Article

CNN-based Laue spot morphology predictor for reliable crystallographic descriptor estimation

Tom Kirstein [1,†], Lukas Petrich [1,†], Ravi Raj Purohit Purushottam Raj Purohit [2], Jean-Sébastien Micha [3], and Volker Schmidt [1,*]

[1] Institute of Stochastics, Ulm University, Ulm, 89096, Germany
[2] Univ. Grenoble Alpes, CEA, IRIG, MEM, Grenoble, France
[3] Univ. Grenoble Alpes, CNRS, CEA, IRIG, SyMMES, Grenoble, France
* Correspondence: volker.schmidt@uni-ulm.de
† These authors contributed equally to this work.

Abstract: Laue microdiffraction is an X-ray diffraction technique that allows for the non-destructive acquisition of spatial maps of crystallographic orientation and strain state of (poly)crystalline specimens. To do so, diffraction patterns, consisting of thousands of Laue spots, are collected and analyzed at each location of the spatial maps. Each spot of these so-called Laue patterns has to be accurately characterized with respect to its position, size and shape for subsequent analyses including indexing and strain analysis. In the present paper, several approaches for estimating these descriptors that have been proposed in the literature such as methods based on image moments or function fitting are reviewed. However, with increasing size and quantity of Laue image data measured at synchrotron sources, some datasets become unfeasible in terms of computational requirements. Moreover, for irregular Laue spots resulting e.g. from overlaps and extended crystal defects, the exact shape and, more importantly, the position is ill-defined. To tackle these shortcomings, a procedure using convolutional neural networks is presented allowing for a significant acceleration of the characterization of Laue spots, while simultaneously estimating the quality of a Laue spot for further analyses. When tested on unseen Laue spots, this approach led to a speedup of 77 times by using a GPU, while maintaining high levels of accuracy.

Keywords: Laue microdiffraction; Laue spot morphology; Laue spot quality; convolutional neural network; polycrystalline material; runtime performance

1. Introduction

With the advent of sources, optics and detectors dedicated to X-ray characterization, X-ray diffraction techniques have become highly used tools for quantitative studies of microstructures in many materials science and engineering applications due to their non-destructive nature and high spatial resolution. Among them, Laue microdiffraction is a spatially resolved X-ray scattering technique which is particularly sensitive to the structural arrangement of the atomic lattice planes. It allows to capture spatial maps of crystallographic orientation and strain state of (poly)crystalline specimens [1,2]. The maps are obtained by systematically scanning the specimens with a polychromatic incident X-ray beam and a subsequent analysis of the resulting scattering patterns recorded on a planar detector. These so-called Laue patterns, see Figure 1, consist of individual Laue spots (i.e., local maxima in the recorded image) which originate from the diffraction phenomena corresponding to geometrical reflections on the lattice planes in the crystals of the probed area in the specimen. Here, each crystal produces a characteristic pattern of spots, all of which superimpose to the recorded Laue pattern. The precise position of the Laue spots in the experimental Laue pattern is crucial for the reliable determination of the structural crystal parameters, in particular the strain or, equivalently, the lattice parameters of the crystallographic unit cell.

More precisely, the usual workflow for analyzing a Laue pattern of a single probed position in the specimen under consideration consists of three main steps [3–5]: First, the local (sub-pixel) maxima in the recorded image are determined, which are assumed to be

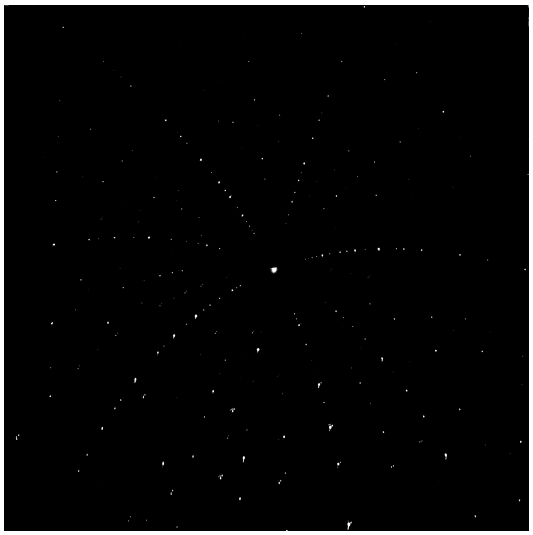

Figure 1. Laue pattern of a ZnCuOCl single crystal (hexagonal unit cell).

the positions of the Laue peaks (peak search routine) featuring the geometrical orientation of the corresponding reflecting lattice planes with respect to the incoming X-ray beam direction. Then, for each peak obtained in this way the corresponding crystal and reflecting planes—namely their Miller indices *hkl*—are sought (indexing routine). Finally, for each crystal the lattice parameters of the unit cell of the crystals are refined (given a reference deviatoric strain tensor for the unit cell) by matching the expected/simulated peak positions with the observed ones from the image.

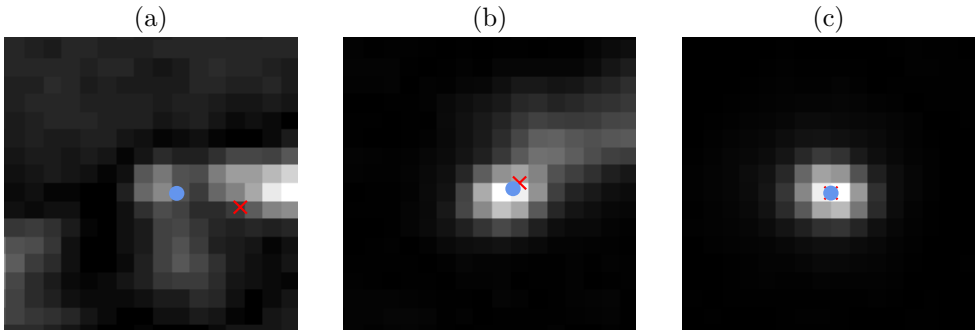

Figure 2. Cutouts of Laue spots of bad (a, b) and good quality (c) with Sim^0-values of 0.738, 0.923 and 0.992, respectively. The corresponding (rescaled) Sim-values are 0, 0.487 and 0.952, respectively. For the definition of Sim^0-values and Sim-values, see Section 2.2.1. Moreover, the blue dot and red cross indicate the peak position predicted by the neural network and the approach based on Gaussian functions, respectively. The cutouts are centered on the peak position predicted by the initial peak search algorithm.

During the peak search step, instead of simply proceeding with the initial peak characterization given by the brightest pixels of a Laue spot, spot shape properties and, most importantly, the peak position have to be estimated with sub-pixel accuracy in order to keep the subsequent analyses as precise as possible. Challenges are polycrystalline materials with small variations of orientations and/or strains. Furthermore, the presence of a complicated spatial distribution of extended crystal defects is problematic. The reason for this is that they cause overlapping Laue patterns originating from the individual crystals and complex/non-circular Laue spots, respectively, see Figure 2. Defining a precise peak position for these multimodal spots that allows for a stable estimation without being affected by small changes of pixel values is hardly possible. The ideal approach would be to split them and treat their sub-peaks individually. However, this would involve a complex

and time-consuming spot analysis that is unfeasible in practice: Currently, several thousand of highly resolved images (4×10^6 pixel) per dataset corresponding to a raster scan sample map have to be processed. Each of these images in turn contains more than 1000 spots corresponding to the superposition of Laue patterns each coming from individual grains. In the near future the demand will raise to several tens of thousands of images with 36×10^6 pixel per image. For this reason, the analysis workflow has to be as fast as possible to keep up with the flood of data measured at synchrotron sources. Recent progress has been made with machine learning approaches to unlock and speed up the limiting indexing step of Laue patterns [6,7], while the previous step of peak search and segmentation has to keep up in order to not be the next performance bottleneck.

