Hide and Seek: Using Masked Vision Transformer to Detect Surface Structures on Laser Polished Metals

Julius Neuß* and Sven Linden*

Fraunhofer Institute for Laser Technology ILT, Germany
**Corresponding author's e-mail: julius.neuss@ilt.fraunhofer.de, sven.linden@ilt.fraunhofer.de*

Polishing metallic materials with laser radiation (LP) is based on the melting of a thin surface layer. In the molten phase, surface roughness is smoothed because of the interfacial tension and the material solidifies with a smoothed surface. It also creates surface structures, which introduce roughness in a specific wavelength range. In most cases, multiple surface structures occur simultaneously. Altogether, they present a significant obstacle in achieving polishing results necessary for use in industry (Metric Ra often smaller 100 nm). However, once successfully detected, the role of surface structures towards roughness may be systematically reduced [1,2]. During process parameter development, surface structures must be analyzed manually by highly skilled process engineers using various, expensive analysis techniques to optimize process parameters (E.g., white-light interferometry and scanning electron microscope). To automate this process, this work evaluates state-of-the-art image processing methods to detect surface structures solely based on white-light interferometry. Therefore, process experts have identified up to eight different surface structures in over 2,500 white-light interferometry images to build a meaningful and representative dataset for surface structure classification on various materials. We benchmarked state-of-the-art machine learning models and show that both, ResNet [3] and vision transformer (ViT) [4] models are suitable techniques for classifying surface structures, achieving up to 82% accuracy on our dataset. Furthermore, we explore the self-supervised learning approach data2vec [5] to make unlabeled data usable by pretraining in a self-supervised fashion and show that features learnt are already descriptive enough to distinguish surface characteristics of different types without requiring any annotation. Building on state-of-art techniques, we propose a novel masking strategy to further improve the quality of the learnt features regarding surface properties, which may be beneficial in a much broader context than just metallic surfaces. With this new technique, we build a bridge for vast, unlabeled data which is often collected in large quantities from industrial machines but cannot be used in supervised machine learning without prior manual labelling.

DOI: 10.2961/jlmn.2025.01.2009

Keywords: laser polishing, vision transformer, self-supervised learning, machine learning, surface structures, data2vec, ResNet

1. Introduction

Laser polishing (LP) is a finishing process using a galvanometric scanner to guide laser radiation melting the surface, making it highly suitable for polishing complex freeform surfaces. Its highly non-linear thermo-physical relationship between process and material makes process experts spend lots of time to find process parameters which meet inquired surface quality. This is in direct contrast to the rising industry demand towards swift time-to-market and low development cost for the goal of product individualization of small lot sizes down to one.

Two process variants for polishing of metals can be distinguished: laser macro polishing with cw laser radiation [6] and laser micro polishing with pulsed laser radiation [7]. Both process variants use, in most cases, laser radiation of wavelength between 1,030 nm and 1,064 nm. While for macro polishing, a continuous melt pool with depth between 30 µm to 100 µm is created, the mechanism of micro polishing works by creating discrete melt pools, with melt pool resolidifying before the next pulse.

The goal of laser polishing is to reduce surface roughness as much as possible. In an ideal world, the process would smoothen all initial roughness leaving a perfectly flat surface. However, the combination of suboptimal process parameters, laser form and inhomogeneous material lead to the additional formation of roughness artifacts, so-called surface structures. In this context, we distinguish between remains of the surface structures from the previous processing step(s) and surface structures introduced by the laser polishing process.

Therefore, to smooth the surface for a given material and initial surface as much as possible, structures introduced by the laser polishing process also need to be considered. The mechanisms of structure formation and their relation towards process parameters are well researched [1,2], from which we have taken the next logical step towards more automated process parameter optimization to help reduce staff time and make the optimization process more efficient: In this paper, we conduct a feasibility study of automated surface structures classification on the domain of white-light interferometry measurements using computer vision.

Showcasing that surface structures can be classified only using white-light interferometry could be more easily automated compared to other methods such as scanning electron microscopy or cross-section analysis and thus would result in a huge reduction in time and cost in process parameter development.

In this context, the de-facto standard, supervised learning, imposed a major hindrance as collecting data on structure formation after laser polishing has not been a major focus of research. To overcome this challenge, in this paper, we compared two approaches with approach A using supervised-learning and approach B, self-supervised-learning (SSL): For approach A, we annotated over 2,500 white-light interferometry images with a total of eight depicted surface structures and trained a ResNet on this data using supervised learning. For approach B, we first pre-trained Vision Transformer (ViT) models on raw WLI data without annotations and later finetune it in a supervised fashion on the same data as used for approach A.

We cover literature on structure formation during laser macro polishing, the data set creation using white-light interferometry images (WLI), preprocessing of the data, and small adjustments to existing SSL algorithms to make them more suitable for processing our domain-specific data. The comparison on supervised-learning vs. SSL is done using a state-of-art supervised learning model for image classification, ResNet models, against self-supervised learning models, pre-trained vision transformer (ViT) models.

2. Related Work

2.1 Surface structures in laser polishing

To smooth a surface for a given material and initial surface as much as possible, the surface roughness introduced by the polishing process itself needs to be as low as possible. [2] gives detailed background information on common surface structures induced by laser polishing and categorizes them by laser macro and micro polishing. In following a brief overview of the information is given. Surface structures are formatted *in this fashion* for better readability. Ongoing, in chapter 3 we restrict the superset of surface structures mentioned here to structures identifiable from white-light interferometry images and structures that lead to a high relative increase in induced roughness.

Initial roughness structures: Ineffective laser polishing may not fully remove initial surface artifacts and thus can lead to remains of the surface structures from the previous processing steps, namely *remains of initial roughness*.

Laser macro polishing: Surface structures which arise during laser macro polishing appear due to the dynamics of melt and solidification front, namely *ripples* and *undercuts* and as a production of plastic deformation and changes in the microstructure, namely *bulges*, *step structures* and *martensite needles*.

