Advanced Intelligent Discovery

ADVANCED INTELLIGENT DISCOVERY Open Access

RESEARCH ARTICLE OPEN ACCESS

Data-Driven High-Throughput Volume Fraction Estimation From X-Ray Diffraction Patterns

Hawo H. Höfer[1] | André Orth[1] | Robert Wang[2] | Ben Breitung[2] | Simon Schweidler[2] | Jasmin Aghassi-Hagmann[2] | Markus Reischl[1]

[1]Institute for Automation and Applied Informatics, Karlsruhe Institute of Technology, Eggenstein-Leopoldshafen, Baden-Württemberg, Germany | [2]Institute of Nanotechnology, Karlsruhe Institute of Technology, Eggenstein-Leopoldshafen, Baden-Württemberg, Germany

Correspondence: Hawo H. Höfer (hawo.hoefer@kit.edu; hawohoefer@protonmail.com)

Received: 6 October 2025 | **Revised:** 6 March 2026 | **Accepted:** 10 March 2026

Keywords: high-throughput processing | machine learning | quantitative analysis | X-ray diffraction

ABSTRACT

In high-throughput applications, crystallographic analysis via X-ray diffraction (XRD) is often limited by long exposure times and the need for manual data interpretation. This study presents a novel machine-learning-based approach for volume fraction estimation from XRD patterns addressing both challenges. The method efficiently processes noisy XRD patterns acquired with polychromatic emission spectra, and enables volume fraction estimation from thousands of patterns per second. Compared to conventional techniques, it requires much lower XRD pattern quality, allowing for shorter exposure times. This makes our method particularly suited for high-throughput scenarios such as self-driving labs. We utilize simulated XRD patterns to train two neural networks to process XRD data in a specified material system. A convolutional neural network (CNN) estimates volume fractions, while a u-net-style network restores pattern interpretability by resolving peak duplication caused by polychromatic emission spectra. We use synthetic datasets to showcase the method's noise tolerance and ability to analyze XRD patterns with multiple emission lines. Furthermore, we verify our method's applicability to real XRD patterns using a small experimental dataset.

1 | Introduction

X-ray diffraction (XRD) is a widely used technique for analyzing crystalline materials, providing insights into sample composition and crystallographic structure. It plays a crucial role for self-driving laboratories and combinatorial material exploration. XRD typically requires parallel monochromatic radiation, which is obtained by filtering a polychromatic source with subsequent optics. The resulting X-rays generate diffraction patterns upon interaction with the sample. For accurate analysis, low-noise XRD patterns are essential, which traditionally necessitates long exposure times and/or high radiation intensities. However, these factors, along with the need for manual pattern evaluation, often create bottlenecks in research workflows.

Other applications of high-throughput experiments have benefited from analysis using data-driven approaches [1–4], and XRD analysis is no exception. Conventionally, XRD patterns are analyzed manually by Rietveld methods [5], a process that can be labor-intensive depending on pattern complexity and quality. To reduce the need for eliminate human intervention, several approaches have been explored:

- phase mapping in ternary phase diagrams using hierarchical and fuzzy clustering [6–12],
- CNN-based classifiers for impurity detection [13], phase identification [14–20], space/extinction group analysis [21–23],

- composition analysis leveraging fuzzy clustering, optimization and neural networks [24–28], and
- U-net style networks denoising XRD patterns, either as standalone methods or preprocessing for classification [22, 29].

Three main groups of approaches to solving XRD composition analysis can be identified:

- optimization based approaches [5, 24] iteratively refine input parameters to approximate XRD patterns,
- classical machine learning (ML) methods (e.g., fuzzy-c-means clustering, support vector machines) perform well, particularly with limited experimental data [25, 26], and
- large-scale neural networks or network systems trained on large synthetic datasets containing up to millions of XRD patterns [18, 27].

Classical methods often outperform convolutional or long-short-term-memory neural networks on small datasets [25, 26]. Similarly, dispatch-based methods [27] show improved performance when combined with support vector regression rather than fully connected single-hidden-layer neural networks. While identification-stripping has proven effective for phase identification, it remains unreliable for composition analysis [18]. Neural networks can achieve performance comparable to other methods trained on experimental data when relying on fancy-PCA-based data augmentation [28].

Analyzing compositions in more complex systems demands models with higher parameter counts and/or sophisticated usage schemes [18, 27], alongside extensive training data. Since large experimental datasets are scarce, synthetic XRD datasets generated from crystal structure definitions have become standard for training. Synthetic XRD patterns are typically created through advanced simulation pipelines [13, 17, 18, 22, 27]. For example, multiphase XRD patterns are simulated by superposing single-phase simulations with Gaussian noise, random 3rd- to 5th-order Chebyshev backgrounds and air scattering effects [17].

As illustrated by the presented literature, machine learning can be used for classification and regression on XRD patterns. However, relatively few contributions focus on the robustness of automated composition analysis with respect to polychromatic radiation and noise. For classification using vector machines, multiple papers found a strong effect of the noise level on clustering accuracy [19, 20]. Little to no effect could be shown for hierarchical clustering methods [8]. To the best of our knowledge, no research has been done on the effect of polychromatic radiation or noise level on the precision of XRD composition analysis using machine learning methods.

1.1 | Contributions

We provide **HiVE**, a method for **Hi**gh-throughput **V**olume fraction **E**stimation from XRD patterns. All training and data generation code, including our simulation tool **YAXS** (**Y**AXS: an **A**ccelerated **X**RD **S**imulator) is published along with this paper.[1]

Furthermore, insights are presented on neural network based XRD analysis which enable the use of XRD in high-throughput experiments.

- We show that our **Com**position **a**nalysis **Net**work (**ComaNet**) performs similarly when analyzing and low noise XRD patterns recorded using monochromated radiation (Cu$K\alpha_{1/2}$) compared to XRD patterns recorded without monochromators (Cu$K\alpha_{1/2}/\beta$) and high noise.
- We thoroughly describe a method to generate synthetic datasets of multiphase XRD patterns generated using polychromatic emission spectra.
- We show the novel method can reduce exposure times and remove the need for monochromators while retaining prediction quality.
- We evaluate our method on a small experimental dataset created specifically to contrast model performance on Cu$K\alpha_{1/2}$ and Cu$K\alpha_{1/2}/\beta$-radiation.

