ARTICLE OPEN

Check for updates

Compact representations of microstructure images using triplet networks

Michiel Larmuseau [1,2,3 ✉], Michael Sluydts[1,2], Koenraad Theuwissen[4], Lode Duprez[4], Tom Dhaene[3] and Stefaan Cottenier[1,2]

The microstructure of a material, typically characterized through a set of microscopy images of two-dimensional cross-sections, is a valuable source of information about the material and its properties. Every pixel of the image is a degree of freedom causing the dimensionality of the information space to be extremely high. This makes it difficult to recognize and extract all relevant information from the images. Human experts circumvent this by manually creating a lower-dimensional representation of the microstructure. However, the question of how a microstructure image can be best represented remains open. From the field of deep learning, we present triplet networks as a method to build highly compact representations of the microstructure, condensing the relevant information into a much smaller number of dimensions. We demonstrate that these representations can be created even with a limited amount of example images, and that they are able to distinguish between visually very similar microstructures. We discuss the interpretability and generalization of the representations. Having compact microstructure representations, it becomes easier to establish processing–structure–property links that are key to rational materials design.

INTRODUCTION

Every material has its own combination of physical properties such as hardness, toughness and ductility. These properties determine the industrial applications for which the material is suited. By improving the properties of the material, its performance for the application increases, potentially leading to higher efficiencies, longer lifetimes and an overall reduction in costs. Thus, the discovery of new materials with enhanced properties is a key driver of technological advancement. To discover new materials in a systematic way, it is necessary to model the properties of the material, corresponding to a given composition and set of processing parameters. Although it is possible to directly link these parameters to the properties of the material, there are two main drawbacks to this approach. First, data on properties of tailor-made metals is scarce, as these metals are expensive to produce in small quantities. This makes it difficult to build robust models. Second, the obtained model would be highly specific for the industrial equipment that is used to make the metals. Furthermore, it is known that the structure of the metal at a small scale, the microstructure, determines the properties of the material and in its turn is strongly affected by the composition and processing conditions[1]. Therefore, the microstructure, which is typically characterized by a set of microscopy images of prepared surfaces of the metal, is an essential link between the processing and the properties of the material. In the literature, this is designated as processing–structure–properties links[2,3]. Although images are an important source of microstructure information, raw image data are too complex to directly link to properties. For this reason, the information in the images must be condensed in a more simple, compact representation.

The question of how a microstructure can be best represented has been investigated by metallurgists for decades[4]. In the past years, inspiration has come from the field of computer vision, where the introduction of machine learning methods has led to

the development of performant representations[5–8]. However, all these representations are obtained by applying fixed procedures to the available images. No microstructure-specific information is used. Herein, we investigate whether it is possible to further improve the representations by using a machine learning algorithm to learn how a microstructure can be best represented from the available data. As increasingly large microstructure databases are becoming available, such a data-driven approach seems promising. From the field of deep learning, we propose triplet networks as a method to learn optimal representations directly from the available microstructure data[9,10]. These representations have two desirable properties. First, a distance between two data points can be defined and can be used as a similarity measure: visually similar microstructure images will have representations that lay close to each other in the representation space. Second, the dimensionality of the representation can be freely chosen. In order to build robust machine learning models, it is recommended that the number of model inputs remains small[11]. This implies that the microstructural representation should be preferably as low-dimensional, or compact, as possible. We investigate how many dimensions a microstructural representation needs to have in order to be able to encode sufficient detail about the material. The combination of these two desirable properties makes it possible to faithfully visualize many microstructures in a single plot.

Research in automated microstructure recognition has mainly focused on distinguishing between groups of materials with significantly different compositions and processing conditions and consequently clear visual differences. When examining the literature one quickly finds several recent examples, all of which report close to perfect performance[5,7,12,13]. However, the question remains whether it is possible for a machine learning model to learn to distinguish between materials with only minor differences in composition and processing. This question is highly important

for two reasons. First, being able to distinguish between materials with a different composition and processing is necessary to successfully establish a link between processing and structure. Second, as the properties of steel alloys are very sensitive to small changes in composition and processing conditions[14], detecting these changes is essential to establish a successful link between microstructure and properties. As the resulting microstructures are often visually similar, expert metallurgists also have difficulties recognizing these differences. Machine learning models could consequently prove invaluable in making these analyses faster and more accurate. In this study, we use a dataset of 60 materials, each characterized by their chemical composition and processing procedure. None of them is exactly identical to another material in the set; some of them have rather similar compositions and processing procedures, whereas others have very different ones. According to the regular metallurgical classifications, these 60 materials can be divided in 5 classes: pearlite with austenitic matrix, martensite with prior austenite grains, tempered martensite, quenched martensite and ferritic steel. In contrast to what is usually done in the literature, we do consider these 60 materials as independent ones. This allows us to examine to which extent the triplet network representations are able to discern differences between very similar materials. If that turns out to be the case—which it will—then the triplet network sees more information in the images than conventional metallurgy does.

RESULTS AND DISCUSSION

Microstructure representations used in the literature

We briefly discuss the most commonly encountered microstructural representations in the literature. These representations are used as a comparative benchmark for our method.

A first set of methods are inspired by the work of Chowdhury et al.[5]. The methods we evaluate include Haralick features, contrast features and local binary patterns. Haralick and contrast features extract information based on the distribution of the greyscale values of neighbouring pixels. Local binary patterns also consider the immediate neighbourhood of the pixels in the greyscale image and keep track of how many times each type of neighbourhood occurs. All these features therefore aggregate local information to obtain a global description of the image, typically stored as a numerical vector.

The visual bag of words approach used by DeCost et al.[15] inspires the second set of methods. A dictionary of commonly occurring visual keypoints is constructed and for each image the number of occurrence of each of these keypoints is counted. The size of the resulting feature vector depends on the number of common keypoints that are in the dictionary.

A third type of features is obtained by applying principal component analysis on the two-point statistic[8], which is essentially the auto- and cross-correlation function of the binary image. Only the most important principal components are retained, resulting in a compact description of the microstructure. An additional benefit is that this description allows for image reconstruction, as discussed in Fullwood et al.[8]. However, this is partially because the input image is reduced to binary values; hence, a lot of information is already omitted beforehand. As the two-point statistic is used, we expect the method to be mainly useful for materials that consist out of two distinct components such as dual phase steels.

We also examine the recently introduced translation-invariant texture features[7]. Here, a dictionary of visual words is created by aggregating the intermediate output of a Convolutional Neural Network (CNN). As described in the study, we apply three different encodings of this output (Vector of Locally Aggregated Descriptors (VLAD), mean and max encoding). We consider both the

output of the last block and second to last block of a pretrained deep learning network with a VGG16 architecture[16].

Lastly, we have also included the features used in Gola et al.[6]. Here, a modified version of the Haralick features, called textural features by the authors, are used in combination with morphological features of the grains. We consider the performance of these textural and morphological features separately, but also use a genetic algorithm to select the most relevant subset of these features, as is described in Gola et al.[12].