Laser micro polishing: Surface structures which arise during laser micro polishing appear due to too low or too high fluence, namely *undercuts*, *micro waviness* and *border bulging*, due to sulfidic and oxidic inclusions released during melting, namely *holes*, carbidic inclusions released during melting, namely *micro structures* and due to plastic deformation, namely *step structures*.

2.2 Convolutional Neural Networks

The use of machine learning to analyze image data, in particular deep learning, has established itself as the backbone of process monitoring since the development of convolutional neural networks (CNN) [8]. Prominent models from this family, like the ResNet [3] model, are composed of multiple blocks connected through residual links, each containing a series of stacked convolution and pooling layers. With this combination of layers, CNNs create spatial hierarchies by progressively summarizing information from lower-level feature maps. This allows higher-level layers to capture larger contextual information and broader patterns, enabling the network to understand relationships between different parts of an image visualization [9]. CNNs are also suitable for transfer learning by pre-training them on large dataset like ImageNet [10] and afterwards retraining the last layers on the actual problem [11].

2.3 Vision Transformer

Originating from the field of Natural Language Processing (NLP), Transformer models [12] have created a new way of processing data sequences. Using self-attention mechanisms, these models can understand global relationships in the data better than CNN models. This model architecture has gained widespread public attention through its use in large language models (LLM), such as GPT-3 [13] (ChatGPT). Vision Transformer [4] (ViT) models transfer the self-attention mechanism to the field of computer vision. In general, ViTs are well suited for transfer learning. When pretrained on a large dataset, ViTs outperform state-of-the-art CNN architectures in classification tasks [14] with fewer computational expenses [4,15].

Compared to CNNs, the ViT shows significantly less image-specific inductive bias. While CNNs ingrain locality, two-dimensional neighborhood structure, and translation invariance throughout every layer, ViT selectively applies local and translation-equivariant properties only in the MLP layers. The multi headed self-attention (MH-SA) blocks instead only operate in a global context. [4]

Another advantage of vision transformers over CNNs is the improved explainability provided by the ability to visualize the various attention heads in the self-attention mechanism [15]. There are also approaches for similar analysis for CNNs, but they require comparatively complex methods to extract such information from their weights [9].

2.4 Self-Supervised Learning

Self-supervised learning (SSL) is a type of machine learning that enables a model to learn how to extract significant features or representations from unlabeled data without relying on explicit human-labeled annotations. This technique trains the model to predict certain characteristics or relationships within the data, utilizing the inherent patterns or structure present in the unlabeled samples [16]. These learned representations can then be applied to downstream tasks like classification, clustering, or regression, often achieving comparable or superior performance to models trained with supervised learning using labeled data [15]. There are two main approaches for SSL in the vision domain: Contrastive and Non-Contrastive Self-Supervised Learning. Contrastive learning aims to learn representations by maximizing the agreement between similar pairs of data points while minimizing the agreement between dissimilar pairs. The core idea is to create a discrimination task where the network learns to distinguish between positive pairs (similar) and negative pairs (dissimilar) [17,18]. The limitation of contrastive learning in computer vision is the high dimensionality of the images because there are countless ways in which one image can differ from another. An optimal set of contrastive images that cover all the ways they can differ from a given image is nearly impossible to create or find [16].

We therefore chose to investigate data2vec [5], a non-contrastive SSL algorithm, and its effect pre-training ViT models on unlabeled data of laser polished surfaces. Data2vec does not rely on calculating dissimilarities for negative image pairs. Instead, it uses two identical encoder networks, a student, and a teacher. The teacher network inputs the original image while the student takes the masked version. The training objective for the student is to predict the average latent representation of the original image in the teacher network based on the masked image. To accomplish this objective, the student network must infer the missing information based on the unmasked parts of the image. By doing so, the student learns to understand its underlying representation.

In detail, the student and the teacher network are both ViT encoder models, but only the student's weights are updated using backpropagation during training. The weights of the teacher θ_t are instead given by an exponentially moving average (EMA) of the student's weights

$$\theta_s: \theta_t \leftarrow \lambda\theta_t + (1-\lambda)\theta_s, \qquad (1)$$

improving the stability of the training and preventing a model collapse. λ linearly increases from λ_0 to the target value λ_e over the first λ_n updates, after which the value is kept constant. The feature and positional embedding weights are excluded from the EMA update rule. These are learned normally within the student network and then shared with the teacher. The training criterion for the student is a smooth L1 loss (equation 2), but for $\beta = 0$, it will change to a mean squared error. Also, only masked parts of the image fed into the student network participate in the loss calculation. The patch encoding for unmasked patches does not influence the loss.

$$\mathrm{SmoothL1}(x,y) = \begin{cases} 0.5 \cdot (x-y)^2/\beta, & \text{if } |x-y| < \beta \\ |x-y| - 0.5 * \beta, & \text{otherwise} \end{cases} \qquad (2)$$

.

3. Application domain, dataset, and dataset preparation for the computer vision task

In this chapter we derive the selection of structures for the image classification task. To guide our decision, a tabulated analysis is given, which categorizes surface structures introduced from chapter 2 into categories: typical measurement methodology, detectability through White Light Interferometry (WLI) and impact on induced roughness. From this information, we derive which structures we want to classify using computer vision. Further, we introduce a dataset which was specifically annotated for this task. At the end we give insights into our image preparation techniques for image classification.