HiVE enables the use of XRD in high-throughput scenarios and automated systems by removing the need for human intervention and allowing XRD patterns to be recorded in with much lower exposure times. XRD patterns at these lower qualities are often hard to understand. For cases where manual inspection is required, we provide **PatraNet** (**Pa**ttern **tra**nslation **Net**work) to restore interpretability by filtering noise and removing peaks caused by removing monochromators.

ComaNet is validated using a synthetic dataset of XRD patterns of binary mixtures of CuO and Fe_3O_4, enabling easy variation of noise levels and emission spectra. Performance is compared regarding XRD patterns produced using monochromated and non-monochromated radiation at different noise levels. Additionally, we analyze **PatraNet** using synthetic data and the aforementioned experimental dataset in the CuO-Fe_3O_4 composition space and show that translation of XRD patterns between emission spectra is possible.

2 | Methods

The first part of this section outlines the proposed method in general. Next, the synthetic data generation is described. Finally, the two network architectures and their training and evaluation schemes are laid out.

2.1 | Overview

HiVE is a method for performing **Hi**gh-throughput **V**olume fraction **E**stimation for XRD patterns where the possible constituent phases (phases of interest) are known, and no other phases are present in significant quantities. While **HiVE** can handle small amounts of unknown impurities, larger amounts of unknown phases will skew the volume fraction estimation. Use of **HiVE** requires the following steps:

1. From the phases of interest, synthetic datasets are generated using our simulation tool **YAXS**.

2. **ComaNet** and **PatraNet** are trained using these synthetic datasets.
3. Model training is validated using synthetic data.
4. Data generation is verified by evaluating performance using experimental examples.
5. Models are applied to real-world data.

Due to the small model size, and efficiency of data generation tooling, the first two steps usually take on the order of minutes. Models are usable and trainable on consumer-grade hardware, albeit with longer training times. Due to this lightweight nature of our models, specialized models can be trained for a novel material system to be analyzed. Thus, the combinatorial explosion occurring when models need to be trained for processing arbitrary XRD patterns can be avoided.

The application of **HiVE** to real-world data is described in Figure 1. Traditionally, X-rays are monochromated either before or after interaction with the sample to remove secondary peaks, reducing radiation intensity. High-quality XRD patterns for composition analysis therefore require long exposure times. In contrast, our workflow relies on **ComaNet** for composition analysis. It analyzes thousands of XRD patterns per second without the need for human intervention, and can tolerate XRD patterns with secondary peaks (caused by multiple emission lines) and extensive amounts of noise. Therefore, monochromators can be omitted from XRD instruments, and exposure times can be reduced. However, these XRD patterns may not be visually interpretable. If visual inspection of patterns is required, **PatraNet** can be used to restore interpretability by filtering noise and removing peaks caused by secondary emission lines.

2.2 | Synthetic Dataset Generation

Neural networks require a large number of training data—examples consisting of inputs into the network and the corresponding desired outputs. Composition estimation networks therefore require hundreds of thousands of XRD patterns and matching sample compositions. Since these data are not available, we generate synthetic datasets for neural network training through simulation. Our synthetic datasets consist of XRD patterns of the phases of interest, and potential output variables like the composition, or properties of the modeled phases.

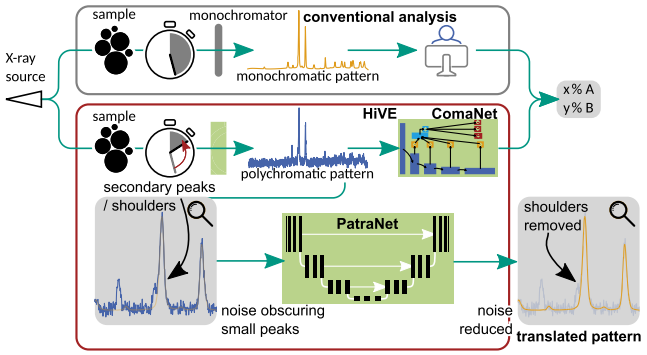

FIGURE 1 | Conventional (top, gray box) versus automated, machine learning-driven (bottom, red box) XRD analysis. Novel elements are marked in green. Instead of manual phase identification, and Rietveld analysis of high-quality patterns, we rely on a neural network (ComaNet) to analyze low-quality patterns recorded with lower exposure times and secondary peaks. Interpretability is restored to low-quality patterns ("translating" them) using a u-net style network (PatraNet) to remove secondary peaks and filter noise.

Figure 2 guides the following mathematical description of our XRD pattern simulation. Individual XRD pattern I (see Figure 2A) are constructed from the raw pattern I_{raw}, the background signal $\mathcal{B}$ and Gaussian noise ξ. We simulate Backgrounds using random 10th-order Chebyshev polynomials with coefficients sampled uniformly from $\mathcal{U}(0,1)$

$$\begin{aligned} I(2\theta) &= I_{\text{raw}}(2\theta) + \mathcal{B}(2\theta) + \xi \\ \text{with } \xi &\sim \mathcal{N}(0,\sigma) \\ \sigma &\sim \mathcal{U}(0,\sigma_{\max}) \end{aligned} \tag{1}$$

Background values are mapped to the interval between 0 and the maximum background height $h_{\mathcal{B}}$

$$\mathcal{B}(2\theta) \in [0, h_{\mathcal{B}}] \,\forall\, 2\theta \in [2\theta_{\min}, 2\theta_{\max}] \tag{2}$$

The raw pattern I_{raw} is a sum of single-emission-line patterns I_{λ_i} weighted with the emission line's strength f_i relative to the other emission lines in the emission spectrum.

To simulate phases not present in training data (phases to be ignored for quantification), we add an impurity pattern to the raw pattern. This allows models trained on a generated dataset to handle nonspecified phases. Single emission line patterns I_{λ_i} are composed of the impurity pattern I_{imp} and the constituent phase c's patterns I_{c} weighted according to their volume fractions φ_{c} (Figure 2B)

$$\begin{aligned} I_{\text{raw}}(2\theta) &= \sum_i f_i I_{\lambda_i}(2\theta) \\ I_{\lambda_i}(2\theta) &= I_{\text{imp}}(2\theta, \lambda_i) + \sum_c \varphi_c I_c(2\theta, \lambda_i) \end{aligned} \tag{3}$$

For generation of our datasets, we utilized a 1:7 mixture of single-phase and multiphase samples. We chose this instead of uniformly sampling the composition space in order to increase performance on single-phase patterns.