We do not include results from image segmentation methods[13,17,18]. For these methods, the aim is not to assign a given microstructure image to the correct material class, but rather to assign each individual pixel of the image to the correct material class. Although such an approach clearly gives a more complete analysis of the image, it requires training data where each of the pixels is manually labelled by human experts. In case one considers a four class classification problem, such a labelling procedure is still doable, albeit labour intensive[13,17]. In our case, where we consider 60 different classes, manually assigning each pixel to one of these classes is nearly impossible. When microstructures become more complex, it is also harder for human experts to label the data consistently. We therefore do not include any methods that use pixel-labelled data in our benchmark.

Making microstructure representations with triplet networks

We present a deep learning model[9] that allows to construct low-dimensional representations of microstructure images. As a deep learning model can have several millions of trainable parameters, a large amount of data is required to train a network in such a way that the model is sufficiently general. As we only have a dataset with less than 1000 images, training such models from scratch is not an option. We therefore use a multi-stage approach, which gradually refines the representations starting from a pretrained model, as illustrated in Fig. 1. We modify the architecture of the pretrained convolutional network and finetune its parameters to the microstructure classification task. The resulting model then serves as the starting point of a so-called triplet network[10], which
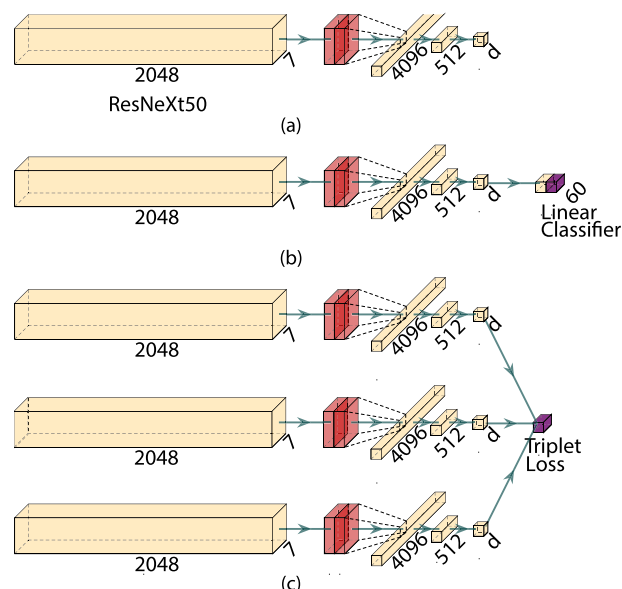

Fig. 1 Steps in the training procedure of the triplet network. a The output network of a pretrained ResNeXt50 model is replaced by bottleneck layers. **b** The output layers are fine-tuned to the microstructure classification task. **c** The representations are further improved by optimizing the triplet network.

will generate the final representations. Each step is discussed in detail in the following paragraphs.

As a starting point, we use the so-called ResNeXt50 architecture[19]. The output of the ResNeXt50 model is a 4096-dimensional vector, which clearly is too large for our purposes. To this end, we replace the final fully connected layers of the network, reducing the output dimensions to form a bottleneck, similar to the encoder network in an autoencoder[20]. An advantage of this architecture is that we can freely choose the dimensionality of the representation space, which we will denote as d.

To speed up the training of the triplet network, we start by fine-tuning the model on a classification task, where the model must learn to distinguish between the 60 different classes of materials present in the dataset. Fine-tuning a pretrained model is referred to as transfer learning[21]. In the literature, it is found that by using transfer learning, one can obtain models that are much more general and easier to train[22]. To perform the classification, an additional fully connected layer is added to the network. This layer linearly maps the representations onto a C-dimensional space, where C denotes the number of classes. After applying the softmax activation function, the probabilities of belonging to each of the classes is obtained. The negative log-likelihood cost function is used to optimize the network taking these probabilities as input. The probabilities can be interpreted as levels of confidence, providing a clear advantage over hard classification. The layers we have included for performing the classification task, can be seen as a logistic regression model. This model takes the d-dimensional representations of the images and applies logistic regression to classify these images. As logistic regression is a linear classification method, this implies that the model will only try to make the representations belonging to the different classes linearly separable. This is not necessarily a desired feature of a microstructural representation, as images of the same material can still be relatively remote from each other in the representation space.

A desired feature of a representation should be that 'similar' images, meaning images belonging to the same material, should be close to each other in the representation space. To this end, we propose to use triplet networks[10]. Triplet networks consist of a single deep learning network that maps each image $\mathbf{x}_i$ onto a representation vector $\mathbf{f}(\mathbf{x}_i)$. Rather than training with one image at the time, three images are used, of which one is used as reference, the anchor $\mathbf{x}_i^a$, one is used as an example of the same class, the positive example $\mathbf{x}_i^p$ and the final one is used as an example of a different class, the negative example $\mathbf{x}_i^n$. The criterion used during the parameter optimization is the triplet loss[23]:

$$\sum_i^N \max\left(0, ||\mathbf{f}(\mathbf{x}_i^a) - \mathbf{f}(\mathbf{x}_i^p)|| - ||\mathbf{f}(\mathbf{x}_i^a) - \mathbf{f}(\mathbf{x}_i^n)|| + a\right), \quad (1)$$

where N is the number of triplets used in the batch and a is a positive, real number that represents the margin between the positive and negative pairs. Intuitively, this loss will minimize the distance between the representations of microstructures belonging to the same class and maximize the distance between microstructures belonging to different classes. By using triplet networks, we thus introduce distance as similarity measure between images. The triplet network only needs to know which images belong to the same class and which images do not. This is an important difference with the CNN, where each material needs to have its own label as class information is explicitly encoded in the softmax layer. Because of this, we expect to more easily extend triplet models to new microstructures in the future, when applying this model on different datasets.

The datasets

Two datasets are used in this work. The first dataset contains 778 optical microscopy images of 5 visually different groups of austenitic, martensitic and ferritic steels, as is shown in Fig. 2a. Within these groups, we consider materials that have slightly different compositions and processing conditions, leading to a total of 60 different material classes. The ranges of compositions per group are listed in the Supplementary Table 1. For each material, images were taken at half and quarter thickness, at three different locations in the cross-section. All images are converted to greyscale.

In Supplementary Fig. 1, we show a typical microstructure image for each of the classes in the dataset. Some of these images appear to be almost identical, even to the trained human eye. For each class, there are 7 to 24 different pictures with a 1000×1200 pixel resolution. We express the magnification of the images in terms of the inter-pixel distance, which is the physical distance between two neighbouring pixels. For this dataset, the inter-pixel distances range from 0.1 to 5 μm. The number of images and the different magnifications for each class is shown in Supplementary Fig. 2.
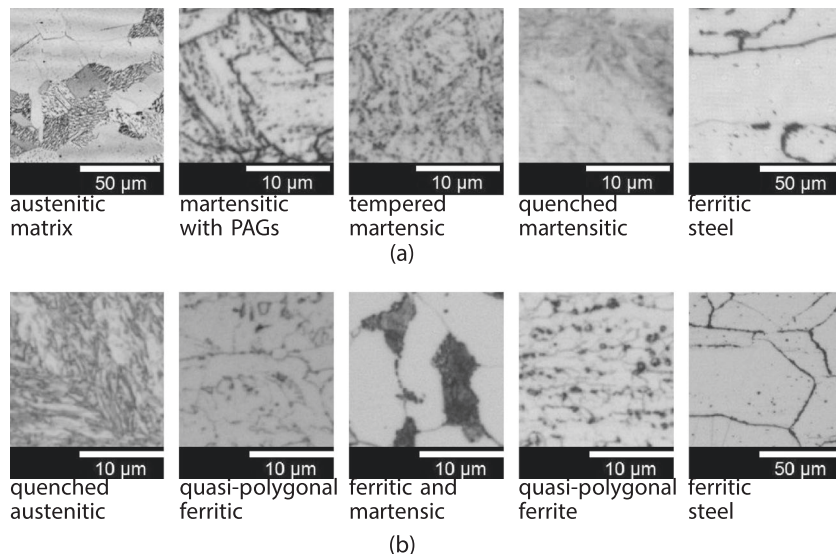

Fig. 2 Examples of the optical microscopy crops seen by the model. a For the first dataset, we show one crop for each group, which is described in more detail in Table 1. **b** For the second dataset, we show crops of the first five material classes.