Regarding the peak search step, different approaches can be implemented to extract the positions of scattering peaks from digital images. On the one hand, see e.g. [8] employed by [9], it relies on image moments [10] similar to those of bivariate probability distributions. Most importantly, the centroid (which corresponds to the intensity-weighted mean value of the spatial distribution of peak pixels) is used as estimator for the peak position for further analyses. In order to do this, first, the region of interest (ROI) located at each Laue spot, also known as blob, has to be determined, for which the moments are then computed. The classical strategy to build the ROIs is to apply thresholds [8], but procedures based on machine learning [11] have also been proposed. While the image moments are always well defined regardless of whether the spatial distribution of pixel intensities looks like a Laue spot or not, it is crucial to find well-fitting ROIs that contain exactly one peak, in order to obtain valid peak characteristics. Additionally, when a peak is subdivided into several overlapping components, a reliable determination of each individual subpeak is hard to perform automatically, see e.g. Figures 2a,b.

On the other hand, instead of using the non-parametric approach stated above, other methods are based on fitting a parametric function to the Laue spots, usually using least-squares minimization techniques. For this, mainly Gaussian functions (related to the bivariate normal probability distribution), but also Lorentzian functions (related to the Cauchy probability distribution) or combinations of the two, namely pseudo-Voigt functions, are used [12–14]. The parametric approach has the advantage that the descriptors of well-fitting spots are easier to interpret (provided a suitable physical or structural model). Additionally, goodness-of-fit measures can be used as estimates of the model applicability. However, a drawback of the parametric approach is that it can be rather time consuming when the experimental Laue spot differs from the fitted function.

In the present paper, a procedure using convolutional neural networks (CNNs) is proposed to rapidly estimate geometric descriptors of Laue spots and select high-quality peaks for a subsequent strain refinement step. While minimum human intervention is sought for the highest throughput in the Laue analysis workflow, relying on a black-box classification system for selecting peaks would make it hard or even impossible to adapt for difficult specimens/datasets. For this reason, the neural network takes a cutout of the recorded image and returns the precise peak position and key descriptors that are essential for the quality of Laue spots, instead of simply providing a binary decision whether a Laue spot is of good or bad quality. This way, the exact criterion for removing a Laue spot is still explainable and customizable based on these spot descriptors without sacrificing the computational speed.

2. Materials and methods

In Laue diffraction, a crystal is irradiated with X-rays and the resulting diffraction pattern is captured on a detector. The diffraction pattern consists of a series of peaks, which correspond to the diffraction of the X-rays by the crystal lattice. For analyzing the image data collected in this way, the following steps are typically performed (for example in LaueTools [14]):

1. Pre-processing: The recorded Laue diffraction image is pre-processed to remove noise and to improve the contrast. This may involve techniques such as smoothing, filtering, or contrast enhancement.
2. Peak detection: A peak detection algorithm is used to locate the approximate positions of the peaks in the diffraction pattern. This may involve thresholding the image to identify regions of elevated intensity corresponding to the Laue spots, and then using the pixel with the highest intensity as the initial location of the potential peak for each spot.
3. Peak fitting: Once the initial peak positions have been identified, parametric functions (such as a Gaussian or Lorentzian function, which are related to the bivariate normal and Cauchy probability distribution, respectively) are fitted for a precise characterization of the peaks with sub-pixel accuracy.
4. Peak indexing: The positions of the peaks in the diffraction pattern are used to determine the crystal structure. This is done by comparing the observed peak positions to the expected positions of a known crystal structure, or by using a peak-matching algorithm to determine the most likely crystal structure.
5. Data analysis: The precise characterization of the peaks can be further utilized to determine the crystal structure of each individual crystal, namely the crystallographic unit cell lattice parameters or equivalently the strain tensor components. Compiling the results over a dataset of images collected during a sample raster scan allows the imaging of the location of crystals and crystalline defects.

Overall, the process of peak search in Laue diffraction image analysis involves several steps that are designed to identify and analyze the diffraction peaks in the diffraction pattern, in order to obtain structural parameters of experimental crystals. The present paper is concerned with the first step (peak fitting), whose outcome—the accurate characterization of peaks—strongly affects the subsequent steps. The following sections describe the employed methods and materials in detail.

2.1. Description of experimental datasets

The Laue diffraction patterns used in the present study were collected during a series of experiments conducted at the BM32 beamline at the European Synchrotron Radiation Facility (ESRF). The data comprises of five datasets from a variety of materials, namely defect-free single crystals of Ge, Si, ZnCuOCl, and Al_2O_3, as well as polycrystalline Laue patterns from materials (low to high absorption) with strains ranging from 0.001 % to 0.2 %. This includes a dataset from a thick single crystal of Al_2O_3, where the elongation of Laue spots is a result of depth effects. These scans were chosen to cover a range of strain levels, as well as to represent different types of crystalline structures.

The Laue diffraction images were collected using top-reflection geometry, where the 2D detector was mounted at the top of the sample and perpendicular to the incoming X-ray beam (collected scattering angles ranging from $2\theta = 50^\circ$ to 130° and the sample surface was tilted by 40°. The sample-detector distance for all datasets was between 78.5 mm and 79.5 mm. The X-ray energy used in the experiments was in the range from 5 keV to 23 keV with a beam size of about 500 nm $\times$ 500 nm.

The images were recorded using a sCMOS detector with a resolution of 2016 pixel $\times$ 2018 pixel, a pixel size of 73.4 µm and a bit depth of 16 bits per pixel. Further details regarding the experimental setup at the BM32 beamline—including the synchrotron source, the optics and beamline components, are given in [15].

2.2. Geometric characterization of Laue spots

Before the individual Laue spots can be characterized, they first have to be detected and located in the Laue image. For this reason, an initial approximation of the peak positions is obtained by the peak search algorithm implemented in the LaueTools software package, see [14]. Specifically, connected components in the thresholded Laue image, corresponding to the Laue spots, are determined and the initial peak position of each spot

is then given by the position of the maximum pixel intensity or the center of mass of these regions. However, not all spots obtained in this way are useful with respect to subsequent Laue image analysis. In particular, it is important to avoid irregular or asymmetrical spots whose description by a simple single crystal model is not relevant. Including such spots in a strain refinement based on a single crystal model provides a poor average estimation of strain levels in the case of crystal defects or assemblies of crystals. To detect these hereafter called low-quality Laue spots rapidly, a new algorithm was developed to model shape properties of 2D Laue spots.