Each single phase pattern I_{c} is a sum of Pseudo-Voigt (V_{p}) peaks with simulated peak positions $2\theta^{\lambda}_{\text{hkl,c}}$ and intensities $I^{\lambda}_{\text{hkl,c}}$. We

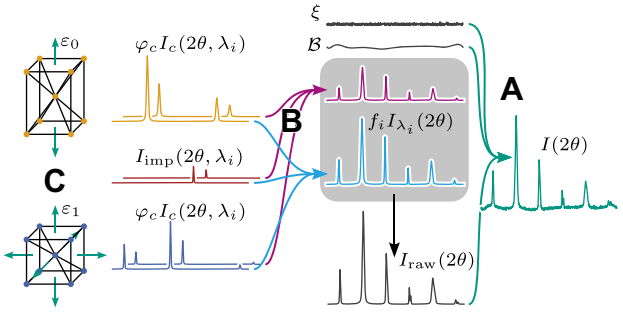

FIGURE 2 | Schematic of the data generation process. Per phase and wavelength XRD patterns $I_{(\cdot)}$ are combined with impurity patterns, noise ξ, and a background $\mathcal{B}$ to produce the final simulated XRD pattern $I(2\theta)$.

emphasize that these peak positions and intensities depend on the phase c and the emission line's wavelength λ using the sub- and superscripts λ and c. The subscript hkl indicates the peak's crystallographic plane. Peak widths are computed using the size-broadening method implemented in GSAS-II [30] and randomly sampled Caglioti parameters U, V and W for Gaussian line broadening. Models trained on these datasets should be able to handle XRD patterns from a variety of devices and with varying grain sizes.

The impurity pattern I_{imp} is constructed from $n \in \{0 \dots 2\}$ impurity peaks with uniformly random heights $I_{\text{imp},j}$ and positions $2\theta_j^\lambda$

$$\begin{aligned} I_{\text{imp}}(2\theta,\lambda) &= \textstyle\sum_{j=1}^{n} I_{\text{imp},j} V_{\text{p}}(2\theta_j^\lambda, 2\theta) \\ I_c(2\theta,\lambda) &= \textstyle\sum_{\text{hkl}} I_{\text{hkl},c}^\lambda V_{\text{p}}(2\theta_{\text{hkl},c}^\lambda, 2\theta) \end{aligned} \tag{4}$$

The impurity peak positions and intensities ($I_{\text{imp},j}$) are uniformly sampled from the used 2θ-interval, and the interval from 0 to a maximum impurity height h_{imp}

$$\begin{aligned} 2\theta_{\text{imp},j} &\sim \mathcal{U}(2\theta_{\min}, 2\theta_{\max}) \\ I_{\text{imp},j} &\sim \mathcal{U}(0, h_{\text{imp}}) \end{aligned} \tag{5}$$

Peak positions $2\theta_{\text{hkl,c}}^\lambda$ and intensities $I_{\text{hkl,c}}^\lambda$ for single-phase patterns are simulated from CIFs using the algorithms implemented in *pymatgen* [31]. To improve generalizability of models trained on the dataset, we emulate external and internal forces by straining the unit cell. Strain conditions ε_{c} are applied to each phase's (c) unit cell (Figure 2C). These strain conditions are constructed with a maximum amplitude of ε_{c}, and preserve the phase's symmetry group [18]. Furthermore, peak position shift is applied according to sample displacement (SD) and goniometer radius (R), to capture one of the largest sources of error in XRD analysis [32]

$$\Delta 2\theta(2\theta) = 2.0\,\text{SD}/R\cos\,(2\theta/2) \tag{6}$$

Additionally, *pymatgen*'s algorithms are modified to account for unit cell volume V and wavelength λ [33]

$$I_{\text{hkl,c}}^\lambda = \underbrace{|F_{\text{hkl}}|^2 p \overbrace{\frac{1+\cos\,(2\theta)^2}{\sin\,(\theta)^2\cos\,(\theta)}}^{F_{\text{L}}}}_{\text{pymatgen}} \underbrace{\frac{\lambda^3}{V^2}}_{\text{adjust}} \tag{7}$$

Here F_{hkl} is the structure factor and p is the multiplicity of the peak. The following fraction is the Lorentz polarization factor (labeled F_{L}).

Datasets are saved before the noise ξ is added, and it is applied during training. This enables easy variation of noise levels between training runs and during training. After noise application, patterns are normalized to $[0, 1]$ using their own minimum and maximum.

2.2.1 | Parameter Values

The noise level in an XRD pattern does not depend on the height of the maximum peak. Similarly, the height of peaks associated with potential impurities does not depend on the maximum peak height of the clean pattern, either. We therefore need to add noise and impurity peaks before normalizing patterns. This implies that we need absolute values for the maximum noise level $\sigma_{\max}$, maximum impurity peak height h_{imp} and background height $h_{\mathcal{B}}$ in simulation. These values are purely artifacts of the simulation, and their absolute value is irrelevant for the discussion. Thus, any numeric values are shown as fractions of a reference height h_{ref}, which we determine from a 100% CuO pattern with arbitrarily selected simulation parameters. For it's definition please refer to the supporting information. A summary of all simulation parameters can also be found in the supporting information (Table S2).

The highest and lowest noise levels $\sigma = \{2^{-6}h_{\text{ref}}; 2^{-2}h_{\text{ref}}\}$ are shown in Figure 3. These limits were chosen arbitrarily, such that the lower limit represents clean XRD patterns, while the upper limit represents incredibly short exposure times where peaks are barely discernible by eye.

2.2.2 | Volume Fraction Estimation

ComaNet performs composition analysis for our method, and requires a dataset of XRD patterns and matching compositions for training. This study analyzes the effect of emission spectrum on composition analysis prediction quality. Datasets of XRD patterns are therefore simulated for each of the emission spectra shown in Table 1. For each emission spectrum, instances of **ComaNet** are trained and evaluated. In $\mathcal{C}_{\alpha_1}$, the nonphysical monochromatic reference spectrum, only the Copper K-α_1 emission line is used. The $\mathcal{C}_{\alpha_{1/2}}$ emission spectrum is representative of many laboratory Copper XRD instruments. We add the Cu$K\beta$-emission line in $\mathcal{C}_{\alpha_{1/2},\beta}$ to simulate removing the instrument's monochromator. $\mathcal{C}_3$ and $\mathcal{C}_4$ are constructed to be similar to the $\mathcal{C}_{\alpha_{1/2}}$ spectrum, but with nonphysical emission lines e_3 and e_4 close to the two standard lines. They are used to investigate potential differences between data generated using spectra containing similar emission lines compared to ones that differ more. The positions of the utilized emission lines illustrated by Figure 5, where they are shown overlaid with a schematic of the Copper emission spectrum.