M. Larmuseau et al.

The second dataset is smaller and is used to verify how well our model generalizes to unseen types of microstructure images. The variety of microstructure types is larger in this dataset and most of the microstructures commonly encountered in current industrial steels such as martensite, bainite and pearlite are included. It consists of thirty 1000×1200 pixel images belonging to ten different material classes. For each class there are three images, all with the same magnification. The inter-pixel distances range from 0.1 to 0.5 µm. Figure 2b shows a few examples of images belonging to different classes for the second dataset. An example for each of the material classes is shown in Supplementary Fig. 3 and short description of these classes can be found in the Supplementary Table 2.

Using the datasets described above, we first try to answer the question of how many dimensions we need to represent microstructure images, while still being able to distinguish between different materials. Second, we analyse the strengths and the weaknesses of the triplet model in more detail and investigate the visual features the model deems important using saliency maps. Next, we compare the discriminative power of the presented representations to other methods found in the literature. Finally, we check how general the representations are by applying the model to a microstructure recognition task with different materials. The main performance metric we use throughout this work is the accuracy, defined as

$$Accuracy = \frac{\#Crops\ correctly\ classified}{\#Total\ crops\ classified} \cdot 100\%. \quad (2)$$

We define the train accuracy as the accuracy on the dataset used to train the model and the test accuracy as the accuracy on the unseen test set, which was constructed using the procedure described in the "Methods" section. All accuracies reported in this section are obtained by training a random forest model on the representations under consideration[24].

Evaluating the accuracy for different representation sizes

One of the main advantages of the presented method is that it allows the user to choose the dimensionality of the representation. Although the ideal dimensionality might differ for other datasets, it still provides us with a valuable indication on how compactly we can represent a microstructure. In Fig. 3, we show the classification accuracy as a function of the representation dimension, for features obtained both from the CNN bottleneck layer and the triplet network for the task of recognizing the correct material class. In both cases, a random forest model[24] is used to perform the final classification. For a three-dimensional representation, we already see we can do better than all currently existing methods. The difference between the triplet network representations and the CNN representations is only significant for two and three dimensions, where the triplet network performs better. The result for a representation dimension of 512 indicates the performance in case no bottleneck layer is used. As expected, the introduction of the bottleneck layer decreases the performance. Despite the representation dimensionality being reduced by a factor of 50, the performance only drops by a few percent. For instance, in the three-dimensional case, the classification accuracy is still about 63%, whereas for the 512-dimensional case the accuracy increases to about 71%. Considering the benefits of having such a low-dimensional representation, such as the possibility of visualization, we consider this drop in performance acceptable. Based on these results, there is a strong indication that deep learning is indeed able to capture the relevant information of microstructure images in a very compact way. This finding has important consequences, as it implies that even with very little experimental data available, it should be possible to link the compact representation of the microstructure to properties and the processing conditions of the material. As data on properties is
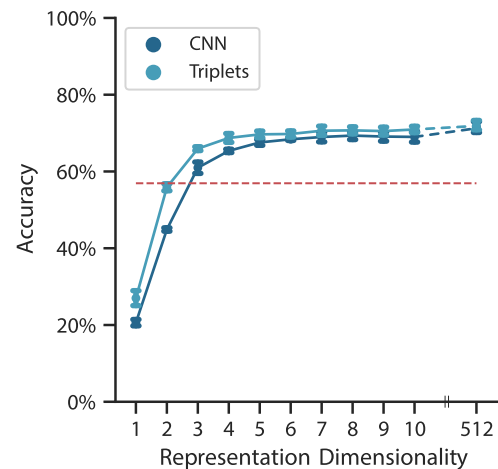

Fig. 3 The effect of the representation dimensionality on the classification accuracy. The microstructure classification accuracy of the proposed deep learning methods as a function of the representation dimension for the task of recognizing the correct material class for dataset 1. The red line indicates the best performance of the methods found in the literature. All accuracies were obtained using a random forest classifier on the microstructural representations. The error bars represent the standard deviation on three different splits of the training and test data.

scarce and expensive, reducing the number of inputs for the structure–property models will lead to a better performing model[11].

In DeCost et al.[7], the t-distributed stochastic neighbor embedding dimensionality reduction technique[25] is used to create two-dimensional maps of microstructures. An advantage of the low dimensionality of our representation space is that it is possible to directly visualize the representations in a faithful way without having to rely on dimensionality techniques. Figure 4 shows the representations in two dimensions. In the left figure, we see that microstructure images belonging to the same class, represented by dots of the same colour, tend to form elongated lines. Indeed, the representations were obtained by minimizing the error on the softmax predictions, which requires the representations to be linearly separable from each other. In the figure on the right, this is not the case, as we see more compact and isolated clusters. This is because the triplet loss requires representations belonging to the same class to be as close to each other as possible. From the figures, it is noticeable that representations with similar colours tend to lay close together. Materials from the same groups have the same colourmap, which is defined in Table 1. Within each group, the intensity of the colour varies. Attention was paid to assigning similar values of intensity to microstructures that are similar in terms of processing. Especially for the triplet network, we see that the representations with lighter and darker intensities are remote from each other. This implies that the network has indeed learned a meaningful distance metric, which reflects the underlying processing of the microstructure. It is especially striking how well the different groups are separated by each other. Without any information on which materials belong to the same group, the deep learning model naturally clusters materials belonging to the same group together, indicating that the learned similarity measure corresponds well to human perception of visual similarity. It also justifies our approach of defining 60 different material classes, as within each group the representations show clear substructures that contain much more information than would have been the case if the triplet networks was trained using only the information of the material groups.
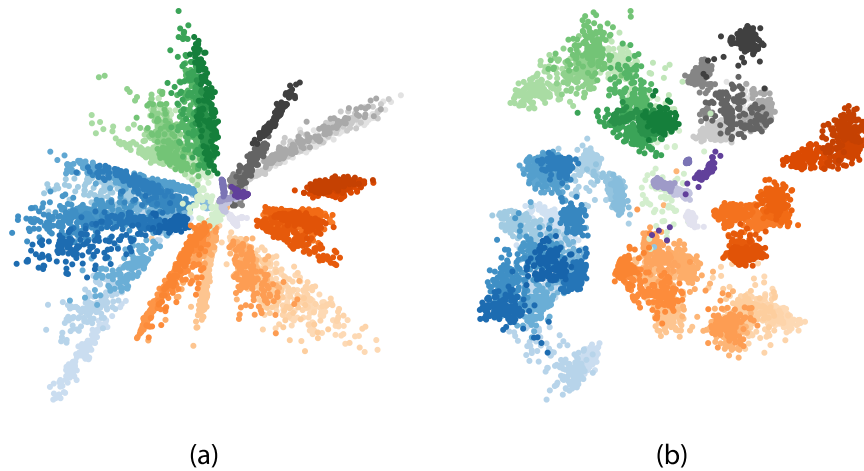
(a)   (b)

Fig. 4 The representations in two dimensions. The two-dimensional representations obtained with (**a**) the bottleneck CNN and (**b**) the triplet network. The representations of the images belonging to the same class have the same colour. The colourmaps defined in Table 1 are used to indicate representations of the same group. Both models naturally learn to separate the different groups from each other. Within the group, the triplet model is also able to correctly cluster together the image representations belonging to the same material.