3.1 Characterization of surface structures in white-light interferometry images

Both laser macro and micro polishing induce surface structures making low target roughness hard to achieve. As each surface structure has an individual impact on target roughness, we chose a subset of surface structures, which have either at least a medium impact on target roughness or are simple to detect by human eye or both. An exception was made for undercuts and bulges, as when they appear, their impact on target roughness is quite high and reduction is quite simple by altering process parameters (see [1], chapter 5 for more information on alteration of process parameters). The result of our choice can be depicted from Table 1. Chosen surface structures for the dataset are formatted **bold**. The choice subsumes our observations. Column "Lateral size [µm]" shows the typical range and may differ for unique surfaces or materials. One can observe that the impact on target roughness is highly dependent on the lateral size of the surface structure. This follows a general trend that with larger structures, their amplitude is also higher leading to higher impact on target roughness. For the final subset, we added two more surface structures: laser macro polishing (CW) and laser micro polishing (Pulsed) depending on whether a process was performed on the surface or not. Also these classes do not strictly conform to the notation of "surface structure" as we use it, we think, its an important class for the machine learning models to learn as recognizing these apparent patterns can aid in transferring the AI model to other systems. This allows for the determination of the model's uncertainty based on the recognition rate of these surface characteristics, without the need for manual annotation of a new data set.

Table 1 Surface structure and their impact on roughness.

Surface Structure	Lateral size [µm]	Detectable with WLI [Yes, No]	Impact [-]
Remains init.	100-2000	Yes	High
Ripples	100-640	Yes	High
Undercuts	20-160	No*	High
Bulges	40-320	No*	High
Step struc.	50-640	Yes	High
Martensite	5-15	Yes	Low
Micro wavi.	50-100	No	Medium
Border bulg.	10-50	No	Medium
Holes	10-40	Yes	Medium
Micro struc.	1-2	No	Low

*: Undercuts and bulges cannot be distinguished separately, as the overlapping of the tracks creates a new form of structure that does not resemble either struc-

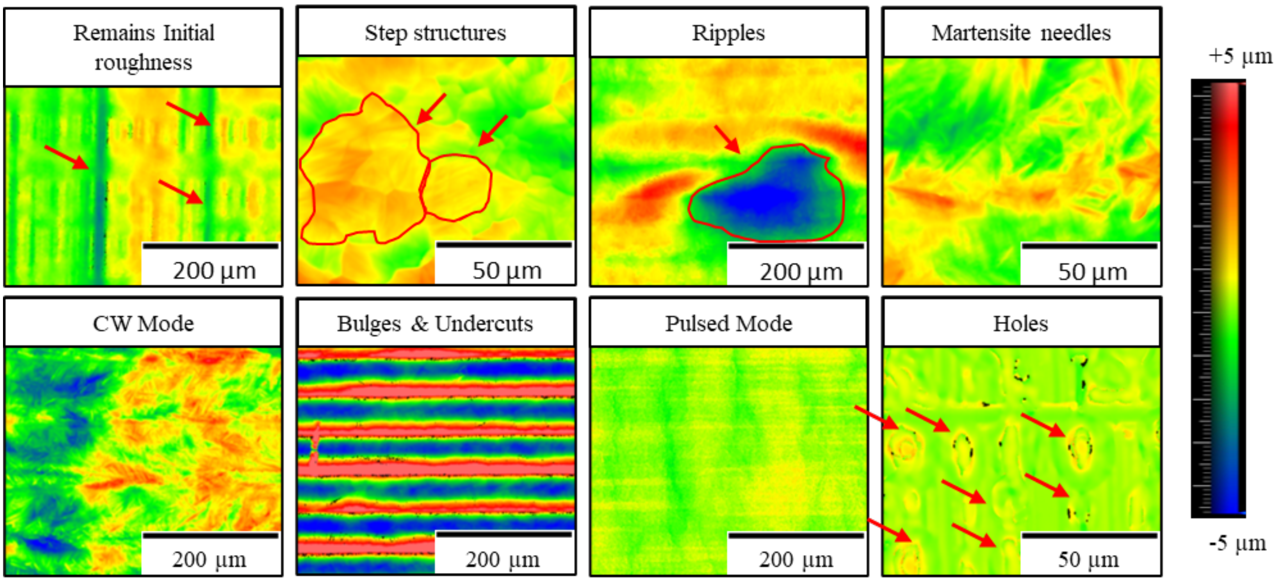

Fig. 1 Images of surface structures induced by laser radiation of continuous wave (CW) mode and pulsed mode colorized in false color scheme (Red: high, blue: low). Pixels represent relative image height in range of [-5, 5] µm. Images are shown in individual resolution to bring forth structures.

An overview of the typical appearance of each chosen surface structure within a WLI image is shown in Figure 1.

3.2 Data acquisition and preprocessing

The surface height maps (XYZ matrices) are acquired with the NexView™ NX2 white light interferometer by Zygo Corporation. The technology enables contactless surface topography measurement at different optical resolutions. For machine learning, height maps were interpreted like an image tensor with a single channel, like a grayscale image. A comprehensive data set, created as part of this work, consists of cross-project measurements from several years of research. Each image goes through a preprocessing pipeline to remove artifacts from the acquisition.

The first preprocessing step is to remove a plane for compensation of any inclination of the sample relative to the measuring device. In the next step we removed outliers larger than five standard deviations and lastly replaced missing values . This is achieved by computing a Gaussian filtered version of each measurement and sampling from this filtered version to superimpose the missing values in the original measurement. Figure 2 shows the effect of the successively applied preprocessing steps.

By merging and pre-processing various WLI measurements from previous projects, it was possible to create a data set comprising 8,362 measurements with a resolution between 0.0005 mm/pixel and 0.0015 mm/pixel. The resolution range was chosen such that all relevant surface structures are clearly recognizable by human eye. 2,536 random images from the LPM dataset were further manually annotated by at least two experts. An image contains a valid annotation if at least one expert found a respective class. A distribution of such labelling can be seen in Figure 3. Since multiple structures can appear superimposed on the same surface, the image classification problem becomes a multi-classification problem. This dataset will be named LPM (Laser Polished Metal) dataset for the remainder of this paper.

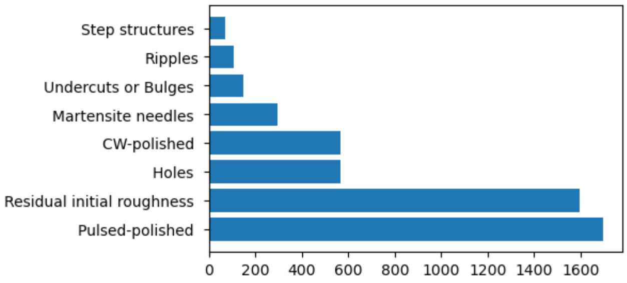

Fig. 3 Class distribution of the LPM dataset.