Synthetic XRD patterns for a single composition with different emission spectra can be found in Figure 4. XRD patterns

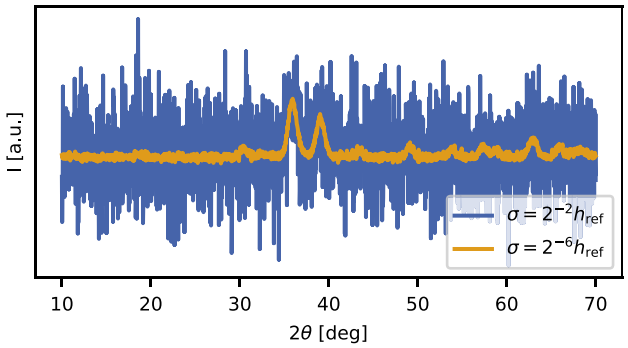

FIGURE 3 | Two non-normalized XRD patterns with varying σ and composition. Peak heights depend on composition and peak visibility depends on σ.

TABLE 1 | Emission spectra $\mathcal{C}_i$ (rows) and their constituent emission lines. Weights and emission energies can be found in the supporting information (Table S3).

	$\mathbf{CuK\alpha_1}$	$\mathbf{CuK\alpha_2}$	$\mathbf{CuK\beta}$	$\boldsymbol{e_3}$	$\boldsymbol{e_4}$
$\mathcal{C}_{\alpha_1}$	✓	—	—	—	—
$\mathcal{C}_{\alpha_{1/2}}$	✓	✓	—	—	—
$\mathcal{C}_{\alpha_{1/2},\beta}$	✓	✓	✓	—	—
$\mathcal{C}_3$	✓	✓	—	✓	—
$\mathcal{C}_4$	✓	✓	—	✓	✓

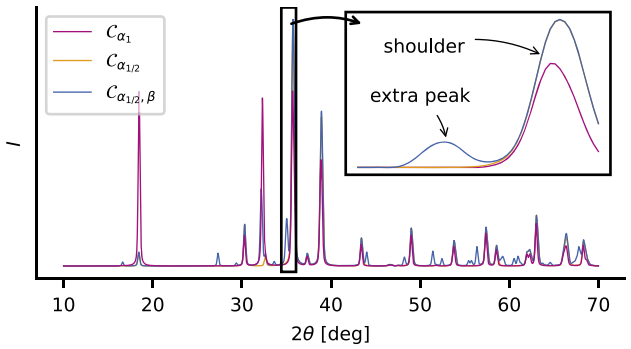

FIGURE 4 | Comparison of synthetic XRD patterns for the same mixture for three emission spectra. Peak count and shape depends on the used emission lines.

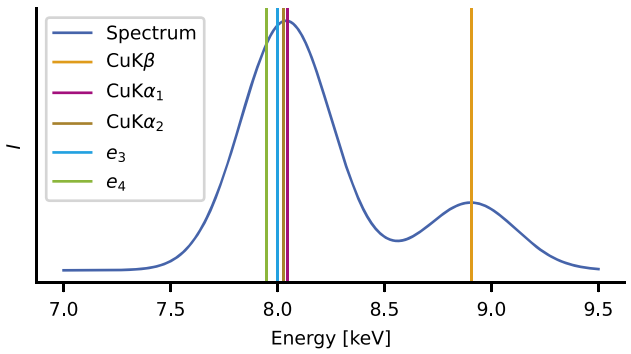

FIGURE 5 | Copper emission spectrum [34] (schematic) and the emission lines used in this article.

generated using the $\mathcal{C}_{\alpha_{1/2},\beta}$ spectrum differ slightly in peak shape from the $\mathcal{C}_{\alpha_1}$ and $\mathcal{C}_{\alpha_{1/2}}$ spectra. The Cu$K\alpha_1$ and Cu$K\alpha_2$-peaks overlap and form shoulders at some 2θ-positions. In the $\mathcal{C}_{\alpha_{1/2},\beta}$-pattern, Cu$K\beta$-peaks sometimes overlap with Cu$K\alpha_{1/2}$-doublets, causing strong shouldering. There is overlap between peaks, but especially the $\mathcal{C}_{\alpha_{1/2},\beta}$ spectrum contains more stronger shouldering effects, and more peaks overlap compared to the $\mathcal{C}_{\alpha_1}$-pattern. Peaks with overlap exhibit shoulders, and may separate into two peaks associated with the same cyrstallographic plane at high 2θ - depending on the Caglioti parameters u, v, and w and the domain size τ.

The volume fraction estimation dataset $\mathcal{D}_\text{v}$ is a collection of input XRD patterns $I(2\theta)$ and outputs $(\boldsymbol{\varphi}, Y_\text{sec})$. It's outputs are comprised of the volume fractions $\boldsymbol{\varphi}$ and the secondary parameters Y_sec used to construct the XRD pattern

$$\begin{aligned}&\mathcal{D}_\text{v} = \{[I(2\theta), (\boldsymbol{\varphi}, Y_\text{sec})]_i \mid i \in \{1 \dots 2 \cdot 10^6\}\}\\&\text{with } I(2\theta) = f(\boldsymbol{\varphi}, Y_\text{sec})\\&\boldsymbol{\varphi} \in [0,1]^C \text{ and } \sum_{c=1}^{C} \varphi_c = 1\end{aligned} \tag{8}$$

Here f is the function creating an XRD pattern $I(2\theta)$ using $\boldsymbol{\varphi}$ and Y_sec via the process described above. The subscript c indicates the value of the quantity for component c, and C is the number of components.