Table 1. Description of the different groups of materials in dataset 1.

Group id	Description	Etching	Classes	Colourmap	Accuracy
1	Austenitic matrix	Nital	1–6	Grey	81.1 ± 2.7
2	Martensitic with PAGs	Bechet–Beaujard	7–27	Orange	60.7 ± 1.6
3	Tempered martensitic	Nital	28–46	Blue	60.5 ± 1.1
4	Quenched martensitic	Nital	47–55	Green	61.6 ± 5.9
5	Ferritic steels	Nital	56–60	Purple	97.6 ± 2.3

Dataset 1 only contains martensitic, austenitic and ferritic microstructures. To each group, we have assigned a specific colourmap, which is used in the plots in the following sections. We also mention the etching procedures that were used to obtain the images, as it can greatly impact the visual appearance of the microstructure. We also list the corresponding labels of the material classes that belong to these groups. The last column lists the classification accuracies within the groups of materials for the three-dimensional triplet representation using a random forest classifier.

Comparing the network performance to human experts

As deep learning models extract information from the pixel level, they have the potential to extract much more complex patterns than would be possible for humans. From the results in the previous section, we already know that it is possible to correctly recognize more than 60% of the crops. This suggests that the model is capable of distinguishing between materials with only small differences in processing conditions, which would be indistinguishable for the human eye, and triggers further investigation. On the other hand, 60% is well below typical values reported in literature, which might be an indication that some material classes cannot be discerned from each other.

Figure 5 shows the confusion matrix for microstructure classification on the test set, using random forests applied to the three-dimensional triplet representations. All results are averaged over three independent splits of training and test set. We have indicated the groups of the materials, as defined in Table 1, with coloured squares. As was already noted in Fig. 4, the model nearly perfectly assigns each image to the right group. We find that 98.9% of the crops are correctly assigned. This is a high score and likely on par with what a human expert could achieve. Even for experts obtaining 100% accuracy would be difficult, as it is for instance hard to tell the difference between quenched (group 3) and tempered martensite (group 4) based on a single 200×200 crop.

The accuracy for the different groups is listed in Table 1. We see that within each group the accuracy is still above 50%. This is

remarkable, as the materials within groups are visually very similar. A small sample of experts saw no clear difference between the materials belonging to the same group based on a single 200×200 image. There are big discrepancies in the accuracies of the different groups. For groups 1 and 5, the accuracy is significantly higher than for the other groups. We identify two reasons why this might be the case. First, these group contain fewer materials than the other groups, making it a priori easier to correctly recognize the right material. Second, the compositional ranges of these two groups, which are given in the Supplementary Table 1, are much wider than for the other groups. Groups 2 and 3 have a comparable number of classes and similar compositional ranges. Their performance is similar, but the Nital etching used in group 3 seems to reveal more relevant features for the machine learning algorithm than the Bechet–Beaujard etching used for group 2. For group 4, we note that the variance among the different runs is substantially higher than that of the other groups. We find that the performance of this group is most sensitive to the images that are used as a test set. In some cases, the test set mainly consists of images that are taken closer to the edge of the material, whereas there are only few such examples in the training set. The location where the images are taken seems to greatly affect the model performance and explains why, despite the group containing only eight materials, we obtain a relatively low accuracy for this group. More metrics such as precision and recall for both recognition tasks can be found in the Supplementary Tables 10 and 11. Lastly, it is clear from Fig. 5 that there are some material classes between which the model often confuses, especially for groups two, three
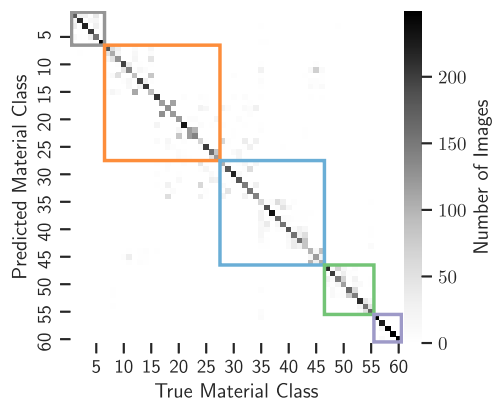
Fig. 5 The confusion matrix of the three-dimensional triplet model for the first dataset. The greyscale value indicates the number of images that belong to the material class shown on the x-axis and are assigned by the model to the material class shown on the y-axis. For the ideal model, only the diagonal would be coloured. The coloured boxes show the groups using the colourmaps that are defined in Table 1. The model mainly confuses materials belonging to the same group, which are visually very similar.
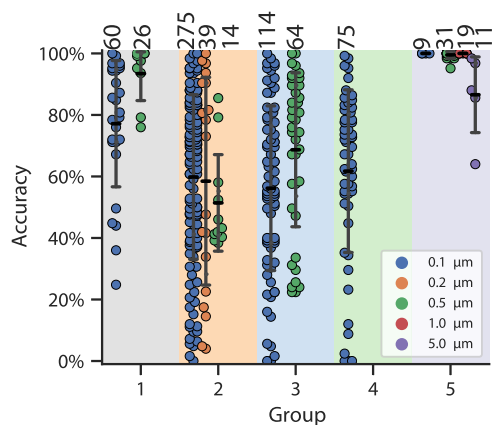

Fig. 6 The performance of the three-dimensional triplet network by group and magnification. The model is able to deal with different magnifications, measured by the inter-pixel distance, without being explicitly taught to do so. The number on top of each column indicates the number of images in the dataset for the given group and magnification. The background colouring uses the colourmaps defined in Table 1. The thick black lines represent the mean accuracy and the error bars represent the Standard Deviation (SD) on three different splits of the training and test data. The dots depict the average accuracy of all crops belonging to the same image.

and four. It is an indication that in these groups some of the material classes are indistinguishable. In Supplementary Fig. 5, we study the effect of the number of classes more explicitly. We show that it is possible to obtain more than 90% accuracy, while retaining more than 30 different material classes.

Analysing the effect of different magnifications

It is interesting to analyse how our method deals with the different amount of training data for the different magnifications. As can be seen in the Supplementary Fig. 2, the dataset contains images at different magnifications for many of the materials, but the number of images for each magnification differs strongly. From Fig. 6, we find that the model achieves a better score on the lower magnifications (0.5 µm inter-pixel distance) for the first group, even though there are more than twice as many training images for the lower magnifications (0.1 µm inter-pixel distance).