For that purpose, a Laue image is written as a map $I : W \to \mathbb{R}$, where $W \subset \mathbb{Z}^2$ is a rectangular pixel array and $I(v)$ is the 16 bits integer value of the pixel located at $v = (v_1, v_2) \in W$. The array of pixel intensity values of an individual Laue spot is usually described by an 2D Gaussian function [16]. More precisely, if we let $s = (s_1, s_2) \in W$ be an initial guess of a Laue spot peak position, then normalization constants $\alpha, \beta > 0$ should exist such that for small vectors $a = (a_1, a_2) \in \mathbb{Z}^2$ it holds that $I(s_1 + a_1, s_2 + a_2) \approx \alpha \cdot g(a_1, a_2) + \beta$. Here $g : \mathbb{R}^2 \to \mathbb{R}$ is a normalized (bivariate) Gaussian function (also known as the probability density of the bivariate normal distribution with vanishing covariances) given by

$$g(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2} \exp\left[\left(\frac{x_1 - \mu_1}{\sigma_1}\right)^2 + \left(\frac{x_2 - \mu_2}{\sigma_2}\right)^2\right] \qquad \text{for all } (x_1, x_2) \in \mathbb{R}^2,$$

where $\mu_1, \mu_2 \in \mathbb{R}$ and $\sigma_1, \sigma_2 > 0$ are some location and scale parameters, respectively. Thus, to analytically describe the Laue spot at the initially predicted peak position $s = (s_1, s_2)$, the restriction $I(A(s))$ of the Laue image $I : W \to \mathbb{R}$ to the 32 pixel $\times$ 32 pixel cutout

$$A(s) = \{s_1 - 16, \ldots, s_1 + 15\} \times \{s_2 - 16, \ldots, s_2 + 15\} \subset W$$

around $s = (s_1, s_2)$ is considered, where $I(A(s))$ denotes the elementwise application of I on $A(s)$, in the sense that the output is a matrix $I(A(s)) \in \mathbb{R}^{32\times 32}$. The normalization constants $\alpha, \beta > 0$ as well as the location and scale parameters $\mu_1, \mu_2 \in \mathbb{R}$ and $\sigma_1, \sigma_2 > 0$ of the Gaussian function g are then fitted to the restricted image $I(A(s))$ using a gradient descent algorithm as implemented in the LaueTools software package [14].

In the following, we show how 2D Gaussian functions, fitted to image data, can be utilized to effectively characterize Laue spots with respect to their size, shape and position, in order to judge their usefulness for the subsequent strain analysis.

2.2.1. Similarity to a Gaussian function

As noted in [5], not all Laue spots are well described by Gaussian functions. Therefore, before analyzing the fitted parameters of a Gaussian function, we first have to verify that it fits well to the image data of the considered Laue spot candidate. For that purpose, we investigate the goodness of fit on the 32 pixel $\times$ 32 pixel cutout $A(s) \subset W$ introduced above. More specifically, we compute a descriptor of similarity of $I(A(s))$ and $\alpha g(A(s) - s) + \beta$, given by

$$\mathrm{Sim}^0(s) = \frac{\sum_{a_1,a_2\in A(s)} I(a_1, a_2) \cdot (\alpha g(a_1 - s_1, a_2 - s_2) + \beta)}{\sqrt{\sum_{(a_1,a_2)\in A(s)} I^2(a_1, a_2) \cdot \sum_{(a_1,a_2)\in A(s)} (\alpha g(a_1 - s_1, a_2 - s_2) + \beta)^2}}.$$

However, when using this similarity descriptor directly, Laue spot candidates featuring non-Gaussian properties, such as asymmetry (see Figure 2), will still have a high Sim^0-value. On the other hand, Laue spot candidates which appear to be well described by a Gaussian function will usually have a Sim^0-value of above 0.95. In order to make the entire

interval $[0,1]$ descriptive with respect to the goodness of fit, we instead use the rescaled value $\mathrm{Sim}(s)$ to evaluate the similarity of a Laue spot at $s \in W$ to a Gaussian function, where

$$\mathrm{Sim}(s) = \begin{cases} 0, & \text{if } \mathrm{Sim}^0(s) \leqslant 0.85, \\ \dfrac{\mathrm{Sim}^0(s) - 0.85}{0.15}, & \text{else.} \end{cases}$$

In cases where the Sim-value is low, the description of Laue spots using the fitted Gaussian functions is not accurate. Thus, such Laue spots should not be used for further analysis.

2.2.2. Precise peak position

For the analysis of Laue patterns, accurate estimation of peak positions is of great importance. For a peak located at pixel $s = (s_1, s_2)$, the peak of the fitted Gaussian function is given by $\mathrm{Pos}(s) = (s_1 + \mu_1, s_2 + \mu_2)$. This precise peak position can achieve sub-pixel accuracy to reliably locate those Laue spots which have a high Sim-value. Similar to how the fitted location parameters μ_1, μ_2 of the Gaussian function are used in order to estimate the sub-pixel peak position of Laue spots, the scale parameters σ_1, σ_2 can be used to describe their shape and size.

2.2.3. Shape and size descriptors

As mentioned above, it is of great importance to only consider high-quality Laue spots for strain determination in order to achieve the highest reliability. This quality does not only depend on the Gaussian similarity of Laue spots, but also on their shape and size. Hence, in general, large elongated spots (even symmetrical) are discarded. For that reason, we consider the aspect ratio and the size of spots as featuring descriptors, which are given by

$$\mathrm{Asp}(s) = \frac{\min(\sigma_1, \sigma_2)}{\max(\sigma_1, \sigma_2)} \qquad \text{and} \qquad \mathrm{Area}(s) = \pi \sigma_1 \sigma_2 \,,$$

respectively. This multivariate description approach can be utilized to reject or accept spots for strain analysis. However, performing the 2D Gaussian fitting of Laue spots, as stated above, for a large number of Laue images and spots can be rather time consuming. Thus, we propose an alternative method for the estimation of Laue spot descriptors, which is based on CNNs.

2.3. CNN-based prediction of geometric spot descriptors

In order to efficiently determine the peak position $\mathrm{Pos}(s)$, size $\mathrm{Area}(s)$ and aspect ratio $\mathrm{Asp}(s)$ of a Laue spot with preliminary peak position at $s \in W$ we will estimate these descriptors directly from the image data $I(A(s))$ corresponding to the Laue spot at s, instead of using the iterative gradient descent-based approach considered in Section 2.2.

Since the input $I(A(s))$ is an image, conventional methods such as linear regression, random forests or dense neural networks [17] are not suitable. In particular, these approaches do not maintain the spatial correlation of input arguments (i.e., the values $I(v)$ for $v \in A(s)$), which is of great importance for analyzing image data. For that reason, CNNs, which leverage this spatial structure using convolutions [18], have been popularized in the literature. In the following, we present details regarding the specific network architecture used in the present paper. For more information on CNNs in general, we refer to [19] and [18].

2.3.1. Adjusted descriptors

To characterize Laue spots by means of CNNs, some of the descriptors introduced in Section 2.2, namely the size $\mathrm{Area}(s)$ and the peak position $\mathrm{Pos}(s)$, are not suitable. For this reason these spot descriptors are adjusted. Recall that the (precise) peak position of the Laue spot at $s \in W$, is described by $\mathrm{Pos}(s) = (\mathrm{Pos}_1(s), \mathrm{Pos}_2(s)) \in \mathbb{R}^2$, while the values of $\mathrm{Sim}(s)$ and $\mathrm{Asp}(s)$ belong to the interval $[0,1]$. Furthermore, note that a spot has low

quality if the precise peak position $\mathrm{Pos}(s)$ deviates coordinate-wise by more than 1 from the initially estimated peak position s, which occurs if $|\mathrm{Pos}_1(s) - s_1| > 1$ or $|\mathrm{Pos}_2(s) - s_2| > 1$. This is due to the fact that the precise peak position $\mathrm{Pos}(s)$ is just a refinement in the sub-pixel scale of the otherwise correct initial peak position s. Therefore, the values of $\mathrm{Pos}_1(s)$ and $\mathrm{Pos}_2(s)$ are only of interest when $(\mathrm{Pos}_1(s) - s_1, \mathrm{Pos}_2(s) - s_2) \in [-1, 1]^2$. If the precise peak position $\mathrm{Pos}(s)$ deviates coordinate-wise by more than 1 from the initially estimated peak position s, errors in the prediction of other descriptors could be potentially large which would overemphasize the effect of those low-quality spots during training. Thus, instead of estimating $\mathrm{Pos}_i(s)$ for $i \in 1, 2$, we consider the adjusted position $\mathrm{Pos}^\star(s) = (\mathrm{Pos}_1^\star(s), \mathrm{Pos}_2^\star(s))$ given by

$$\mathrm{Pos}_i^\star(s) = \begin{cases} 0, & \text{if } \mathrm{Pos}_i(s) - s_i \leqslant -1.1, \\ 1, & \text{if } \mathrm{Pos}_i(s) - s_i \geqslant 1.1, \\ \dfrac{\mathrm{Pos}_i(s) - s_i + 1.1}{2.2}, & \text{else,} \end{cases}$$

for $i = 1, 2$. Note that the precise peak position $\mathrm{Pos}(s)$ can be reconstructed from $\mathrm{Pos}^\star(s) = (\mathrm{Pos}_1^\star(s), \mathrm{Pos}_2^\star(s))$ for spots such that $\mathrm{Pos}(s) - s \in [-1, 1]^2$. For the remaining (low-quality) spots, the precise peak position is of no interest as its estimation cannot be assumed to be reliable and these spots will thus be omitted in further analysis.