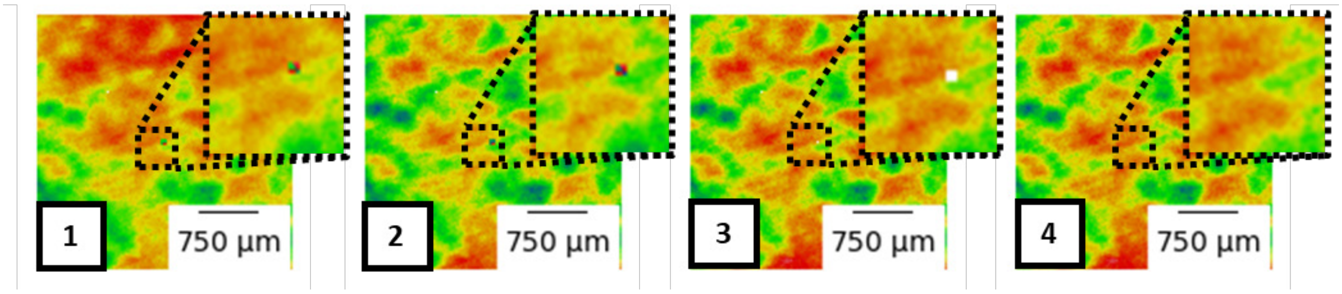

Fig. 2 Effect of the preprocessing pipeline. (1) Raw WLI image, (2) removal of intrinsic plane, (3) removal of spikes, (4), filling missing missing values with gaussian fit (e.g., from acquisition or spike removal).

Table 3 ViT model configurations.

Model	# Heads	# Emb. dim.	# Patch Size	# Params
ViT-S/8-144	6	144	8	3.4M
ViT-S/8	6	384	8	22M
ViT-S/16	6	384	16	21.6M
ViT-B/16	12	768	16	86.1M

Regarding the class distribution, step structures, ripples and undercuts/bulges are underrepresented in relation to other surface structures. This imbalance can have a negative impact on the classification performance for the less represented classes, but the problem is partly compensated by a weighted loss function during finetuning [19]. The weighted loss function assigns higher weights to the underrepresented classes (like step structures, ripples, and undercuts/bulges), thereby penalizing misclassifications of these classes more heavily and encouraging the model to learn them more effectively.

4. Methodology

The aim of this work is to benchmark the capabilities of ViT models for surface structure recognition against ResNet models, which are considered state of the art in the industrial context [8]. For this purpose, different model configurations of both architectures are compared. Both models are not only trained classically supervised from scratch, but also pre-trained model weights (ImageNet1k [10]) are used, publicly available [20]. For the ViT models, the SSL algorithm data2vec is additionally applied in original and a modified form, which is explained later in detail.

4.1 Models

The PyTorch [21] implementation of the ResNet model is used as a benchmark model in three different configurations, which are listed in Table 2. These models require a three channel 224x224 pixel input. We decided not to modify the first layer to take one channeled input, since we planned to use a model pretrained on ImageNet. Instead, the single channeled LPM data was copied to match the three-channel input requirements.

All ViT models used in this work are based on the implementation [22] used by Meta AI for their data2vec paper [5]. Compared to the original ViT paper [4] Meta AI replaces the learnable positional encoding by a learnable relative positional encoding and additionally uses the mean over all patch embeddings as input for the fully connected layer instead of the embedding of the class token. However we decided to use the positional encoding from the original paper, to be able to replace the attention mechanism with FlashAttention2 [23]. This reduced the step time from 240µs to 64µs and the memory usage from 10.9 GB to 2GB, without a significant difference in the training results [24]. The naming of these models follows the scheme from the ViT paper [4]: The letter following ViT, B(ase) or S(mall), represents the number of attention heads. The number after the slash ('/') indicates the patch size. Some model configurations in this work use different sizes for hidden embedding compared to the literature. If this is the case, the size of the hidden embedding is given after a hyphen ('-'). Table 3 describes all ViT model configurations used in this paper, whereby each of these models consists of 12 Attention Blocks. For finetuning on the LPM data set, a head consisting of a single fully connected layer is added to each of these models.

Table 2 ResNet model configurations.

Model	# Params
ResNet-18	11.7M
ResNet-34	21.8M
ResNet-50	25.6M

4.2 Self-supervised Pre-training

To leverage the vast pool of unlabeled data from the LPM dataset, the data2vec (D2V) approach is employed. This method allows the ViT model to extract valuable information and representations from unlabeled data without the need for additional labelling efforts. During the self-supervised pre-training phase, the student network is presented with a 50% randomly masked image, and its objective is to predict the averaged embedding of the last eight layers from the teacher model. To update the teacher model's weights, an exponential moving average (EMA) is utilized with a decay rate λ of 0.999, gradually increasing during training until reaching 0.9999. This adaptive decay rate helps stabilize the training process and allows the teacher model to provide more reliable guidance to the student. The initial learning rate for the training process is set to 0.0005 [5].

D2V hides information in the spatial domain. For datasets like ImageNet, where the task is to classify an image by the represented object or animal, masking in the spatial domain takes out a significant proportion of the underlying representation. If, for example, a dog's head is masked, the model must learn that a dog's head follows a dog's body. Thus, the model learns to recognize a dog's body without knowing what a dog is. For WLI data, masking parts of the data may not be sufficient as many structures occur periodically. Removing parts of these images does not necessarily remove substantial information. We therefore developed a novel extension for the D2V algorithm: Frequency range masking.