For group 2, we see an opposite trend, as the accuracy decreases with decreasing magnification. Especially for the 0.5 µm magnification, the performance drops significantly. This is because sufficient detail is no longer present at this magnification to correctly classify the images, as is confirmed by human experts. For group 3, we observe a similar trend as for group 1 with lower magnifications being preferred, even tough fewer images are present in the dataset. For group 5, all magnifications have very high scores, except for the 5.0 µm magnification, which no longer contains sufficient detail about the material. There are two main observations from this all. First, the model is able to cope very well with different magnifications. A possible explanation is that the grouped convolutions in the ResNeXt architecture allow the model to look at the image at different length scales at the same time. A second observation is that the model has a clear preference for certain magnifications, and that this preference strongly depends on the material group. This can be understood by considering the trade-off between statistical representativity, which requires the image to cover a sufficiently large surface of the material, and sufficient detail, which requires the image to have a sufficiently high resolution. As the model examines the images at the pixel level, it is able to detect small details in the image very well, but it requires to examine a larger surface of the material to correctly recognize the microstructure. This can also be seen in the examples in Fig. 7. The yellow regions for groups one, two and five are areas where the model doubts between several materials, because it cannot detect enough relevant features to correctly recognize the material based on a single crop. At lower magnifications, crops will contain more relevant information. For groups three and four, the opposite is the case and the model recognizes clear features from two different classes in the yellow regions. This implies that for these two groups the 200×200 crops already contain a lot of information. We conclude that the model will prefer lower magnifications provided there is still enough detail in the image and that the preferred magnification is strongly dependent on the material under consideration.

We have also included the average accuracy per image in Fig. 6. We find that analysing such averages is helpful in detecting where the model systematically fails to correctly recognize the correct material class. In the Supplementary Figs. 9–13, we show some example images that are completely misclassified. It turns out that although the model is relatively robust to irregularities such as dust particles on the microscopy, regions that are slightly out of focus and the presence of over-etched regions, it is sensitive to the precise etching procedure and this can overlay the subtle differences between some of the material classes.

Interpreting the model predictions

To shed more light on where the model looks at, we propose a simple method based on averaging predictions of many crops belonging to the same image. Per pixel, we average the predictions of all crops that contain that specific pixel. Thus, we obtain a probability distribution per pixel over the different material classes. In Fig. 7, we show the result of this procedure for a typical image of each group, which was taken from the test set. Based on these heatmaps, we obtain a clear idea of where the model looks at to recognize the materials. However, the reason why specific regions are more easily recognized is not always immediately clear. We briefly discuss an example for each group and give a possible explanation. For the first group, we see that the model mainly looks at the density of precipitates and of so-called pearlitic lamellae to assign the image to the right material. Also, the presence of triple junctions such as the one in the upper centre of the image seems to affect the model predictions. For the second group, we see that mainly the grain size affects the model decision. The regions with larger grains are assigned to the right material, whereas the regions with smaller grains are assigned to
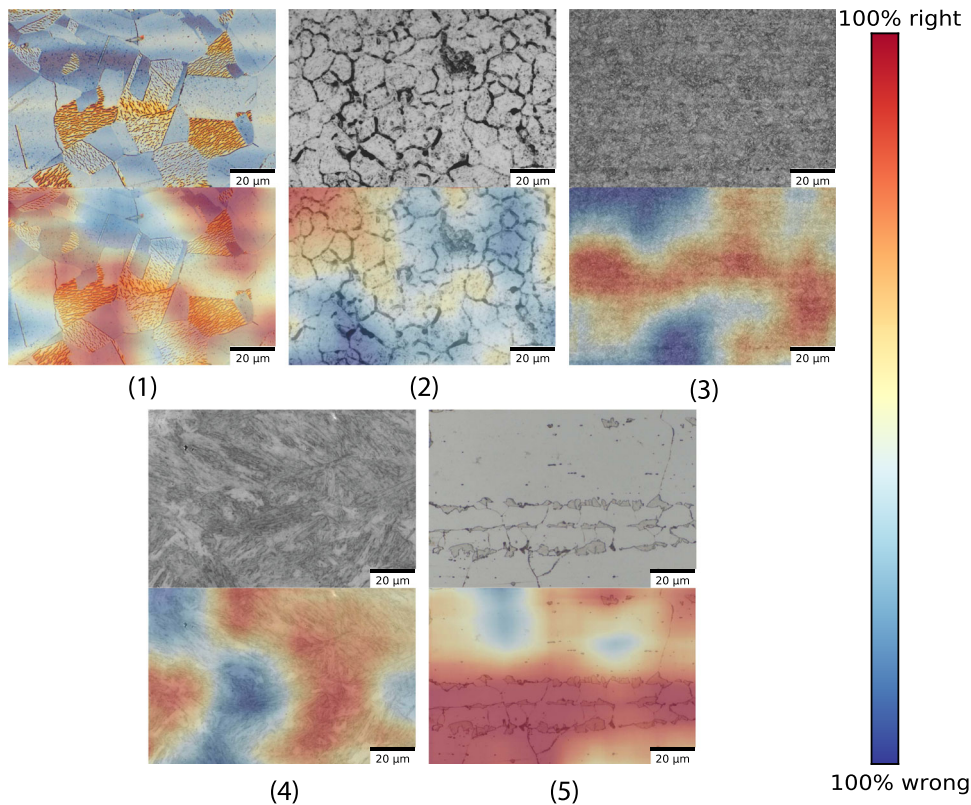
Fig. 7 Heatmaps of the three-dimensional triplet network predictions. By averaging the predictions of 10,000 crops taken randomly from the image, we can construct heatmaps of regions where the model looks at for assigning an image to a material class. For each of the material groups, one of the original images of the test set is shown on top and the overlaid heatmap is shown below. The colourmap is chosen such that a region on the image is marked red if the model assigns this region to the right material and blue if the region is assigned to a different material. Yellow indicates that model is unsure of which class the material should be assigned to.

other materials. Regions with medium sized grains are marked yellow, as the model doubts between the right material and other materials. In the third group, we show an image where the model fails to correctly classify the material. The presence of elongated prior austenitic grains causes the model to assign this image to a different material class. In the regions where these grains are rounder, the model doubts between the right material class and the other. A closer inspection of the training data, shows that the prior austenitic grains are typically indeed rounder for the correct material class. Hence, it is understandable why the model is confused. In the fourth group, we see some clear regions that are marked in blue. The blue region in the centre of the image has darker, elongated grains, whereas the blue region in the upper left corner has rounder grains. Although these two regions are marked blue, they are very different and further analysis shows that these two regions are indeed assigned to different material classes by the model. In the image of the last group, we see that areas with small, elongated grains and second phases (the small polygonal areas) are marked blue. This combination of small, elongated grains and second phases is not the most representative of this class and similar features are seen in the classes the model confuses with . Our analysis indicates that the model has learnt that physically relevant features such as the size and shape of grains are important to distinguish between different materials. A larger version of the images can be found in the Supplementary Figs. 14–18.