The secondary parameters are:

- domain size $\boldsymbol{\tau} \in \mathbb{R}^{C\times 1}$,
- domain size broadening mixing parameter $\boldsymbol{\eta} \in \mathbb{R}^{C\times 1}$,
- left lower half of the strain matrices $\boldsymbol{\varepsilon} \in \mathbb{R}^{C\times 6}$ applied to the unit cells,
- Caglioti parameters $(u, v, w \in \mathbb{R})$, and
- sample displacement SD $\in \mathbb{R}$.
- background scale $h_\mathcal{B} \in \mathbb{R}$
- Chebyshev coefficients $\boldsymbol{c} \in \mathbb{R}^{11}$

Domain size $\boldsymbol{\tau}$ and strain $\boldsymbol{\varepsilon}$ are physical characteristics and specific to each phase, due to for example different crystallization properties and Young's moduli, respectively. $\boldsymbol{\eta}$ is modeled separately for each phase, like in contemporary Rietveld analysis software like GSAS-II [30]. Caglioti parameters model instrument characteristics and therefore apply to all components equally. Similarly, SD is equal for all components because we expect even mixing of the sample.

Y_sec is thus defined as

$$Y_\text{sec} = (*\boldsymbol{\tau},\ *\boldsymbol{\eta},\ *\boldsymbol{\varepsilon}, u, v, w, \text{SD}, h_\mathcal{B},\ *\boldsymbol{c})^\text{T} \in \mathbb{R}^{8C+16} \tag{9}$$

$(\cdot)_\text{c}$ indicates per-component variables, and C is the number of components, and $*(\cdot)$ indicates flattening and expansion of the argument $*(\cdot): \mathbb{R}^{M\times N} \to \mathbb{R}^{N\cdot M}$ (example in Supporting Information, Equation S(6)).

To simulate the peaks $(I^\lambda_{\text{hkl,c}}, 2\theta^\lambda_{\text{hkl,c}})$, we apply 10 000 strain conditions $\boldsymbol{\varepsilon}_\text{c}$ to each of the component's unit cells. For each emission spectrum, we simulate 2 million XRD patterns. For every pattern, a random strained unit cell is picked, and the secondary parameters and background's Chebyshev coefficients are sampled. The resulting patterns are split into training, validation and test datasets with ratios {0.6, 0.2, 0.2}.

2.2.3 | XRD Pattern Translation

XRD patterns produced using polychromatic emission spectra and short exposure time contain secondary peaks and large amounts of noise. Because of these characteristics, they may not be interpretable. **PatraNet** is introduced to translate patterns lacking interpretability to conventionally recorded XRD patterns by removing secondary peaks and noise.

To train **PatraNet** for removing peaks introduced by secondary emission lines, we designed the (synthetic) translation dataset $\mathcal{D}_t$. It is constructed similarly to the volume fraction estimation dataset: XRD patterns are generated using the method described in subsection 2.2, but both inputs I_{in} to the models and their desired outputs I_{out} are XRD patterns. Input patterns are generated with the $\mathcal{C}_{\alpha_{1/2},\beta}$ emission spectrum, and output patterns use the $\mathcal{C}_{\alpha_{1/2}}$ emission spectrum

$$\mathcal{D}_t = \{[I_{in}(2\theta), I_{out}(2\theta)]_i \mid i \in \{1 \dots 2 \cdot 10^6\}\} \tag{10}$$

For each input–output pair, all construction parameters and compositions are identical except for the emission spectra used to compute peak positions. As inputs, we use patterns created with the $\mathcal{C}_{\alpha_{1/2},\beta}$-emission lines (weights [35] in Table S3), and the outputs are simulated using $\mathcal{C}_{\alpha_{1/2}}$. XRD patterns are normalized slightly differently compared to $\mathcal{D}_v$. Input patterns $\tilde{I}_{in}$ are normalized as usual, but the output patterns $\tilde{I}_{out}$ are normalized using the input's minimum and maximum

$$\begin{aligned} I_{in}(2\theta) &= \frac{\tilde{I}_{in}(2\theta) - \min_{2\theta}\tilde{I}_{in}}{\max_{2\theta}\tilde{I}_{in} - \min_{2\theta}\tilde{I}_{in}} \\ I_{out}(2\theta) &= \frac{\tilde{I}_{out}(2\theta) - \min_{2\theta}\tilde{I}_{in}}{\max_{2\theta}\tilde{I}_{in} - \min_{2\theta}\tilde{I}_{in}} \end{aligned} \tag{11}$$

This eliminates intensity shift between the XRD patterns' baselines and improves visualization. $\tilde{(\cdot)}$ indicates a non-normalized quvaeantity. An example for inputs and outputs (without noise and background) of $\mathcal{D}_t$ is displayed in Figure 4. The inputs and outputs are labeled with $\mathcal{C}_{\alpha_{1/2},\beta}$ and $\mathcal{C}_{\alpha_{1/2}}$, respectively.

2.2.4 | Implementation

The XRD simulation process described above is implemented in our simulation tool **YAXS**. It is available in our GitHub repository.[2] YAXS is a tool for reproducibly generating distributions of XRD patterns according user-defined parameters specified in YAML-files. These configuration files can conveniently be stored in version control systems, and enable versioning of training data without saving the data itself. YAXS utilizes multithreaded algorithms for computing peak position, intensities and widths, and then renders them using GPU-accelerated methods. Fully exploiting modern hardware capabilities, it is capable of simulating 2 million XRD patterns of binary mixtures (13 GB, the size of one training dataset used in this work) in just over 30 s, reaching speeds of 190 MB s^{-1} on our hardware. This is approximately the same speed as creating a copy of the dataset.

2.3 | ComaNet

To estimate the components' volume fractions, we developed the CNN-based **ComaNet** (**Com**position **a**nalysis **Net**work) architecture. Each instance of **ComaNet** is trained for a particular combination of materials and 2θ-range. It predicts volume fractions and the secondary parameters Y_{sec} described in Equation (9). This regularizes the training, and Y_{sec} can be used to recover a version of the XRD pattern without background, noise and impurities.

2.3.1 | Model Architecture

A schematic of the model architecture can be found in Figure 6. The input XRD pattern (as 2048 steps in 2θ) is passed through multiple convolutional layers d_i. Intermediate features are extracted and passed though reduction layers r_i to adjust their sizes to similar dimensions. To reduce overfitting, 50% dropout is applied after each reduction layer. The resulting features are concatenated and fed into two dense layers, which predict sample composition φ_i and the secondary parameters Y_{sec}. The exact model definition can be found in the training repository.[3] In total, **ComaNet** has 53k parameters.