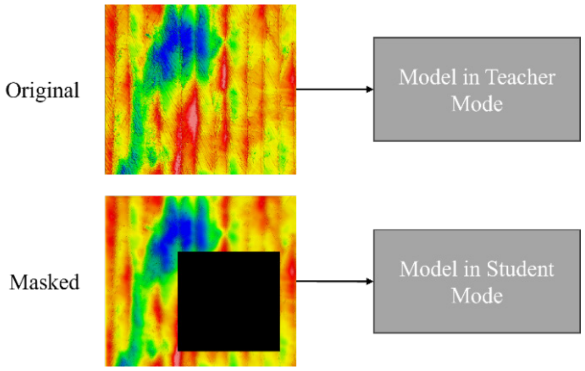

Fig. 4 Schematic representation of data2vec model inputs when combining spatial and frequency masking.

Table 4 Finetuning differently pretrained ViT models.

Model	Pretraining	Validation Acc.
ViT-S/8-144	d2v	0.74
ViT-S/8-144	**d2v + freq**	**0.781**
ViT-S/8	d2v	0.754
ViT-S/8	d2v + freq	0.780
ViT-S/16	d2v	0.741
ViT-S/16	d2v + freq	0.771
ViT-B/16	d2v	0.768
ViT-B/16	d2v + freq	0.779

To remove various frequency ranges, we used band-stop filters based on the ISO11562 [25] standard. Using FFT, these were then applied to the WLI measurements to remove information in different frequency ranges. Interval ranges were chosen based on the spatial size of the individual surface structures and are therefore suitable for the targeted extraction of relevant information from the measurements. Ranges can be found in [1,6]. Figure 4 shows an original image, that is fed into the teacher versus a both spatially and frequency range masked image, which is fed to the student. The task is the student is to predict the same inference as the teacher.

4.3 Experiments

The experiments that were carried out as part of this work can be divided into two categories: *Model architecture* and *data2vec variations*. On the one hand, different model architectures and training algorithms were tested to achieve the best possible recognition of surface structures and on the other hand, different variations of the data2vec SSL algorithm were analyzed to evaluate the encodings learned from unlabeled data.

The performance of the individual models on the LPM validation dataset was evaluated using various metrics. Accuracy was used for easy comparison of the different models during the training process. In the context of structure recognition, a data point is considered correctly classified if each of the occurring structures has been correctly identified (recall: multi-label problem). If, for example, only 3 out of 4 existing structures are recognized or an additional structure is recognized, the data point is incorrectly classified. A single structure is considered recognized if the predicted logits exceed the fixed threshold of 0.5. This metric only allows a first, rough assessment of the model performance, as factors such as class imbalances are not considered. For a detailed analysis of the model performance for the individual structure classes, the Receiver Operating Characteristic (ROC) [26] curve and the Precision-Recall tradeoff is used.

To quantitatively analyze the feature encodings learned through the SSL, the LPM validation dataset is used with the same metrics as for evaluating the finetuned models. The only difference is that during finetuning only the classification head is trained, which consists of only one fully connected layer. This evaluates how well the features learned on unlabeled data are suitable for classifying surface structures.

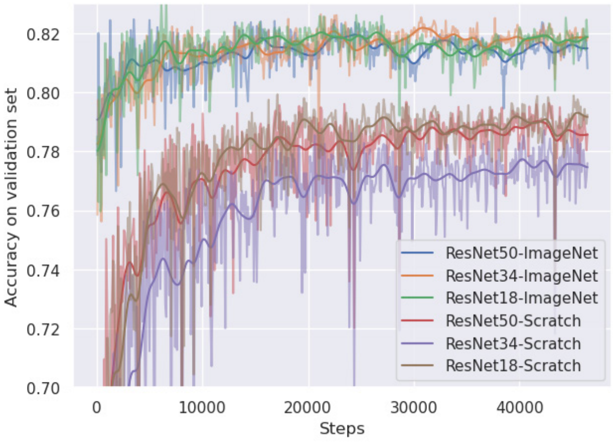

Fig. 5 Development of validation accuracy during the training of the various ResNet models.

A randomly initialized ViT encoder model serves as a benchmark.

An additional qualitative analysis is performed using a t-SNE [27] visualization of the learned features. This shows the similarities between different structures in the learned encoding and whether the encoding is suitable for performing a class separation.

The ResNet model was trained in each configuration from Table 1 for 500 epochs on the LPM data. Each model configuration was trained once from scratch and once with ImageNet1k weights. The ViT models were also trained for 500 epochs from scratch. In addition, each ViT variation was pretrained once with data2vec and once with data2vec + frequency masking for 3,000 epochs. After pre-training, the model head was tuned for 100 epochs each and then the whole model was tuned for another 150 epochs.

The batch size for the finetuning and pre-training of all models was 128. The learning rate for the pre-training of the ViT models was 0.0005 with a linear warmup over 120 epochs and a cosine annealing decay over a single period. The optimizer for pre-training and finetuning the ViT models was an Adam optimizer [28] with $\epsilon = 1e^{-8}$ [5]. For the ResNet finetuning a SGD optimizer with an momentum of 0.9 was used [3]. The loss function for finetuning the ViT and ResNet models is a weighted binary cross entropy loss.

The input size chosen was 224x224 to enable a fair comparison between ResNet and ViT models and to simplify training. In contrast to the ResNet models, ViT models can handle variable input sizes. Depending on the size of the WLI measurements, five to ten random 224x224 patches were sampled.

5. Results and Discussion

5.1 Finetuning ResNet

The first set of experiments focuses on the finetuning of pretrained and randomly initialized ResNet models, as well as randomly initialized ViT models. We opted for this comparison because the implementation of such models involves comparatively little effort due to libraries such as PyTorch.

Table 5 Finetuning ViT models from scratch in comparison with ResNet models.

Model	Pretraining	Validation Acc.
ViT-S/8-144	-	0.725
ViT-S/8	-	0.727
ViT-S/16	-	0.739
ViT-B/16	-	0.746
ResNet-18	-	0.792
ResNet-18	**ImageNet**	**0.818**
ResNet-34	-	0.784
ResNet-34	ImageNet	0.817
ResNet-50	-	0.774
ResNet-50	ImageNet	0.81

Table 4 summarizes the validation accuracies of the different model configurations.