Comparing the triplet representations to other representations

We benchmark the discriminative power of the presented method to other microstructural representations found in the literature by evaluating the accuracy on two different microstructure recognition tasks. The first task is to correctly recognize the group, as defined in Table 1, to which the microstructure on the image crop belongs. The second task is to correctly recognize the specific material. More information on the implementation of the other microstructural representations can be found in the Supplementary Methods.

In Table 2, we show the test accuracies for both microstructure recognition tasks. We see that the task of recognizing the right group results in very high accuracies, in accordance with what is reported in the literature. As expected, the recognition of the right material turns out to be much harder. It is noteworthy that the older and more conventional microstructural representations such as the correlations, keypoints and morphological features do not have a very high discriminative power. These features were hand-crafted by metallurgists to distinguish different metallurgical phases, similar to the groups we are considering and were not designed to spot small differences in microstructures. Texture-based methods such as the Haralick features and the texture features introduced in Webel et al.[26] achieve very good accuracies despite their low dimensionality. Still, we find that mainly deep learning-based features obtain the highest accuracies. Of all methods found in the literature, we find that the VGG16 C_{43} features with mean pooling obtain the best performance, which is in line with the findings in DeCost et al.[7]. However, this representation has a dimensionality of 512, which is much higher than what we aim for in this paper. We see that the triplet networks presented in this paper outperform the other methods found in the literature. This is not unexpected, as the network used to create these features was specifically trained on the dataset, whereas other methods use the network output of a standard pretrained model without further optimization of the

Table 2. Comparison of the classification accuracy of several microstructural representations on dataset 1.

Method	Dimensionality	Test acc. group [%]	Test acc. material [%]
Haralick[5]	13	94.4 ± 0.2	46.5 ± 2.3
lbp[5]	20	94.3 ± 0.3	34.4 ± 1.1
Greyco[5]	4	72.1 ± 0.5	18.5 ± 0.5
Correlations[8]	20	78.0 ± 1.0	19.7 ± 1.9
Surf[15]	100	69.1 ± 0.9	13.6 ± 0.8
VGG16 VLAD C_{43}[7]	16,384	62.7 ± 6.5	48.3 ± 6.6
VGG16 mean C_{43}[7]	512	99.2 ± 0.3	56.2 ± 1.5
VGG16 max C_{43}[7]	512	97.9 ± 0.3	47.6 ± 2.3
Morphological[6]	21	77.1 ± 2.4	17.5 ± 1.0
Texture[26]	8	90.6 ± 1.5	44.4 ± 3.8
Morph. + texture[6]	29	93.4 ± 1.0	45.8 ± 3.8
Morph. + texture + genetic algorithm[12]	16	93.6 ± 0.0	47.8 ± 0.0
Triplets	2	98.8 ± 1.8	55.6 ± 1.1
Triplets	3	99.1 ± 1.8	65.9 ± 0.7
Triplets	10	99.2 ± 1.5	71.0 ± 1.2

We show the accuracy (acc.), both for the task of correctly recognizing the right group (see Table 1) and for recognizing the right material. All accuracies were obtained by using a random forest classifier on crops coming from images of the test set. We also list the SDs, which are computed by repeating the entire training procedure three times with different splits for the train and test data. The proposed models clearly perform better than other models found in the literature, while using much more compact representations.

Table 3. Comparison of the classification accuracy of several microstructural representations on dataset 2.

Method	Dimensionality	Out-of-bag acc. [%]	Test acc. [%]
Haralick[5]	13	98.4 ± 0.3	91.5 ± 3.2
lbp[5]	20	92.0 ± 1.0	83.6 ± 4.2
Greyco[5]	4	63.0 ± 1.0	50.9 ± 1.0
Correlations[8]	20	79.8 ± 1.0	69.8 ± 2.6
Surf[15]	100	28.1 ± 1.4	28.4 ± 1.2
VGG16 VLAD C_{43}[7]	16,384	65.5 ± 3.8	77.3 ± 0.8
VGG16 mean C_{43}[7]	512	99.4 ± 0.1	97.8 ± 0.5
VGG16 max C_{43}[7]	512	98.6 ± 0.2	96.1 ± 2.2
Morphological[6]	21	72.0 ± 1.6	66.7 ± 0.7
Texture[26]	8	89.6 ± 1.9	83.4 ± 2.2
Morph. + texture[6]	29	93.3 ± 1.1	88.9 ± 1.8
Morph. + texture + ga[12]	16	92.7 ± 1.9	87.5 ± 1.8
Triplets	2	71.7 ± 4.8	66.2 ± 6.5
Triplets	3	83.7 ± 3.9	78.9 ± 5.3
Triplets	10	97.7 ± 0.6	94.6 ± 2.0
Triplets	512	99.9 ± 0.1	99.5 ± 0.4

Both the accuracy (acc.) for the test data and the out-of-bag accuracy are listed. All accuracies were obtained using a random forest classifier. We also list the SDs, which are computed by using threefold cross-validation. Even on unseen materials the proposed methods perform as well as the other representations with similar dimensionality.

model parameters. Still, our method outperforms other methods with such small feature vectors. Already with a two-dimensional representation, we can get a performance comparable to the best methods found in the literature. These results suggest that the currently used methods yield vector representations that are much larger than they should be.

Supplementary Table 12 gives more results, where some other less performing methods are included. We also list the out-of-bag accuracy of the random forest on the training set, which should be a good indication of the performance on the test set[24]. The out-of-bag accuracy is however significantly higher than the accuracy on the test set for all features. This is due to the fact that the images used for training the decision trees and the ones left out for determining the out-of-bag score overlap, as they are crops from the same original images. This stresses the need of examining new independent images to have an unbiased evaluation of the model performance. In our comparison, we systematically use a crop size of 200×200, because most modern deep learning architectures are trained on crops of similar size. Therefore, the crop size might favour deep learning method compared to ways of representing the microstructure. In the Supplementary Fig. 7, we study the effect for the crop size in more detail for the Haralick features. We find that using larger crop sizes can increase the performance by another 3%.

Assessing the generalization to new materials

As the dataset, which we used to train our models, contains five different groups of materials, it is interesting to check how well the representations generalize to unseen types of materials. In the literature, this is often referred to as zero-data learning or zero-shot learning[27,28]. To this end, we use the triplet network that was trained on dataset 1 to obtain representations for the images of dataset 2. These representations are then again used to train a

random forest classifier to recognize the ten predefined material classes of dataset 2. Thus, we can assess how well the representations that are computed by the triplet network generalize to new groups of materials. As before, we make a comparison to other methods found in the literature.

To measure the performance of each of the representations, we show in Table 3 the out-of-bag accuracy on the training set and the test accuracy. We use threefold cross-validation and for each fold we train the model on crops coming from two images of each class, and we use the crops of the third image as a test set. Thus, we never use crops from the same image simultaneously in the training and test set.

Many representations obtain accuracies close to 90%. This is expected, as there are clear visual differences between most of the material classes, which was not the case for the materials in dataset 1. There is a clear tendency for higher dimensional representations to perform better. The only exceptions are the VLAD-encoded representation, of which the dimensionality is too high to learn meaningful patterns for the small dataset under consideration. The triplet network representations perform at least as good as other representations of similar dimensionality, which indicates that they generalize well to new materials. However, the difference in performance between the three- and ten-dimensional representations is much bigger than for dataset 1. This seems plausible, as trying to capture all relevant information of a dataset in only two or three dimensions requires the model to learn features that are more specific to the materials in the training set compared to the ten-dimensional case.