2.3.2 | Training

ComaNet is trained on the volume fraction analysis dataset $\mathcal{D}_v$ using a two-part loss: The composition estimation loss $\mathcal{L}_\varphi$ and a loss for the secondary parameters $\mathcal{L}_{sec}$

$$\mathcal{L}_{ComaNet} = (1-\alpha)\mathcal{L}_\varphi + \alpha\mathcal{L}_{sec} \tag{12}$$

$$\mathcal{L}_\varphi = \mathrm{TV}(\hat{\varphi}, \varphi) = \frac{1}{2}\sum_{i=1}^{C}|\hat{\varphi}_i - \varphi_i| \tag{13}$$

$$\mathcal{L}_{sec} = \mathrm{MSE}(\hat{Y}_{sec}, Y_{sec}) = \sum_i (\hat{Y}_{sec,i} - Y_{sec,i})^2 \tag{14}$$

The two components are weighted using the weighing factor α which can be set as a hyperparameter. Since all secondary outputs are normalized to $[0, 1]$, the corresponding secondary loss for each component is bounded between 0.0 and 1.0. We used $\alpha = 0.05$. At 2 components, our secondary loss is therefore bounded by 32, and our choice for α weighs it a little more important than the volume fraction loss. $\mathcal{L}_\varphi$ is responsible for learning volume fractions of the constituent phases. We choose the total variation distance TV $\in [0, 1]$, which describes the distance between two vectors that sum to 1 (probability distributions). It accurately describes the quantity to be minimized, and can be intuitively interpreted as the fraction of physical sample that was not identified correctly. Furthermore, it is bounded between 0 and 1, allowing for easy interpretability and comparison of predictions across models.

The secondary loss $\mathcal{L}_{sec}$ serves to regularize the training and allows for easier debugging of the model's predictions. The

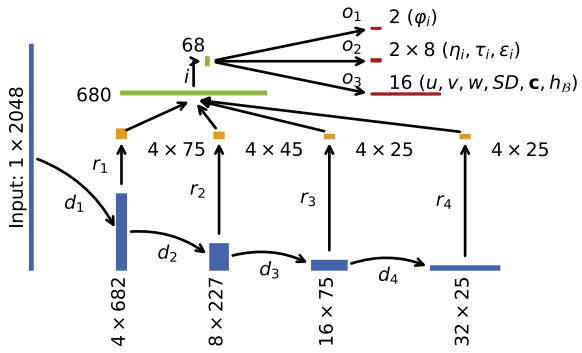

FIGURE 6 | **ComaNet** model schematic with data sizes after each layer. Convolutional layers are marked by d_i, fully connected layers are marked by r_i, i, and o_i.

secondary outputs Y_{sec} can be used to reconstruct the input XRD pattern (after a background fit). This reconstruction can be used to verify that the corresponding prediction is plausible.

The **ComaNet** architecture is implemented using *PyTorch* [36], and trained using the *ADAM* optimizer. We utilize a learning rate of 1×10^{-3}, and *PyTorch*'s ReduceLROnPlateau to schedule learning rates during training. One training run of 20 epochs takes approximately 14 min on an NVIDIA RTX6000 Ada 48 GB graphics card.

2.3.3 | Metrics

We measure the performance of **ComaNet** using a quality metric ($\mathcal{Q}$) derived from the total variation distance (TV, see Equation (13)) between the true volume fractions φ and prediction $\widehat{\varphi}$

$$\begin{aligned} \mathcal{Q} &= 1 - \mathrm{TV}(\widehat{\varphi}_c, \varphi_c) \\ \mathcal{Q} &\in [0,1], \quad \varphi_c \in [0,1], \quad \sum_{c=1}^{C} \varphi_c = 1 \end{aligned} \tag{15}$$

This quality metric is 1 if the prediction matches the ground truth. It becomes zero when the model predicts the opposite of the ground truth, such that

$$\widehat{\varphi}_c \begin{cases} = 0, & \varphi_c \neq 0 \\ \geq 0, & \varphi_c = 0 \end{cases}$$

2.4 | PatraNet

PatraNet performs XRD pattern denoising and removal of peaks produced by secondary emission lines.

2.4.1 | Architecture

We chose a u-net-style [37] architecture, because it has performed well in the context of image denoising [38] and domain transfer [39], and because it has been employed in similar cases in the context of XRD processing [22, 29]. The original u-net architecture is modified to to work on 1D-inputs length 2048 (steps in 2θ). We slim down the architecture by reducing the convolutions per depth to 1, and use 4 channels after the first layer instead of 64. We use a depth of 4 and a kernel size of 3. Each depth doubles the number of channels and halves the first input dimension, as in the standard u-net. Additionally, we add a linear layer with a residual connection at the bottleneck, to allow passing of global information. In total **PatraNet** encompasses 60 k parameters.

2.4.2 | Training

PatraNet is trained on the translation dataset $\mathcal{D}_t$. We utilize 50/50 weighted MSE and MAE losses between the output pattern $\hat{I}$ and the desired output I_{out} to both heavily punish large deviations and reduce small errors

$$\mathcal{L}_{\mathbf{PatraNet}} = 0.5 \cdot \left[\mathrm{MSE}\left(\hat{I}, I_{out}\right) + \mathrm{MAE}\left(\hat{I}, I_{out}\right)\right] \tag{16}$$

For training **PatraNet**, each pattern's noise level σ_{max} is sampled uniformly from $\mathcal{U}(0, h_{ref}/4)$ Then, each pattern's noise is sampled from $\mathcal{N}(0, \sigma_{max})$. Since each pattern's maximum noise is uniformly sampled from $\mathcal{U}(0, \sigma_{max})$, the network is also trained to handle lower noise levels. **PatraNet** is also implemented using *PyTorch*, and trained using the *ADAM*-optimizer. We utilize a learning rate of 5×10^{-3}, and *PyTorch*'s ReduceLROnPlateau learning rate scheduler, as well as early stopping. One training run (20 epochs) takes approximately 15 min on an NVIDIA RTX6000 Ada 48 GB graphics card. For further details please refer to the implementation.[4]

2.4.3 | Evaluation

The performance of **PatraNet** is measured using two separate metrics. We use the MSE to judge overall output accuracy. To intuitively capture errors in peak height reproduction, we use the maximum MAE occurring each pattern as a second metric

$$M_{mean} = \mathrm{MAE}(\widehat{I}, I) = \frac{1}{N}\sum_{i}^{N}\left|\widehat{I}(2\theta_i) - I(2\theta_i)\right| \tag{17}$$

$$M_{max} = \max_{i\in\{1\ldots N\}}\left|\widehat{I}(2\theta_i) - I(2\theta_i)\right| \tag{18}$$

Here I indicates the intensities in the XRD pattern, i indicates the discretized position, and N is the number of steps. Figure 7 shows the two metrics for an example translation. The desired and actual outputs are displayed in red and orange, respectively. M_{max} is indicated by the bar, and the dark gray area indicates M_{mean}.