The finetuning of the ResNet models showed that the models trained on ImageNet achieved significantly better accuracy than the models trained from scratch. There are only small deviations in accuracy between the individual model sizes, both for the pretrained and scratch-trained models. Figure 5 also shows that the models that were pretrained on ImageNet achieved significantly better accuracy values from the start than the models without pre-training. Finetuning ViT models from scratch also showed that the different model configurations perform similarly. In general, they achieved a lower accuracy of approx. 0.73 compared to the 0.78 of the ResNet models from scratch and the 0.81 of the pretrained ResNet models.

5.2 Influence of pre-training on ViT

The next set of experiments focused on the self-supervised pre-training of ViT models and the impact of different approaches on the learned encodings, as well as the downstream performance on the LPM data. Table 4 shows how differently pretrained ViT model perform on the validation dataset after finetuning. In general, the additional removal of information by masking certain frequency intervals has a significant effect in that all models that have been pretrained by additional frequency masking show better results in finetuning than models that have been trained with the classical data2vec approach. In addition, all pre trained ViT models show improved performance compared to from scratch trained models.

This trend is also evident in the experiments to evaluate the self-supervised learned features. Figure 6 shows the development of the validation accuracy during the fine-tuning, but only the model head was trained this time. The individual curves represent the averaged validation accuracy across all ViT configurations from Table 5 for each pre-training method used.

The plots show that the pre-training of ViT models leads to them learning encodings that contain relevant features to recognize surface structures on polished surfaces without the need for an annotated data set. In addition, the plot indicates that the features within the learned encoding receive even more structure-relevant information due to the additional frequency masking.

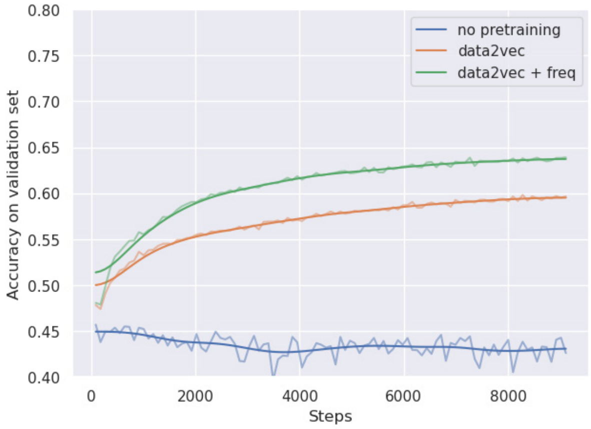

Fig. 6 Development of validation accuracy during the finetuning of only the model head for different pretraining methods, averaged over all ViT configurations.

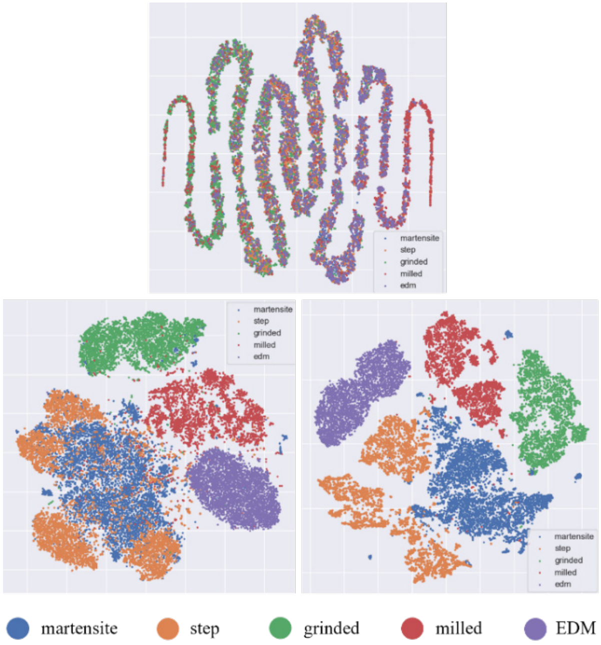

Fig. 7 t-SNE visualization of ViT-S/8-144 encodings after no pretraining(top), data2vec(left), and data2vec + freq (right).

The qualitative analysis of the learned encodings was generated via a t-SNE visualization. Figure 7 shows how similar or dissimilar surface measurements of the different categories are according to the encodings of different pretrained ViT-S/8-144 models. The goal of this visualization is to see if the models were able to learn a semantic understanding of surface texture through self-supervised pretraining. If this is the case, the model encodes surfaces with similar textures with similar features, and natural clusters then form within the t-SNE visualization. If the model encodes the images independently of their surface texture, the individual classes will mix in the visualization. A clear distinction between the individual classes shows that the model has

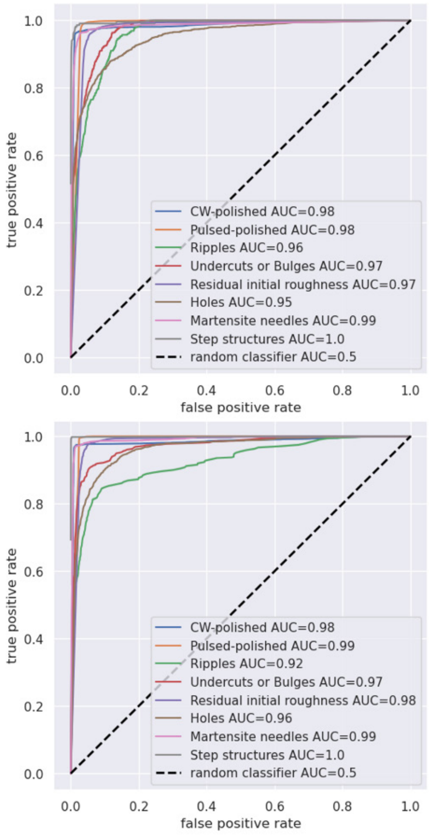

Fig. 8 ROC comparison between ResNet-18 (top) and ViT-S/8-144 (bottom).