In Fig. 8, we show the confusion matrix of the representations from the three-dimensional triplet network. The model mainly confuses between crops from material classes 4, 6 and 8. These classes are indeed visually similar, as they all have features from both bainitic and quasi-polygonal ferritic steel. Furthermore, bainite was not included in the training set, so that we would
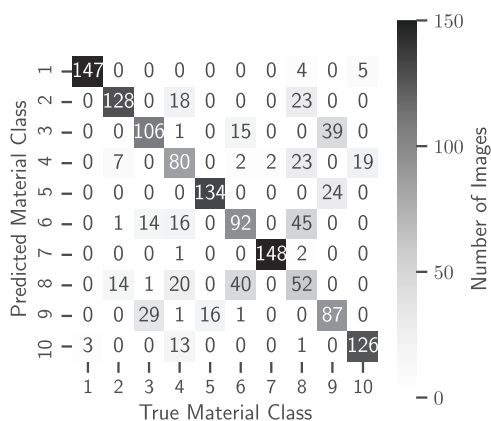
Fig. 8 The confusion matrix of the three-dimensional triplet model for the second dataset. The model obtains a decent performance for all materials, but struggles most with materials that were not included in the original training set.

indeed expect a lower performance compared to the other material classes. The model deals relatively well with all materials and, as before, we find that the triplet representations are able to quantitatively express visual similarity.

It is clear that there is an unavoidable trade-off between the dimensionality of the microstructural representation and its generalizability. Low-dimensional representations tend to be either unable to capture enough detail or are too much tailored to the materials in the training data. As was the case for recognizing the correct group in dataset 1, we obtain near-perfect accuracies on this dataset. This is once more an indication that microstructure recognition of microstructures with clear visual differences is no longer a challenging topic. Research should instead focus on datasets with visually similar materials and possibly try to go beyond the capabilities of human experts.

We propose triplet networks as a method capable of learning highly compact representations of microstructure images. We demonstrate that by using information on the composition and processing, it is possible to obtain very detailed microstructural representations. Despite being low-dimensional, these representations contain sufficient information to discern visually similar materials that only have small differences in composition and processing conditions. Capturing these small differences is a prerequisite for structure–property prediction, as small variations in composition and processing can greatly affect the properties. Furthermore, the method is able to cope well with a large range of magnifications. We introduce a visual way of interpreting the representations and find that they behave surprisingly similar to how expert metallurgists analyse microstructure images, focusing on features such as grain edges and precipitate densities. We present a comparative benchmark of different microstructural representations found in the literature for a microstructure recognition task. Already in two dimensions, the triplet network representations obtain a performance that is competitive with the state-of-the-art. When applying the representations to new materials, the generalization performs at least as good as other representations with the same dimensionality. We observe a clear trade-off between compactness and generalization, as high-dimensional representations perform significantly better than the more low-dimensional ones. As triplet networks learn how to represent a microstructure directly from the data, the representations should become even better when trained on a larger dataset of microstructure images. As the size of microstructure image datasets is rapidly increasing, we expect the presented method has a high potential as a tool for microstructure analysis.

METHODS

Obtaining the training and test set

In order to correctly evaluate the model performance, it is important to split the dataset in a training and test set, so that both are representative and independent. The training set is used to train the model, while the test set is only used to evaluate the model performance. In the first dataset, we use for each class 80% of the images as the training set. Each steel is considered to be of a different class when the processing conditions and composition of the steel differ, regardless of whether the properties of the steels differ or whether it is possible to visually distinguish between the materials. Although this is an objective criterion to define different materials classes, there is no a priori guarantee that it is possible to recognize the right material purely based on an optical microscopy image. Still, by introducing more material classes than is usually done, we aim to let the triplet network learn very detailed representations of the microstructure. Once we have these detailed representations, we apply them to recognize both the correct material class and the correct material group. As the dataset is highly unbalanced in terms of microstructure images per class, we randomly select crops of the original images with a 200×200 pixel resolution until we have 500—possibly overlapping—images in each class. A test set is created by taking, for each class, the remaining 20% of the original images with a 1000×1200 pixel resolution that are not included in the training set and by randomly selecting a total of 125 crops with 200×200 resolution from these images. The training and test set will hence never contain crops from the same image. We repeat the outlined procedure to create crops for the training and test set three times. All reported results are the average performance over these three different train and test splits, and where relevant we also mention the SD over these three splits.

For the second dataset, 50 randomly sampled 200×200 crops were taken from each image. Threefold cross-validation is used to obtain a reliable assessment of the model performance.

Training the triplet network

For our starting model, we adopt a ResNeXt50 convolutional architecture[19], which has over 25 million trainable parameters and was pretrained on the ImageNet dataset[29]. We choose this architecture, because it performed better than other architectures with similar complexity in our experiments, which is discussed in more detail in the Supplementary Table 3. We use pretrained weights only for the convolution layers of the model. The fully connected layers at the end of the network are trained from scratch and we are therefore free to choose the architecture of these layers. All models are implemented using the PyTorch deep learning framework[30] and fine-tuning of the pretrained model is performed using FastAI[31].

Data augmentation is used to artificially increase the number of images in the training set. We randomly apply Gaussian blur, rotations and changes in lightening to the crops to make the model invariant to changes in these conditions. Details on the exact augmentations used in this work can be found in Supplementary Table 6.

One of the challenges of training a triplet network is the selection of the triplets. As it is computationally unfeasible to select all possible triplets, a selection has to be made. Although various criteria have been used in the literature[32–34], we use the batch semi-hard criterion, as it prevents the representations from collapsing onto the same point[23], which was a problem for the other criteria. For the training of the triplet network, we use the bottleneck models discussed in the previous section as starting point and retrained the dense layers at the end of the network, keeping the other parameters fixed. This helps to reduce the number of trainable parameters and thus the memory usage. The details of the hyperparameters used for training the CNN model and the triplet network can be found in the Supplementary Tables 4 and 5.

Choosing the right classification technique

All accuracies reported in this work are obtained by training a random forest model[24]. There are two main reasons to prefer this classifier over other techniques. First, we found that it does not require an additional validation set to tune the hyperparameters of the model if one starts from reasonable default values, thus eliminating the need for additional data. This finding is supported in the literature[35]. Furthermore, the generalization performance of a random forest can also be assessed by looking at the out-of-bag accuracy on the training set[24]. The out-of-bag accuracy relies on the fact that each individual decision tree of the random forest is trained on a subset of the training data. It is obtained by evaluating the

model performance of each sample on the trees that did not use it for training. Second, we found that it yields good results both in low- and high-dimensional feature spaces, allowing for a more objective comparison of features of different dimensionality. Support Vector Machines[36] are another commonly used method for microstructure recognition[6,12,15]. However, we found that such models tend to perform worse in high-dimensional spaces and are very sensitive to the tuning of the hyperparameters[37]. The details of the used model hyperparameters can be found in the Supplementary Table 7.

DATA AVAILABILITY

The data that support the findings of this study are available in www.microstructuredb.com/papers. The weights of some of the models are available on www.microstructuredb.com/papers.