Similarly, the size descriptor $\mathrm{Area}(s)$, given by $\mathrm{Area}(s) = \pi\sigma_1\sigma_2$, takes values in the set of positive real numbers $\mathbb{R}_+ = (0, \infty)$. However, since large Laue spots are considered to be of low quality, instead of estimating the quantity $\mathrm{Area}(s)$ directly we consider the adjusted size

$$\mathrm{Area}^\star(s) = \begin{cases} 1, & \text{if } \mathrm{Area}(s) \geqslant 5\pi, \\ \dfrac{\mathrm{Area}(s)}{5\pi}, & \text{else,} \end{cases}$$

which takes values in the interval $[0, 1]$. Again, for high-quality spots this adjustment can be reversed. Thus, the quantitative characterization of a Laue spot at $s = (s_1, s_2) \in W$ is given by the descriptor vector $\mathrm{Desc}(s) = (\mathrm{Sim}(s), \mathrm{Pos}_1^\star(s), \mathrm{Pos}_2^\star(s), \mathrm{Asp}(s), \mathrm{Area}^\star(s))$. In order to estimate this descriptor vector directly from image data, the CNN architecture stated in the next section is used.

2.3.2. Convolutional neural network architecture

The CNN architecture considered in this paper comprises a fully convolutional stage followed by a fully connected stage, see Figure 3 for a schematic overview. In the fully convolutional stage, convolutional layers with kernel size 3×3 are iteratively applied with the goal of identifying important features in the images through the convolution with several trainable kernels. Additionally, batch normalization layers are inserted after each convolutional layer. Max-pooling layers with strides of size 2 are applied after the three blocks consisting of two convolutional layers each. The pooling layers serve the purpose of downsampling the image size, and, thus, increasing the effective field of view of the subsequent block of convolutional layers without increasing the number of weights. Since the input image has a resolution of 32 pixel $\times$ 32 pixel (i.e., an array of shape $(32, 32, 1)$), the output of the fully convolutional stage is a $(4, 4, 128)$-array, which is then used as the input of the fully connected stage.

The following fully-connected dense layers have 112, 56, 22 and 5 neurons, respectively, where the last layer represents the final output of the CNN. The activation function of the convolutional layers and the inner dense layers is the ReLU function. The output layer uses a sigmoid function, which can only take values in the interval $[0, 1]$. In order to use this neural network to the precise estimation of the descriptor vector $\mathrm{Desc}(s)$, the network has first to be calibrated via training, as described below.

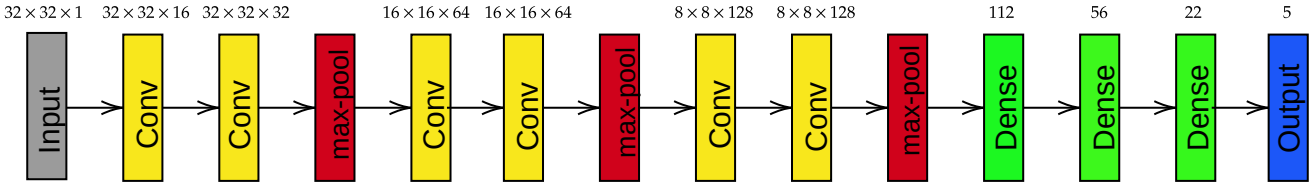

Figure 3. CNN architecture used for the estimation of Laue spot descriptors. The values above the convolutional layers and the dense/output layers specify the number of filters and the number of neurons, respectively. For better clarity, batch normalization layers are included in layer Conv.

2.3.3. Training procedure

Given the architecture described in Section 2.3.2, the neural network can be seen as a function $f_\theta : \mathbb{R}^{32\times 32} \to [0,1]^5$ where, for a given input image $I(A(s))$, the output depends also on the weights $\theta \in \mathbb{R}^{347493}$ of the network. To ensure that the network output $f_\theta(I(A(s)))$ approximates the ground truth descriptor vector

$$\mathrm{Desc}(s) = (\mathrm{Sim}(s), \mathrm{Pos}_1^\star(s), \mathrm{Pos}_2^\star(s), \mathrm{Asp}(s), \mathrm{Area}^\star(s))$$

reasonably well, we first need to find suitable weights θ by training the network. For that purpose, a stochastic gradient descent algorithm with mini-batches of size 32 is used. Specifically, the ADAM optimizer [19,20] with a learning rate of 0.001 is applied to minimize the training loss.

The training loss is characterized by means of the d-dimensional mean absolute error (MAE), where $d = 5$ corresponding to the five descriptors $\mathrm{Sim}(s)$, $\mathrm{Pos}_1^\star(s)$, $\mathrm{Pos}_2^\star(s)$, $\mathrm{Asp}(s)$, $\mathrm{Area}^\star(s)$ considered in this paper. In the general case of d descriptors for some $d > 0$, the loss is given by

$$\mathrm{MAE}\left(\mathbf{y}^{\mathrm{true}}, \mathbf{y}^{\mathrm{pred}}\right) = \frac{1}{nd}\sum_{i=1}^{n} \|y_i^{\mathrm{true}} - y_i^{\mathrm{pred}}\|_1,$$

where $\mathbf{y}^{\mathrm{true}} = (y_1^{\mathrm{true}}, \ldots, y_n^{\mathrm{true}}) \in \mathbb{R}^{n\times d}$ is an ensemble of n true descriptor vectors and $\mathbf{y}^{\mathrm{pred}} = (y_1^{\mathrm{pred}}, \ldots, y_n^{\mathrm{pred}}) \in \mathbb{R}^{n\times d}$ denotes their predictions, with the absolute value norm (also known as L_1-norm) $||y||_1 = \sum_{j=1}^{d} |y_j|$ for any $y = (y_1, \ldots, y_d) \in \mathbb{R}^d$.

In total, 70 epochs were conducted, with 100 training steps each. To avoid overfitting, the initial model development was performed on a small subset of the first dataset, leading to the choice of various hyperparameters such as the number of epochs and the number of filters in the convolutional stage of the architecture. Additionally, in order to synthetically increase the variance in the training data, input images are shifted, rotated and reflected during training, with a corresponding adjustment of the position descriptor $\mathrm{Pos}^\star$.