2.5 | Method Validation

This section contains the experiments we performed to display **HiVE**'s working principle. In the absence of experimental datasets, we validate our approach using synthetic data. We chose the binary CuO–Fe_3O_4 composition space, with varying simulation parameters. The performance of the volume fraction estimation network **ComaNet** is analyzed for various configurations of noise and emission spectrum. We also evaluate our metrics for **PatraNet** at one configuration, to show that it works in principle.

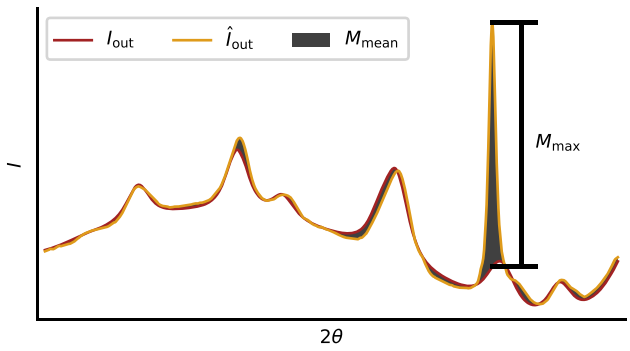

FIGURE 7 | Translation metrics. M_{mean} is influenced by the mean of the differences between input and output, while M_{max} is it's maximum.

2.5.1 | Composition Estimation

Our composition analysis network **ComaNet** estimates component volume fractions from XRD patterns, and predicts secondary parameters which can be used for reconstruction of the input pattern. It's performance is analyzed for various noise levels $\sigma_{\max}$ and emission spectra, to analyze their influence on the prediction quality $\mathcal{Q}$.

- The noise level $\sigma_{\max}$ is varied to judge the influence of exposure time (for values see Table S4).
- The emission spectrum is varied to analyze whether different X-ray sources affect prediction quality for **HiVE** (see Table 1).

Throughout all experiments, the background height was set to h_{ref}. For each emission spectrum, 2×10^6 XRD patterns are simulated. 1.2×10^6 XRD patterns are used for training, 4×10^5 are used for validation during the training, and 4×10^5 are used for the analysis in this section. Their performance is evaluated using $\mathcal{Q}$ (see Equation (15)).

Figure 8 shows box-plots for the 2.5–97.5 percentile interval of $\mathcal{Q}$ for all models and emission lines over training noise level. On the right, the distributions of $\mathcal{Q}$ for each of the emission spectra are displayed. Small differences in the model prediction scores can be observed, but are eclipsed by the spread of error distributions. At high noise levels, the $\mathcal{Q}$-distribution is wider, and models achieve lower scores on average. Examples for $\sigma_{\max}/h_{\text{ref}} = 2^{-6}$ and $\sigma_{\max}/h_{\text{ref}} = 2^{-2}$ are given in Figure 3.

There is a small difference in the $\mathcal{Q}$-distributions for the $\mathcal{C}_{\alpha_1}$ and $\mathcal{C}_{\alpha_{1/2}}$ emission spectra. If peaks caused by the Cu$K\beta$ emission line are added to the pattern, average performance drops slightly. Since close-by emission lines affect peak shape differently compared to faraway ones, we also investigated the emission spectra $\mathcal{C}_3$ and $\mathcal{C}_4$. One and two emission lines in close proximity to the $\mathcal{C}_{\alpha_{1/2}}$-lines, respectively, are added to the emission spectrum. Performance of models trained on these nonphysical emission spectra is comparable to the monochromatic Cu$K\alpha_1$ and the polychromatic Cu$K\alpha_{1/2}$ spectra.

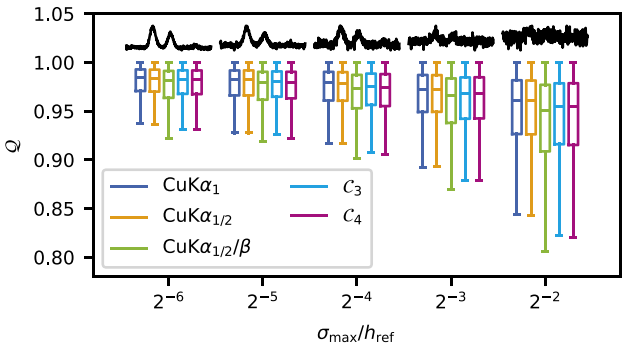

FIGURE 8 | Prediction quality and 2.5% to 97.5% intervals for $\mathcal{Q}$ over normalized $\sigma_{\max}$. 2^{-2} corresponds to the maximum noise we tested, and 2^{-6} is the smallest noise. All tested emission spectra are similar, with the $\mathcal{C}_{\alpha_{1/2},\beta}$-spectrum performing slightly worse on average and spreading slightly wider. Above each noise level, a segment of a synthetic XRD pattern is shown as an example for the noisiest pattern in each dataset.

Compared to the influence of noise level $\sigma_{\max}$, the effects of changing the emission spectrum are small. If the emission lines have similar energies, polychromatic emission spectra do not seem to be detrimental to prediction quality. If emission lines are further apart, like the Cu$K\beta$ line relative to the Cu$K\alpha_{1/2}$-lines, prediction quality is affected more, but only slight differences are visible. Therefore, both mono- and polychromatic emission spectra can be used when evaluating XRD patterns with machine learning. Monochromators could conceivably be removed from existing XRD instruments to increase X-ray intensities at the detector, if higher throughput is needed.

Furthermore, composition estimation can in theory be made highly tolerant to noise. The example patterns in Figure 8 in orange and read barely allow distinguishing peaks from noise, but our models are still able to achieve $\mathcal{Q}$>0.95 for 50% of XRD patterns at the highest noise level (compared to 90% at the lowest noise level we tested). This should allow large increases in throughput, depending on reduction in accuracy one is willing to accept.

We conclude that **ComaNet** is in theory a viable solution for XRD composition analysis and may enable the use of XRD in high-throughput scenarios through exceptional noise and secondary peak tolerance.

