	0.97	10	0.98	5

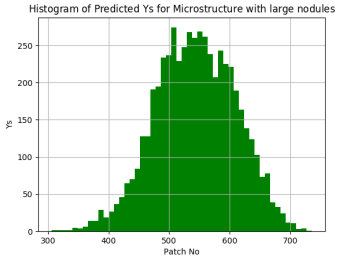


Figure 7: Prediction of Ys for an unseen microstructure

strength and yield strength. We trained a regression model from the latent representations of patches of these microstructures to the property values. Note that the property values correspond to original full-size microstructure images, whereas our model's input size is 128x128. We assume that all 128x128 patches cropped from the same microstructure image have the same property value. A validation set containing 20% of the patches was kept aside for evaluation. Table 2 shows the R^2 value and mean absolute percentage error in prediction of ultimate tensile strength (UTS) and yield strength (Ys) on the validation set. The table shows that we get reasonably good accuracy even with a simple linear regression model, revealing that the learned representation is highly predictive of the properties. With a more expressive model such as support vector regression (with a radial basis kernel) the accuracy goes significantly higher, further strengthening the belief in the predictive power of the representation.

To test generalization, we trained the SVR model for yield strength using only five microstructures and used it to predict the yield strength for the sixth microstructure. Note that this is different from the above experiment on the validation set. Here, the regression model does not see any patches (and the property values) from the excluded microstructure. The missing microstructure corresponds to the lowest cooling rate which results in the largest spherical grains. Figure 7 shows the histogram of predicted values on all patches of this microstructure. It can be seen that the mean prediction is near $550Mpa$. The true value found from experiments is $598Mpa$, so the prediction is off by about 8% not deviating a lot from the 5% error on the validation set. We performed the same experiment using latent representations obtained from unmodified, pre-trained VGG19 network. Table 3 shows that the texture-VAE representations generalize much better as compared to pre-trained VGG19. We think that the reason behind better generalization with our representation is that it encodes physically significant attributes.

Table 3: Prediction of Ys - Generalization

Method	MAPE
TextureVAE + SVR	8.02
VGG19 + SVR	18.22

5 Conclusion

We have presented a variational autoencoder model to learn microstructure representations. The objective function is obtained by replacing the reconstruction loss in the vanilla VAE with the style loss. We applied the model on a set of experimental cast iron microstructures. Through latent space traversals, we showed that the learned representation explicitly encodes factors of variation that are primarily responsible for the mechanical properties (such as ultimate tensile strength and yield strength). Consequently, the representation is highly predictive of mechanical properties. We showed that the regression model built using these representations can reasonably predict the properties for totally unseen morphologies.

Since the learned representation is predictive of mechanical properties and some of its dimensions can be physically interpreted, we expect that it can be used for inverse inference as well - i.e. predicting the structure required to get desired properties. A probabilistic model such as Bayesian network that can represent the joint distribution between latent dimensions and properties can be used to infer the most probable values of the latent dimensions given the properties. The obtained latent vector can then be decoded using the VAE model to give the required microstructure. This is a direction we are pursuing as further work. We believe the present work is a step towards a general framework for learning interpretable microstructure representations.

References

Banko, L.; Lysogorskiy, Y.; Grochla, D.; Naujoks, D.; Drautz, R.; and Ludwig, A. 2020. Predicting structure zone diagrams for thin film synthesis by generative machine learning. *Communications Materials* 1(1): 15. ISSN 2662-4443. doi:10.1038/s43246-020-0017-2. URL https://doi.org/10.1038/s43246-020-0017-2.

Bengio, Y.; Courville, A.; and Vincent, P. 2013. Representation Learning: A Review and New Perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* 35(8): 17981828. ISSN 0162-8828. doi:10.1109/TPAMI.2013.50. URL https://doi.org/10.1109/TPAMI.2013.50.

Bostanabad, R.; Zhang, Y.; Li, X.; Kearney, T.; Brinson, L.; Apley, D.; Liu, W.; and Chen, W. 2018. Computational microstructure characterization and reconstruction: Review of the state-of-the-art techniques. *Progress in Materials Science* 95: 1–41.

Cang, R.; Li, H.; Yao, H.; Jiao, Y.; and Ren, Y. 2018. Improving direct physical properties prediction of heterogeneous materials from imaging data via convolutional neural network and a morphology-aware generative model. *Computational Materials Science* 150: 212 – 221. ISSN

0927-0256. doi:https://doi.org/10.1016/j.commatsci.2018.03.074. URL http://www.sciencedirect.com/science/article/pii/S0927025618302337.