2.3.4. Ground truth data

As mentioned in Section 2.1, five Laue microdiffraction scans are available for the training and evaluation of the neural network. From each of these scans, a ground truth dataset $\mathcal{D}_i$, for $i = 1, \ldots, 5$, is obtained by first identifying all Laue spot candidates with the initial peak search algorithm and, then, fitting a 2D Gaussian function to each spot candidate, see Section 2.2. From this, for a spot candidate at $s \in W$, the true descriptor vector $\mathrm{Desc}(s) = (\mathrm{Sim}(s), \mathrm{Pos}_1^\star(s), \mathrm{Pos}_2^\star(s), \mathrm{Asp}(s), \mathrm{Area}^\star(s))$ is determined, which is combined with the image cutout $I(A(s))$. Thus, in summary, the ground truth datasets $\mathcal{D}_1, \ldots, \mathcal{D}_5$ consist of pairs of input images and corresponding ground truth descriptor vectors for each Laue spot candidate. Since the datasets contain up to 5 000 000 Laue spot candidates, a

quasi-random subsampling was conducted to limit the number of Laue spot candidates to 46 000 for each dataset.

During training, neural networks emphasize learning of the dependencies between typical values of inputs and outputs. More specifically, these typical values appear more often during training, and thus there is more pressure (i.e., higher training loss) to predict the dependency between them correctly than for other values which occur less often. This can lead to large errors if training and test data follow different probability distributions, as shown in [21]. Unfortunately, the distribution of the similarity descriptor Sim varies significantly across the five ground truth datasets, see Figure 4, and this is likely to be true for datasets on which the CNN is applied in the future. To offset these differences, network training is conducted on resampled datasets denoted by $\mathcal{D}_1^{\text{res}}, \ldots, \mathcal{D}_5^{\text{res}}$—in fact, as detailed in Section 3 below, only a selection of these datasets is used during training. The goal of this resampling procedure is to approximate a standard uniform distribution for Sim, meaning that there are no typical values in the training data. For that purpose, the interval $[0, 1]$ is partitioned into 12 bins and ground truth pairs (i.e., Laue spots and their descriptor vectors) are assigned to each bin with respect to their similarity descriptor Sim until they contain 2000 elements. The procedure is also stopped when no pairs remain. For the resulting resampled datasets $\mathcal{D}_1^{\text{res}}, \ldots, \mathcal{D}_5^{\text{res}}$, the histograms of the descriptor Sim are shown in the bottom row of Figure 4.

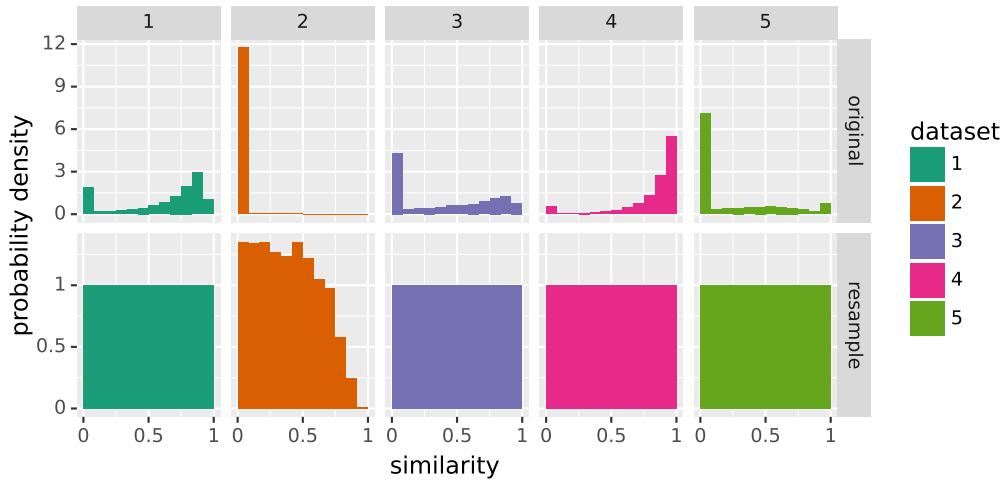

Figure 4. Histograms of the similarity descriptor Sim for original (top) and resampled datasets (bottom).

3. Results

The goal of this section is to evaluate the model performance on previously unseen data. Results are presented that consider all Laue spots, but additionally special focus is put on Laue spots that are well-fitted by the Gaussian functions.

3.1. Evaluation procedure

To evaluate the final prediction performance, the available ground truth datasets $\mathcal{D}_1, \ldots, \mathcal{D}_5$, see Section 2.3.4, are split into disjoint training and test data. Note that, for the evaluation to be accurate, test and training data have to be uncorrelated. However, Laue spot candidates of a given scan tend to be highly correlated, especially when many Laue patterns are collected at several places on a same crystal during the sample raster scan. Therefore, we use the following cross-validation method to ensure that the performance measures considered in the present paper are representative: for each choice of four datasets out of the five available datasets, a CNN model is trained on the union of the resampled data (e.g., $\mathcal{D}_2^{\text{res}} \cup \cdots \cup \mathcal{D}_5^{\text{res}}$) and tested on the remaining dataset (e.g., $\mathcal{D}_1$). Note that, in order to evaluate the prediction performance under real world conditions, testing is

conducted on the original dataset (e.g., $\mathcal{D}_1$) that is not resampled. So, in summary, a total of five models—corresponding to the five so-called cross-validation folds—are built with the same architecture and hyperparameters, but trained on different combinations of four (resampled) datasets, where the model with the number i uses $\mathcal{D}_i$ as test data (for $i = 1, \ldots, 5$). The learning progress of the models is illustrated in Figure 5 for both, the training data and the unseen test data.

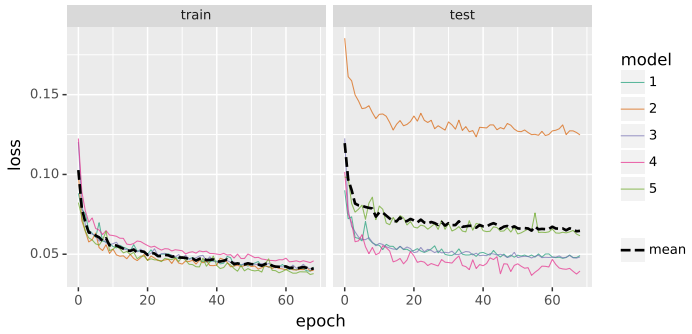

Figure 5. Loss functions for training data (left) and unseen test data (right) during the training of the five CNN models as well as their averaged progress (black dashed lines).

In the following, only the unadjusted spot descriptors are discussed, because they are of practical concern and thus easier to interpret. Since the networks return predictions for the adjusted descriptors (i.e., $\mathrm{Area}^\star(s)$ and $\mathrm{Pos}^\star(s)$), we inverted the formulas given in Section 2.3.1 in order to obtain predictions for the unadjusted descriptors. Recall that the adjustments of the descriptors used for training makes them robust against outliers. However, this is no longer true for their unadjusted counterparts. For example, some badly fitted Gaussian functions lead to extremely high values of the spot size descriptor $\mathrm{Area}(s)$ that are much larger than the cutout $A(s)$. Since these outliers lead to skewed error scores, they are truncated. More specifically, for a Laue spot at $s \in W$, the value of $\mathrm{Area}(s)$ is set to $\min\{\mathrm{Area}(s), 25\pi\}$, i.e., the area of a disk with a diameter of 10 pixel. Similarly, the components of the position vector $\mathrm{Pos}(s)$ are set to $\max\{\min\{\mathrm{Pos}_i(s) - s_i, 15\}, -16\}$ for $i = 1, 2$ to ensure that $\mathrm{Pos}(s)$ is inside the cutout $A(s)$.