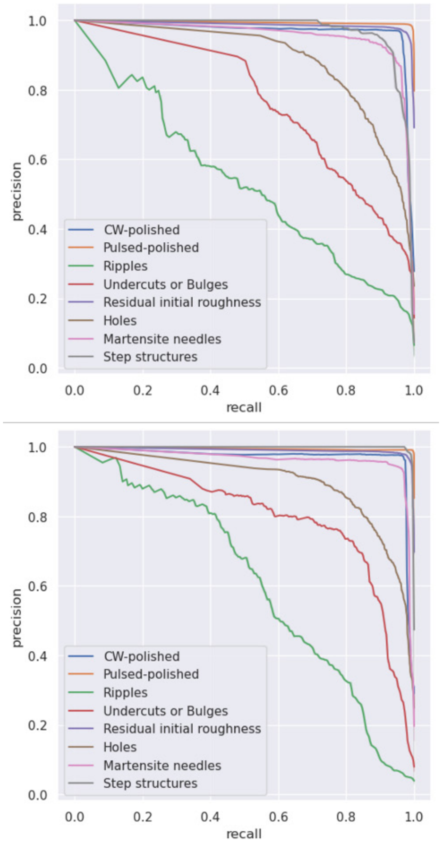

Fig. 9 Precision-recall comparison between ResNet-18 (left) and ViT-S/8-144 (right).

learned a relevant representation of the data through pre-training However, the data set described in Section 3 with its eight different structures is unsuitable for this method, since the structures often overlap, making it impossible to visualize the clusters easily. Therefore, surfaces were selected in which only one structure type predominates. This includes various non-laser polished surfaces: grinded, EDM, milled and laser polished surfaces which have either step or martensite structures.

It can be seen that the encodings from the untrained model is not suitable for distinguishing the surfaces from each other. After pre-training with data2vec, it can be seen that the various pre-treated surfaces can be distinguished from each other and from the laser-polished surfaces. However, the measurements with step and martensite structures are mixed. This changes with the additional masking of frequencies during pre-training. Especially data points with step and martensite structures can be clearly distinguished from each other, which was not the case with the data2vec encoding. Thus, pre-training with masked frequencies has eventually learned an encoding whichs features allow a better distinction between step and martensite structures that were not experienced during pre-training with data2vec and the spatial masks.

Finally, the best-performing ResNet model (ResNet-18) is compared with the best-performing ViT model (ViT-S/8-144 pretrained using data2vec + frequency masking). The ROC curve and the precision-recall tradeoff of the different surface structure classes are compared. The ROC curves of both models shown in Figure 6 are almost identical. The AUC is the same for most classes. Exceptions are Ripples where the ViT model achieves a 0.04 higher AUC and Holes where the ResNet model achieves a 0.01 higher AUC.

The ResNet generally achieves better true positive rates than the ViT for false positive rates (FPR) < 0.1. For FPR > 0.1, however, the ViT achieves better TPR values. This trend is also visible in the PR curves displayed in Figure 9. The PR curve trajectories are generally similar for both models, but ResNet achieves a slightly better trade-off between recall and precision over all classes. Especially for larger recall values > 0.8, ResNet can longer maintain a higher precision value than the ViT model.

Both models can recognize step structures, martensite structures and remnants of the initial roughness very reliably. The precision at a detection rate of >90% is also >90%, making both models suitable for the reliable detection of surface structures in the automation of the laser polishing process. With a detection rate of >80% at a precision of >80%, holes are also detected with sufficient reliability. Only undercuts and bulges, as well as ripples, are detected with insufficient precision at detection rates of >80%. This could be partly due to the type of labelling. Since each WLI measurement was annotated as a whole and then broken down into smaller patches, it cannot be guaranteed that all patches have correct ground truth for very locally occurring structures like holes. This can have a negative impact on both training and the calculation of accuracy, precision, and recall.

5.3 Effort discussion

In addition to the qualitative and quantitative performance differences between ViT and ResNet models, the implementation effort and hardware requirements are factors that should not be neglected. While the ViT models pretrained with data2vec and frequency masking deliver similar results to ResNet models pretrained on ImageNet, the implementation effort for the latter is many times lower. ResNet models and the associated ImageNet weights are publicly provided by PyTorch and the implementation these models to the tasks described in this paper can be implemented by an experienced data scientist in a few days. The computational requirements for training a ResNet-18 model are also already met by mid-range consumer GPUs. ViT models with corresponding ImageNet weights are also provided by PyTorch, but self-supervised learning methods such as data2vec are not. The source code for this is also public, but a transfer to special application like in this work is complicated to implement. Custom modifications such as the addition of frequency masking and the conversion to FlashAttention2 require extensive expert knowledge and can take several weeks. The training of these models can also be performed on consumer GPUs, but especially the pre-training requires several high-end consumer GPUs due to the long training time and large amounts of data. In addition, NVIDIA GPUs have only been optimized for the efficient calculation of Attention since the Ampere architecture [29], which means that FlashAttention2 does not run on older GPU models. In a concrete comparison, an RTX 3090 can feed 1,600 images packed in batches of 64 per second through the ResNet-18 model in evaluation mode without calculating gradients. For the ViT-S/8-144 model, the same GPU could handle 1,400 images with the same setup. This difference is negligible in the later application on the shopfloor.

6. Conclusion and outlook

The presented work investigates on the feasibility of surface structure classification in the context of laser polishing of metals using white-light interferometry (WLI) images. For this purpose, a new data set of WLI images was formed and over 2,500 images annotated on the occurrence of surface structures by process experts from the Fraunhofer Institute of Laser Technology. We cover data set formation, data preparation using a three-step data pipeline and in-depth machine learning comparing Vision Transformers (ViTs) and Convolutional Neural Networks (CNNs).