CODE AVAILABILITY

The code that supports the findings of this study is available on www.microstructuredb.com/papers.

REFERENCES

1. Zaefferer, S., Ohlert, J. & Bleck, W. A study of microstructure, transformation mechanisms and correlation between microstructure and mechanical properties of a low alloyed TRIP steel. *Acta Mater.* **52**, 2765–2778 (2004).
2. Olson, G. B. Computational design of hierarchically structured materials. *Science* **277**, 1237–1242 (1997).
3. Panchal, J. H., Kalidindi, S. R. & McDowell, D. L. Key computational modeling issues in Integrated Computational Materials Engineering. *Comput. Aided Des.* **45**, 4–25 (2013).
4. Torquato, S. Statistical description of microstructures. *Annu. Rev. Mater. Res.* **32**, 77–111 (2002).
5. Chowdhury, A., Kautz, E., Yener, B. & Lewis, D. Image driven machine learning methods for microstructure recognition. *Comput. Mater. Sci.* **123**, 176–187 (2016).
6. Gola, J. et al. Advanced microstructure classification by data mining methods. *Comput. Mater. Sci.* **148**, 324–335 (2018).
7. DeCost, B. L., Francis, T. & Holm, E. A. Exploring the microstructure manifold: Image texture representations applied to ultrahigh carbon steel microstructures. *Acta Mater.* **133**, 30–40 (2017).
8. Fullwood, D. T., Niezgoda, S. R. & Kalidindi, S. R. Microstructure reconstructions from 2-point statistics using phase-recovery algorithms. *Acta Mater.* **56**, 942–948 (2008).
9. Schmidhuber, J. Deep learning in neural networks: an overview. *Neural Networks* **61**, 85–117 (2015).
10. Hoffer, E. & Ailon, N. In *International Workshop on Similarity-Based Pattern Recognition* 84–92 (Springer, Cham, 2015).
11. Friedman, J., Hastie, T. & Tibshirani, R. *The Elements of Statistical Learning* (Springer, New York, 2001).
12. Gola, J. et al. Objective microstructure classification by support vector machine (SVM) using a combination of morphological parameters and textural features for low carbon steels. *Computat. Mater. Sci.* **160**, 186–196 (2019).
13. Azimi, S. M., Britz, D., Engstler, M., Fritz, M. & Mücklich, F. Advanced steel microstructural classification by deep learning methods. *Sci. Rep.* **8**, 1–14 (2018).
14. Olson, G. B. & Azrin, M. Transformation behavior of TRIP steels. *Metall. Trans. A* **9**, 713–721 (1978).
15. DeCost, B. L. & Holm, E. A. A computer vision approach for automated analysis and classification of microstructural image data. *Comput. Mater. Sci.* **110**, 126–133 (2015).
16. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. Preprint at https://arxiv.org/abs/1409.1556 (2014).
17. DeCost, B. L., Lei, B., Francis, T. & Holm, E. A. High throughput quantitative metallography for complex microstructures using deep learning: a case study in ultrahigh carbon steel. *Microsc. Microanal.* **25**, 21–29 (2019).
18. Ajioka, F., Wang, Z.-L., Ogawa, T. & Adachi, Y. Development of high accuracy segmentation model for mmicrostructure of steel by deep learning. *ISIJ Int.* **60**, 954–959 (2020).
19. Xie, S., Girshick, R., Dollar, P., Tu, Z. & He, K. Aggregated residual transformations for deep neural networks. In *IEEE Conf. Computer Vision and Pattern Recognition* 5987-5995 (IEEE, New York, 2017).
20. Baldi, P. In *ICML Workshop on Unsupervised and Transfer Learning* 37–49 (Microtome, Brooklyn, 2012).
21. Pratt, L. Y. In *Advances in Neural Information Processing Systems* 204–211 (Morgan Kaufmann, Burlington, 1993).
22. Yosinski, J., Clune, J., Bengio, Y. & Lipson, H. In *Advances in Neural Information Processing Systems* 3320–3328 (Curran, Brooklyn, 2014).
23. Schroff, F., Kalenichenko, D. & Philbin, J. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conf. Computer Vision and Pattern Recognition* 815–823 (IEEE, New York, 2015).
24. Breiman, L. Random forests. *Machine Learn.* **45**, 5–32 (2001).
25. Maaten, L. V. D. & Hinton, G. Visualizing data using t-SNE. *J. Machine Learn. Res.* **9**, 2579–2605 (2008).
26. Webel, J., Gola, J., Britz, D. & Mücklich, F. A new analysis approach based on Haralick texture features for the characterization of microstructure on the example of low-alloy steels. *Mater. Charact.* **144**, 584–596 (2018).
27. Larochelle, H., Erhan, D. & Bengio, Y. Zero-data learning of new tasks. In *AAAI Conf. Artificial Intelligence* 646–651 (AAAI, Palo Alto, 2008).
28. Rohrbach, M., Stark, M. & Schiele, B. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In *IEEE Conf. Computer Vision and Pattern Recognition* 1641–1648 (IEEE, New York, 2011).
29. Smith, J. et al. Linking process, structure, property, and performance for metal-based Additive Manufacturing: computational approaches with experimental support. *Comput. Mech.* **57**, 583–610 (2016).
30. Paszke, A. et al. In *Advances in Neural Information Processing Systems* 8026–8037 (Curran, Brooklyn, 2019).
31. Howard, J. & Gugger, S. Fastai: a layered API for deep learning. *Information* **11**, 108 (2020).
32. Wu, C.-Y., Manmatha, R., Smola, A. J. & Krahenbuhl, P. Sampling matters in deep embedding learning. In *The IEEE Int. Conf. Computer Vision* 2859–2867 (IEEE, New York, 2017).
33. Oh Song, H., Xiang, Y., Jegelka, S. & Savarese, S. Deep metric learning via lifted structured feature embedding. In *The IEEE Conf. Computer Vision and Pattern Recognition* 4004–4012 (IEEE, New York, 2016).
34. Hermans, A., Beyer, L. & Leibe, B. In defense of the triplet loss for person re-identification. Preprint at https://arxiv.org/abs/1703.07737 (2017).
35. Probst, P., Wright, M. N. & Boulesteix, A.-L. Hyperparameters and tuning strategies for random forest. *WIREs Data Mining Knowledge Discov.* **9**, e1301 (2019).
36. Cortes, C. & Vapnik, V. Support-vector networks. *Machine Learn.* **20**, 273–297 (1995).
37. Duarte, E. & Wainer, J. Empirical comparison of cross-validation and internal metrics for tuning SVM hyperparameters. *Pattern Recognit. Lett.* **88**, 6–11 (2017).

AUTHOR CONTRIBUTIONS

M.L. contributed to the methodology, software, formal analysis and writing of the original draft. M.S. contributed to the methodology and writing of the original draft. K.T and L.D. provided the datasets and contributed to the writing and to the formal analysis. T.D. was responsible for supervision and funding acquisition. S.C. was responsible for supervision, writing and funding acquisition. All authors reviewed the final manuscript.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information is available for this paper at https://doi.org/10.1038/s41524-020-00423-2.

Correspondence and requests for materials should be addressed to M.L.

Reprints and permission information is available at http://www.nature.com/reprints