In order to quantify the prediction errors, the (one-dimensional) mean absolute error, see Section 2.3.3, is considered for three different spot descriptors. Namely, $\mathrm{MAE}_{\mathrm{Sim}}$ for the similarity to a Gaussian function, as well as $\mathrm{MAE}_{\mathrm{Area}}$ and $\mathrm{MAE}_{\mathrm{Asp}}$ for the size and aspect ration of spots, respectively. Furthermore, since the peak position $\mathrm{Pos}(s)$ is a two-dimensional spot descriptor, the averaged Euclidean norm $\overline{\mathrm{PosErr}}$ of the position error vector is employed, which the vector obtained by subtracting the true position from the predicted position. Last not least, the d-dimensional coefficient of determination R^2 is considered, where

$$\mathrm{R}^2(\mathbf{y}^{\mathrm{true}}, \mathbf{y}^{\mathrm{pred}}) = 1 - \frac{\sum_{i=1}^{n} \|y_i^{\mathrm{true}} - y_i^{\mathrm{pred}}\|_2^2}{\sum_{i=1}^{n} \|y_i^{\mathrm{true}} - \overline{y^{\mathrm{true}}}\|_2^2}$$

for an ensemble of n true values of a (d-dimensional) descriptor $\mathbf{y}^{\mathrm{true}} = (y_1^{\mathrm{true}}, \ldots, y_n^{\mathrm{true}}) \in \mathbb{R}^{n \times d}$ and their corresponding predictions $\mathbf{y}^{\mathrm{pred}} = (y_1^{\mathrm{pred}}, \ldots, y_n^{\mathrm{pred}}) \in \mathbb{R}^{n \times d}$, with the mean value $\overline{y^{\mathrm{true}}} = \frac{1}{n} \sum_{i=1}^{n} y_i^{\mathrm{true}}$ and the Euclidean norm $\|y\|_2 = \sqrt{\sum_{j=1}^{d} y_j^2}$ for any $y = (y_1, \ldots, y_d) \in \mathbb{R}^d$.

Note that the coefficient of determination R^2 relates the variation of the residuals to the variation with respect to the (single) estimator $\overline{y^{\mathrm{true}}}$. In the scalar case, i.e. for $d = 1$, the latter is proportional to the variance in the data $\mathbf{y}^{\mathrm{true}}$). The best possible value of R^2 is 1, and 0 is obtained if the predictive power (in the sum-of-squares sense) is equal to that of $\overline{y^{\mathrm{true}}}$, but it can get arbitrarily low (taking even negative values), see [22] for more

information regarding the scalar case. In the following, the coefficient of determination is denoted by R^2_{Sim} for the similarity to a Gaussian function, by R^2_{Pos} for the spot position, by R^2_{Area} for the spot area, and by R^2_{Asp} for the aspect ratio of spots.

3.2. Numerical results for all Laue spots

In Table 1, the results are presented which have been obtained for various error scores on the test data of each cross-validation fold as well as for aggregated error scores, where the aggregated error scores were obtained by aggregating the predictions of all five CNN models on the respective test data. It is important to emphasize that the ground truth for all predicted descriptors is directly derived from the Gaussian functions fitted to the Laue spots. In other words, for spots that can only be inadequately described by a Gaussian function, the descriptors take almost arbitrary values, which are hard, if not impossible, to estimate directly from the image data, see Figure 2a. This can be seen by analyzing the dependence of the similarity descriptor Sim on the prediction errors.

Table 1. Results obtained for various error scores of the five CNN models on the respective test datasets, as well as for aggregated error scores.

model	sample size	MAE_{Sim}	R^2_{Sim}	$\overline{PosErr}$	R^2_{Pos}	MAE_{Area}	R^2_{Area}	MAE_{Asp}	R^2_{Asp}
1	46000	0.057	0.916	0.213	0.298	0.907	0.350	0.075	0.415
2	46000	0.011	0.621	4.155	0.072	27.584	-0.264	0.184	0.047
3	46000	0.074	0.912	0.213	0.385	0.889	0.348	0.089	0.316
4	46000	0.051	0.889	0.051	0.819	0.249	0.318	0.055	0.260
5	46000	0.147	0.594	2.282	0.218	25.297	-0.370	0.136	0.061
aggregated	230000	0.068	0.913	1.383	0.149	10.985	0.129	0.108	0.344

In Figure 6, four top views of 2D histograms are shown, corresponding to different pairs of prediction errors, where the colors indicate the heights of the histogram values at the corresponding positions. These histograms were computed by using each Laue spot in the five test datasets, except those identified as outliers. Note that the prominent line in the bottom-left plot of Figure 6, corresponding to the absolute error of predicted similarity, is caused by spots whose predicted similarity is close to 0, but whose true similarity is non-zero.

3.3. Numerical results for good-quality spots

It is clear that the majority of large errors in Figure 6 occurs for spots whose similarity descriptor Sim is equal to (almost) zero, see also Figure 2. For this reason, a case study has been conducted where a good quality spot was defined as a spot whose similarity descriptor Sim exceeds the threshold of 0.5. Furthermore, large deviations between the initial guess of the peak position $s \in W$ and the position $\mathrm{Pos}(s)$ deduced from the fitted Gaussian function are also a sign of bad spots. For this reason, spots are only retained if it holds that $\|\mathrm{Pos}(s) - s\|_2 \leqslant 0.9$. For sake of simplicity, the criterion considered in this paper for good quality spots only depends on the similarity descriptor and the peak position, but more complex criteria based additionally on the spot size $\mathrm{Area}(s)$ and/or the aspect ratio $\mathrm{Asp}(s)$ could also be used, in dependence on the nature the datasets under consideration. Note that in practice the true value of the similarity descriptor $\mathrm{Sim}(s)$ is unknown and only the predicted value is available. Thus, a few good quality spots are misclassified as bad quality spots and vice versa, see Table 2. By restricting the error scores presented in Table 1 to the predicted good spots, much better results are obtained, as shown in Table 3.

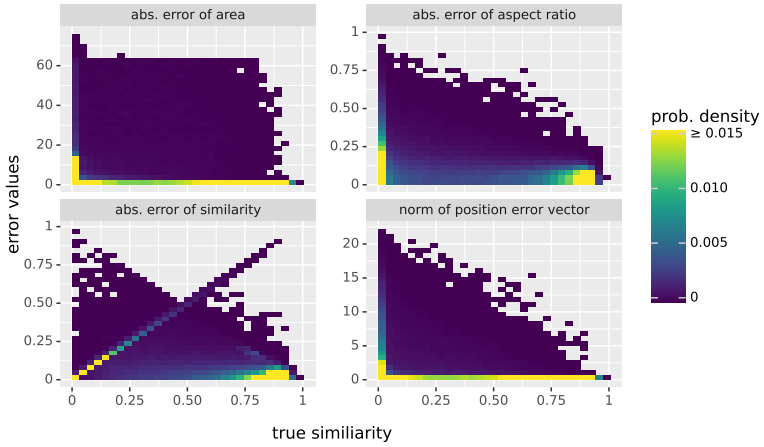

Figure 6. Top views of 2D histograms for different pairs of prediction errors, where the (true) similarity Sim is plotted along the x-axis, and the absolute prediction error for size (top left), aspect ratio (top right), and similarity (bottom left) as well as the norm of the position error vector (bottom right) are plotted along the y-axis. Each underlying data point corresponds to a Laue spot in the five test datasets $\mathcal{D}_1, \ldots, \mathcal{D}_5$. To improve visibility, the colormap was cut off at 0.015, and outliers were removed (instead of truncating them).