Chun, S.; Roy, S.; Nguyen, Y. T.; Choi, J. B.; Udaykumar, H. S.; and Baek, S. S. 2020. Deep learning for synthetic microstructure generation in a materials-by-design framework for heterogeneous energetic materials. *Scientific reports* 10(1): 13307–13307. ISSN 2045-2322. doi:10.1038/s41598-020-70149-0. URL https://pubmed.ncbi.nlm.nih.gov/32764643. 32764643[pmid].

DeCost, B. L.; Hecht, M. D.; Francis, T.; Webler, B. A.; Picard, Y. N.; and Holm, E. A. 2017. UHCSDB: Ultra-High Carbon Steel Micrograph DataBase. *Integrating Materials and Manufacturing Innovation* 6(2): 197–205. URL https://doi.org/10.1007/s40192-017-0097-0.

Ding, K.; Ma, K.; Wang, S.; and Simoncelli, E. P. 2020. Image Quality Assessment: Unifying Structure and Texture Similarity. *CoRR* abs/2004.07728. URL https://arxiv.org/abs/2004.07728.

Dosovitskiy, A.; and Brox, T. 2016. Generating Images with Perceptual Similarity Metrics based on Deep Networks. In Lee, D.; Sugiyama, M.; Luxburg, U.; Guyon, I.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 29, 658–666. Curran Associates, Inc. URL https://proceedings.neurips.cc/paper/2016/file/371bce7dc83817b7893bcdeed13799b5-Paper.pdf.

Ehmann, J.; Pappas, T.; and Neuhoff, D. 2013. Structural Texture Similarity Metrics for Image Analysis and Retrieval. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society* 22. doi:10.1109/TIP.2013.2251645.

Gatys, L.; Ecker, A. S.; and Bethge, M. 2015. Texture Synthesis Using Convolutional Neural Networks. In Cortes, C.; Lawrence, N. D.; Lee, D. D.; Sugiyama, M.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 28*, 262–270. Curran Associates, Inc. URL http://papers.nips.cc/paper/5633-texture-synthesis-using-convolutional-neural-networks.pdf.

Higgins, I.; Mattheyand, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; and Lerchner, A. 2017. β-VAE: Learning Basic Visual Concepts With a Constrained Variational Framework. *ICLR* URL https://openreview.net/pdf?id=Sy2fzU9gl.

Hsu, T.; Epting, W. K.; Kim, H.; Abernathy, H. W.; Hackett, G. A.; Rollett, A. D.; Salvador, P. A.; and Holm, E. A. 2020. Microstructure Generation via Generative Adversarial Network for Heterogeneous, Topologically Complex 3D Materials. *arXiv e-prints* arXiv:2006.13886.

Johnson, J.; Alahi, A.; and Fei-Fei, L. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*.

Kalidindi, S. R. 2015. 1 - Materials, Data, and Informatics. In *Hierarchical Materials Informatics*, 1 – 32. Boston: Butterworth-Heinemann. ISBN 978-0-12-410394-8.

Kingma, D. P.; and Welling, M. 2013. Auto-Encoding Variational Bayes. *ICLR* abs/1312.6114.

Larsen, A. B. L.; Snderby, S. K.; Larochelle, H.; and Winther, O. 2016. Autoencoding beyond pixels using a learned similarity metric. volume 48 of *Proceedings of Machine Learning Research*, 1558–1566. New York, New York, USA: PMLR. URL http://proceedings.mlr.press/v48/larsen16.html.

Li, Y.; Fang, C.; Yang, J.; Wang, Z.; Lu, X.; and Yang, M. 2017. Diversified Texture Synthesis with Feed-Forward Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 266–274.

Liu, R.; Kumar, A.; Chen, Z.; Agrawal, A.; Sundararaghavan, V.; and Choudhary, A. 2015. A predictive machine learning approach for microstructure optimization and materials design. In *Nature Scientific Reports*, volume 5.

Odena, A.; Dumoulin, V.; and Olah, C. 2016. Deconvolution and Checkerboard Artifacts. *Distill* URL http://distill.pub/2016/deconv-checkerboard/.

Ridgeway, K. 2016. A Survey of Inductive Biases for Factorial Representation-Learning. *CoRR* abs/1612.05299. URL http://arxiv.org/abs/1612.05299.

Simonyan, K.; and Zisserman, A. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* abs/1409.1556.

Ulyanov, D.; Vedaldi, A.; and Lempitsky, V. S. 2017. Improved Texture Networks: Maximizing Quality and Diversity in Feed-Forward Stylization and Texture Synthesis. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 4105–4113. IEEE Computer Society. doi:10.1109/CVPR.2017.437. URL http://doi.ieeecomputersociety.org/10.1109/CVPR.2017.437.