Our results demonstrate that both model types are effective in recognizing surface structures in white light interferometry measurements. In comparison the state-of-the-art model, ResNet, slightly outperforms the ViT models. Pretraining of ViT models and employing a novel methodology to mask frequency ranges during pre-training have significantly narrowed the performance gap between the ViT models and ResNet. Here, we showed how pre-training can be performed using the self-supervised learning approach, data2vec. The methodology of self-supervised learning shows interesting use cases, where unannotated data is vast and annotation expensive (Compare Figure 7). With this approach ViT models can store information on large, unannotated data sets which allows for easy downstream applications such as anomaly clustering and searching. If the focus is on implementing surface structure recognition with minimal effort, the use of a ResNet model pretrained on the huge, online available dataset "ImageNet" is still recommended according to our research.

In addition to this work, the pre-training of the ViT models can be further optimized by using an even larger data set or by further exploring the method of frequency masking. It would also be of interest to examine whether the inclusion of WLI measurements from different (laser) processes in the pre-training dataset has an influence on the quality of the encodings learned. Secondly, it is worth investigating how pretrained ViT models adapt to few-shot fine-tuning, i.e., downstream applications with little annotated data.

With our findings we help in automated surface structure recognition, with which we close the gap towards more automated process parameter optimization to help reduce staff time and make the optimization process more efficient. A logical next step is to work on a process parameter optimization pipeline which makes use of automatically recognized surface structure information and compare this approach versus standard design of experiments.

References

[1] T. Kiedrowski: Oberflächenstrukturbildung beim Laserstrahlpolieren von Stahlwerkstoffen. Ph.D Thesis, (Shaker, Aachen, 2009).

[2] C. Nüsser, J. Kumstel, T. Kiedrowski, A. Diatlov, and E. Willenborg: Adv. Eng. Mater., 17, (2015) 268.

[3] K. He, X. Zhang, S. Ren, and J. Sun: Deep Residual Learning for Image Recognition, (2015), <http://arxiv.org/pdf/1512.03385v1>.

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, (2020), <http://arxiv.org/pdf/2010.11929v2>.

[5] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli: data2vec: A General Framework for Self-supervised Learning in Speech, Vision and Language, (2022), <http://arxiv.org/pdf/2202.03555v3>.

[6] E. Willenborg: Polieren von Werkzeugstählen mit Laserstrahlung. Ph.D Thesis, (Shaker, Aachen, 2006).

[7] E. V. Bordatchev, A. M. K. Hafiz, and O. R. Tutunea-Fatan: Int J Adv Manuf Technol, 73, (2014) 35.

[8] L. Zhou, L. Zhang, and N. Konz: IEEE Trans. Syst. Man Cybern, Syst., 53, (2023) 105.

[9] S. Gur, A. Ali, and L. Wolf, Visualization of Supervised and Self-Supervised Neural Networks via Attribution Guided Factorization, (2020), <http://arxiv.org/pdf/2012.02166>.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, Li Fei-Fei, K. Li, and L. Fei-Fei: Proc. IEEE Conf. on CVPR 2009, (2009) 248.

[11] M. Huh, P. Agrawal, and A. A. Efros, What makes ImageNet good for transfer learning?, (2016), <http://arxiv.org/pdf/1608.08614>.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin: Attention Is All You Need, (2017), <http://arxiv.org/pdf/1706.03762v5>.

[13] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, Language Models are Few-Shot Learners, (2020), <http://arxiv.org/pdf/2005.14165v4>.

[14] A. Kolesnikov, L. Beyer, X. Zhai, J. Puigcerver, J. Yung, S. Gelly, and N. Houlsby, Big Transfer (BiT): General Visual Representation Learning, (2019), <http://arxiv.org/pdf/1912.11370>.

[15] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin: Emerging Properties in Self-Supervised Vision Transformers, (2021), <http://arxiv.org/pdf/2104.14294v2>.

[16] I. M. Yann LeCun: Self-supervised learning: The dark matter of intelligence, <https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence/>.

[17] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, A Simple Framework for Contrastive Learning of Visual Representations, (2020), <http://arxiv.org/pdf/2002.05709>.

[18] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick: Momentum Contrast for Unsupervised Visual Representation Learning, , <http://arxiv.org/pdf/1911.05722v3>.

[19] M. R. Rezaei-Dastjerdehei, A. Mijani, and E. Fatemizadeh: 2020 27th National and 5th International Iranian Conference on Biomedical Engineering (ICBME), Tehran, Iran, (2020) p. 333.

[20] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala: PyTorch Doc: MODELS AND PRE-TRAINED WEIGHTS, <https://pytorch.org/vision/stable/models.html>.

[21] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala: "PyTorch: An Imperative Style, High-Performance Deep Learning Library", (Curran Associates, Inc, 2019) p. 8024.

[22] facebookresearch: GitHub: Fairseq(-py) sequence modeling toolkit, <https://github.com/facebookresearch/fairseq>.

[23] T. Dao, FlashAttention-2: Faster Attention with Better Parallelism and Work Partitioning, (2023), <http://arxiv.org/pdf/2307.08691>.

[24] Julius Neuß: Hardware and software. PyTorch Profiler; Model: ViT-S/8-144, batch size: 32, GPU: 1xRTX3090, .

[25] ISO 11562, (1996), <https://cdn.standards.iteh.ai/samples/21977/85a5cc6efab449079651c1d3455b783c/ISO-11562-1996.pdf>.

[26] Classification: ROC Curve and AUC, <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>.

[27] Laurens van der Maaten and Geoffrey Hinton: J. Mach. Learn. Res., 9, (2008) 2579.

[28] D. P. Kingma and J. Ba, Adam: A Method for Stochastic Optimization, (2014), <http://arxiv.org/pdf/1412.6980>.

[29] T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré: FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness, (2022), <http://arxiv.org/pdf/2205.14135v2>.

(Received: July 1, 2024, Accepted: March 20, 2025)