Table 2. Confusion matrix for good and bad quality spots.

		true	
		bad	good
predicted	bad	0.552	0.021
	good	0.013	0.415

Table 3. Values of various error scores for *good* quality spots, computed for the CNNs on the corresponding test datasets, as well as for aggregated error scores.

model	sample size	$\mathrm{MAE}_{\mathrm{Sim}}$	$\mathrm{R}^2_{\mathrm{Sim}}$	$\overline{\mathrm{PosErr}}$	$\mathrm{R}^2_{\mathrm{Pos}}$	$\mathrm{MAE}_{\mathrm{Area}}$	$\mathrm{R}^2_{\mathrm{Area}}$	$\mathrm{MAE}_{\mathrm{Asp}}$	$\mathrm{R}^2_{\mathrm{Asp}}$
1	30952	0.040	0.744	0.045	0.984	0.288	0.924	0.061	0.477
2	175	0.052	0.603	0.139	0.705	3.977	0.339	0.075	0.372
3	18320	0.061	0.603	0.050	0.980	0.227	0.950	0.068	0.425
4	41292	0.044	0.704	0.031	0.987	0.155	0.849	0.052	0.258
5	7633	0.100	0.530	0.091	0.838	2.396	0.500	0.075	0.315
aggregated	98372	0.050	0.705	0.044	0.973	0.391	0.649	0.060	0.453

4. Discussion

Generally speaking, the following is true for all spot descriptors considered in this paper, with the exception of similarity: to accurately describe a Laue spot as seen in the image data, the underlying Gaussian function has to fit the image data reasonably well. Otherwise, descriptors assume almost arbitrary values. It is thus unsurprising that by removing bad quality spots from Table 1 the errors go down as shown in Table 3.

However, the opposite is true for similarity, which is already predicted well on the original data. Here, the coefficient of determination $\mathrm{R}^2_{\mathrm{Sim}}$ decreases if only the good quality spots are retained. The reason for this is that spots which are obviously badly fitted (i.e., the "easy cases") are excluded, leaving only the hard cases whose precise value of similarity is more difficult to predict. But this effect is of little importance for practical applications of the presented approach, which can be seen by the small numbers of misclassified spots, see Table 2. Only 1.3% of spots are bad quality spots that are misclassified as good ones.

Moreover, the 2.1% of spots that are confused the other way around usually have even less impact in practice, as there are still enough spots for an accurate peak analysis.

Considering only good quality spots, the peak position is predicted very well, where $\overline{\text{PosErr}} = 0.044$ and $R^2_{\text{Pos}} = 0.973$ for aggregated error scores, see Table 3. The threshold of 0.5 pixels that would nullify the peak position refinement is clearly surpassed even for the worst dataset. In contrast, the aspect ratio is the hardest descriptor to predict for the CNN models with respect to the coefficient of determination. While this descriptor and the area are both simple functions of the scale parameters σ_1 and σ_2 of the Gaussian function (see Section 2.2.3), the first one is given as a quotient of σ_1 and σ_2 in contrast to the second one, which is proportional to their product. The quotient is much more sensitive to slight deviations of σ_1 and σ_2, making the estimation less stable compared to the product where errors might cancel out. When considering the mean absolute errors MAE_{Area} and MAE_{Asp}, it is important to keep their respective range in mind: For the first one, values up to $25\pi \approx 78.5$ are possible (as a result of the outlier treatment), whereas for the latter only values up to 1 occur (as a result of the definition of Asp).

When analyzing the original distributions of similarity values for each of the five datasets considered in Figure 4, it becomes clear that they differ substantially from each other, e.g., the second dataset $\mathcal{D}_2$ shows a very skewed distribution where almost all spots are badly fitted by their Gaussian function. This explains the higher error scores shown in Table 1. As mentioned above this is caused by the ill-defined descriptors for badly fitted spots. When the bad quality spots are removed from consideration, the resulting score values are more similar to those of the other datasets. Note that in the case of dataset $\mathcal{D}_2$ only relatively few (175) spots remain, but recall that only a (random) subset of all Laue spots is considered in this evaluation (see Section 2.3.4).

One idea to improve the prediction performance of the CNNs might be to train models specifically for a single descriptor such as the similarity Sim. Then, instead of having a single model that predicts all descriptors of interest, multiple models have to be employed. To investigate this further, models have been trained with the same cross-validation strategy as stated in Section 2.3, but whose output is only Sim. The error scores listed in Table 4 show that there is no major improvement compared to the previously considered all-in-one models despite the significantly increased complexity.

Table 4. Comparison of error scores for CNN models that specifically predict the similarity Sim and for models combining all descriptors (Sim, Pos, Area, and Asp) considered in this paper. The error scores were evaluated for the CNNs on their respective test data.

model	MAE_{Sim}	R^2_{Sim}
specific	0.062	0.920
all in one	0.068	0.913

By performing cross-validation on different datasets, the generalization of the prediction performance has been quantified. However, there might be further aspects that impact the performance. While the crystal structure and orientation govern the position of the diffraction peak, the shape of the Laue spot depends on the local crystal misorientation distribution within the probed volume. The datasets considered in the present paper have been chosen to comprise a wide variety of spot shapes from simple ones, which are well-described by Gaussian functions, to very complex ones. For this reason, we assume that the predictor will work equally well on other datasets whose local crystal misorientation distribution is within the considered spectrum, irrespective of crystal structures. Nevertheless, before applying it to datasets that are outside this broad spectrum, the prediction quality should be reevaluated. Another aspect that concerns the generalization is the distance between the sample and the detector. As mentioned above the datasets considered in the present paper have been acquired with a distance between 78.5 mm and 79.5 mm. Increasing this distances leads to a homothetical transform that enlarges the Laue spots, but

leaves their general shape intact. The size of the cutouts (32 pixel × 32 pixel) that are fed to the CNN models has been chosen to work well for these values of the detector distance. However, for much larger values the Laue spots might not fully fit into the cutouts and the CNN models would thus be unable to describe them accurately. In this case, the models would need to be retrained with a larger cutout size, (e.g., 64 pixel × 64 pixel instead of 32 pixel × 32 pixel).

One of the main advantages of the CNN-based estimator presented in this paper is its high computational speed. To make this clear, the runtime performance has been evaluated by comparing it to that of the conventional gradient descent approach implemented in LaueTools, see [14]. For this purpose, 10 repetitions of analyzing 10 000 Laue spots have been conducted to determine the timings reported in Table 5.

The evaluation procedure of runtime performance consists of two setups: As the gradient descent is using only a single processor core, the CNN-based approach has also been restricted to a single core in the first setup for a fair comparison. In this case, a speedup of 2.089 ms/0.647 ms = 3.23 is obtained, see Table 5. In the second setup, the predictions of the neural network are computed on a GPU. Here, a speedup of 2.089 ms/0.027 ms = 77 is achieved. While the CNN-based approach is already significantly faster for the single-threaded setup, on the GPU it can fully leverage its inherent affinity for parallel computing, resulting in massive speedups. It is also worth mentioning that the runtime for the CNN stays the same for all inputs, whereas the gradient descent algorithm takes longer for difficult test datasets with many bad quality spots (such as $\mathcal{D}_2$). The reason for this is that for irregular spots the initial parameters based on pixel values do not already lead to a good fit and several iterations of optimization steps have to be computed. The experiments were performed on an AMD Ryzen 9 3900x CPU and an NVIDIA RTX 2080 Super GPU.

Table 5. Comparison of runtimes (in milliseconds per Laue spot) of the different approaches tested on different datasets. Note that the runtimes of the CNN-based approaches do not depend on the test datasets used.

dataset	CNN with GPU	CNN with CPU	Gradient descent with CPU
$\mathcal{D}_1$	0.027	0.647	1.655
$\mathcal{D}_2$	0.027	0.647	3.335
$\mathcal{D}_3$	0.027	0.647	1.795
$\mathcal{D}_4$	0.027	0.647	1.804
$\mathcal{D}_5$	0.027	0.647	1.857
aggregated	0.027	0.647	2.089

5. Conclusion

In this paper, we described a CNN-based method for the characterization of Laue spots (given by a pixel intensity distribution), which is an important step when analyzing Laue patterns. With the presented method, the conventional approach based on computationally expensive fitting of parametric functions can be replaced by fast CNNs. This way, a significant speedup (up to 77 times when using a GPU) is achieved for the prediction of geometric spot descriptors.

Using the CNN-based method, descriptors derived from the fitted Gaussian functions can be accurately estimated for Laue spots that are well described by these functions (good-quality spots). The remaining spots have little similarity to the parametric functions, and thus, descriptors derived from these fitted parametric functions assume almost arbitrary values. For this reason, such spots are not useful for the subsequent analysis. The CNN allows to quantify this similarity and, in this way, the usefulness of a Laue spot descriptors for the subsequent analysis.

While the currently predicted spot descriptors could be estimated using traditional methods (albeit slower), the approach proposed in the present paper can also be applied to other descriptors which are more difficult to estimate by traditional methods. For

example, there are occurrences of so-called "double peaks" which occur when two Laue spots overlap. By synthetically overlapping the grayscale images of multiple Laue spots, we could synthetically generate realistic training data of double peaks, where descriptors for both peaks are well known. This data could be used to train a neural network in order to learn descriptors of double peaks, which would allow us to further analyze Laue spots that are currently not well described by parametric functions.

Author Contributions: Conceptualization, T.K., J-S.M., L.P. and R.R.P.P.R.P.; methodology, T.K. and L.P.; software, T.K.; validation, T.K. and L.P.; formal analysis, T.K. and L.P.; investigation, J-S.M. and R.R.P.P.R.P.; data curation, R.R.P.P.R.P.; writing—original draft preparation, T.K., L.P. and R.R.P.P.R.P.; writing—review and editing, J-S.M. and V.S.; visualization, L.P.; supervision, V.S.; project administration, J-S.M. and V.S.; funding acquisition, J-S.M. and V.S. All authors have read and agreed to the published version of the manuscript.

Funding: The authors gratefully acknowledge financial support by ANR and DFG in the framework of the French–German project "HoTMiX" (project no. ANR-19-CE09-0035-01 and SCHM 997/37-1).

Data Availability Statement: Datasets of Laue patterns were collected on CRG-IF BM32 beamline at the ESRF, Grenoble, France. They are stored at ESRF and can be publicly accessed on demand and with permission according to the FAIR data policy (including potentially a 3 years embargo), see https://www.esrf.fr/fr/home/UsersAndScience/UserGuide/esrf-data-policy.html.
The latest version of the source code and the CNN model weights are available upon reasonable request to the authors.

Acknowledgments: The authors acknowledge M. Velazquez and V.S. Paul Raj for the synthesis of ZnCuOCl and the provision of the corresponding Laue patterns dataset.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results

References

1. Chung, J.S.; Ice, G.E. Automated indexing for texture and strain measurement with broad-bandpass X-ray microbeams. *Journal of Applied Physics* **1999**, *86*, 5249–5255.
2. Ice, G.E.; Pang, J.W.L. Tutorial on X-ray microLaue diffraction. *Materials Characterization* **2009**, *60*, 1191–1201.
3. Barabash, R.; Ice, G.E.; Larson, B.C.; Pharr, G.M.; Chung, K.S.; Yang, W. White microbeam diffraction from distorted crystals. *Applied Physics Letters* **2001**, *79*, 749–751.
4. Plancher, E.; Petit, J.; Maurice, C.; Favier, V.; Saintoyant, L.; Loisnard, D.; Rupin, N.; Marijon, J.B.; Ulrich, O.; Bornert, M.; et al. On the accuracy of elastic strain field measurements by Laue microdiffraction and high-resolution EBSD: a cross-validation experiment. *Experimental Mechanics* **2015**, *56*, 483–492.
5. Zhang, F.G.; Bornert, M.; Petit, J.; Castelnau, O. Accuracy of stress measurement by Laue micro-diffraction (Laue-DIC method): the influence of image noise, calibration errors and spot number. *Journal of Synchrotron Radiation* **2017**, *24*, 802–817.
6. Song, Y.; Tamura, N.; Zhang, C.; Karami, M.; Chen, X. Data-driven approach for synchrotron X-ray Laue microdiffraction scan analysis. *Acta Crystallographica* **2019**, *75*, 876–888.
7. Purohit, R.R.P.P.R.; Tardif, S.; Castelnau, O.; Eymery, J.; Guinebretière, R.; Robach, O.; Ors, T.; Micha, J.S. LaueNN: neural-network-based hkl recognition of Laue spots and its application to polycrystalline materials. *Journal of Applied Crystallography* **2022**, *55*, 737–750.
8. Wright, J.P. ImageD11. https://github.com/FABLE-3DXRD/ImageD11, 2005. Accessed: 2023-02-03.
9. Henningsson, N.A.; Hall, S.A.; Wright, J.P.; Hektor, J. Reconstructing intragranular strain fields in polycrystalline materials from scanning 3DXRD data. *Journal of Applied Crystallography* **2020**, *53*, 314–325.
10. Burger, W.; Burge, M.J. *Digital Image Processing*; Springer, 2016.
11. Sullivan, B.; Archibald, R.; Azadmanesh, J.; Vandavasi, V.G.; Langan, P.S.; Coates, L.; Lynch, V.; Langan, P. BraggNet: integrating Bragg peaks using neural networks. *Journal of Applied Crystallography* **2019**, *52*, 854–863.
12. Boyce, D.E.; Bernier, J.V. heXRD: modular, open source software for the analysis of high energy X-ray diffraction data. https://www.osti.gov/biblio/1062217, 2013. Accessed: 2023-02-03.
13. Tamura, N. XMAS: a versatile tool for analyzing synchrotron X-ray microdiffraction data. In *Strain and Dislocation Gradients from Diffraction*; Barabash, R.; Ice, G.E., Eds.; Imperial College Press, 2014; pp. 125–155.
14. Micha, J.S.; Robach, O.; Tardif, S. LaueTools: open source Python packages for X-ray microLaue diffraction analysis. https://github.com/BM32ESRF/lauetools, 2015. Accessed: 2023-02-03.

15. Ulrich, O.; Biquard, X.; Bleuet, P.; Geaymond, O.; Gergaud, P.; Micha, J.S.; Robach, O.; Rieutord, F. A new white beam X-ray microdiffraction setup on the BM32 beamline at the European Synchrotron Radiation Facility. *Review of Scientific Instruments* **2011**, *82*, 033908.
16. Rabinovich, D.; Lourie, B. Use of the polychromatic Laue method for short-exposure X-ray diffraction data acquisition. *Acta Crystallographica Section A* **1987**, *43*, 774–780.
17. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*, 2nd ed.; Springer, 2009.
18. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An Introduction to Statistical Learning*, 2nd ed.; Springer, 2021.
19. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press, 2016.
20. Kingma, D.P.; Ba, J. Adam: a method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015; Bengio, Y.; LeCun, Y., Eds., 2015.
21. Gyori, N.G.; Palombo, M.; Clark, C.A.; Zhang, H.; Alexander, D.C. Training data distribution significantly impacts the estimation of tissue microstructure with machine learning. *Magnetic Resonance in Medicine* **2022**, *87*, 932–947.
22. Chicco, D.; Warrens, M.J.; Jurman, G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science* **2021**, *7*, e623.