2.5.2 | XRD Pattern Translation

To assist volume fraction estimation, we provide a network aiding visual inspection of XRD patterns recorded using polychromatic emission spectra. **PatraNet** can be used to translate $\mathcal{C}_{\alpha_{1/2},\beta}$ XRD patterns to $\mathcal{C}_{\alpha_{1/2}}$ XRD patterns, restoring interpretability to patterns containing Cu$K\beta$-peaks by removing said peaks and reducing noise. Since this network is only intended for visual examination, we test one configuration with a noise level of $\sigma_{\max} = 0.25 h_{\text{ref}}$ in the input XRD patterns. This noise level far exceeds what one usually would encounter in low-quality XRD patterns. Adequate performance at this noise level should therefore be sufficient for real-world application. An example for a synthetic XRD pattern translation with maximum noise can be found in Figure 9. Overall, the Cu$K\alpha_{1/2}$-pattern (ground truth) and translated pattern match rather closely, and peaks caused by the Cu$K\beta$ emission line and noise are removed, while the background profile is retained. However, some small peaks are not carried over properly. As such, this method should only be used for visual inspection and not as preprocessing for further analysis.

When testing on the entire synthetic test dataset described in subsection 2.2, **PatraNet** can successfully remove peaks introduced by the Cu$K\beta$-line. Signals can be translated between the emission spectra reliably, but some peak intensities are matched incorrectly, with the largest errors occurring at narrow or small peaks. Figure 10 shows the distribution of the two metrics for **PatraNet** on the synthetic test dataset. High values for M_{mean} indicate that the model either is not able to successfully remove noise, or that the pattern baseline cannot be correctly matched. **PatraNet**

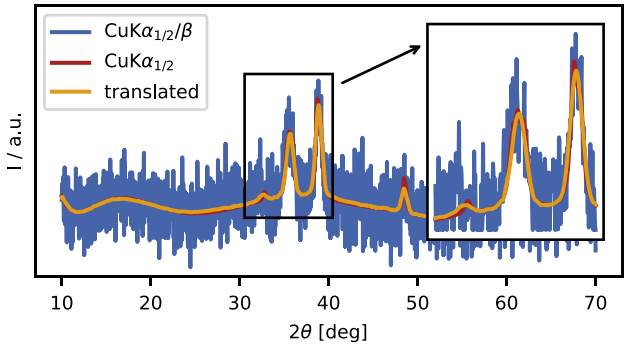

FIGURE 9 | Example XRD pattern translation using **PatraNet**. Here, we chose the mean noise level we used during training ($\sigma = 0.125 h_{\text{ref}}$). While small discrepancies exist in the peak heights, the general shape of the pattern is transferred well.

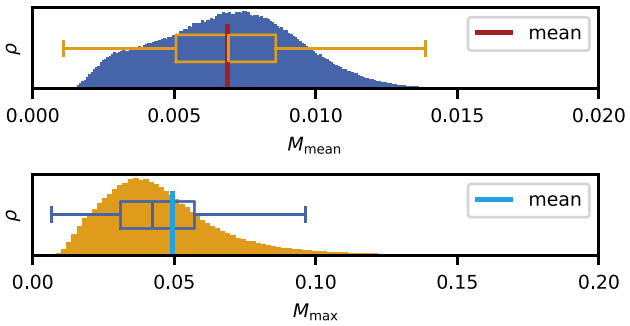

FIGURE 10 | Distributions of metrics for **PatraNet** when training and testing using $\sigma_{\text{max}} = 0.25$.

achieves M_{mean} (MAE per-pattern) of 0.0069 on average with a 95% quantile at 0.011, indicating overall close matches between model output and desired pattern shape and effective noise removal.

However, M_{mean} does not capture errors regarding individual peaks. We therefore utilize M_{max} to assess peak height correctness. As displayed in Figure 10, M_{max} has it's mean at 0.049, and 95% of XRD patterns are translated with $M_{\text{max}} \leq 0.097$.[5]

For reference, the mean M_{mean} between the unprocessed inputs and desired model outputs is 0.064, with the 95% quantile at 0.055. The mean M_{max} between inputs and desired outputs over the entire testing dataset is 0.33, with the 95% quantile at 0.47.

We conclude that **PatraNet** is able to translate XRD patterns from $\mathcal{C}_{\alpha_{1/2},\beta}$ to $\mathcal{C}_{\alpha_{1/2}}$.[6] It simultaneously performs noise filtering, as output XRD patterns in the dataset are constructed without noise. **PatraNet** therefore is suitable for XRD pattern translation in our use case, increasing pattern interpretability. However, we reiterate that it's usage as a preprocessing tool neither tested nor necessary.

2.6 | Experimental Verification

To verify of our method on experimental data, we created a small dataset of five mixtures of CuO and Fe_3O_4 (0, 25, 50, 75, and 100 wt-%CuO). XRD patterns for each mixture were measured using a Cu$K\alpha_{1/2}$-XRD (Bruker AXS D8 with DAVINCI Diffractometer). 2θ was varied from 10° to 70° and a step size of 0.0204°. The patterns were linarly interpolated to the required model input size for processing. To cover multiple noise levels and the presence of Cu$K\beta$-Peaks, we varied the exposure time (0.1, 1, 2, and 5 s) and performed each measurement duration with and without a monochromator. Because we had discovered an issue in predictions with the method, we later recorded another sample containing 70%CuO. Two example XRD patterns from the experimental dataset are shown in Figure 11, and the properties of all experimental samples are listed in Table 2. Accounting for the noise levels and emission spectrum described above, each sample is measured eight times, yielding 48 XRD patterns in our experimental set. All patterns were acquired with 2θ ranging from 10° to 70° and a step size of 0.0204°. These patterns (of length 2934) are interpolated to 2048 steps and normalized to the range $[0, 1]$ before processing by the networks.

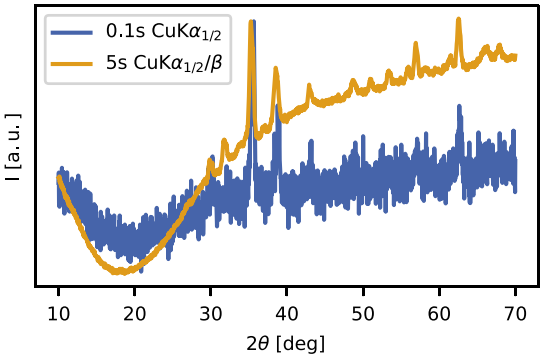

FIGURE 11 | Experimental XRD's for the 50/50 mixture with exposure times of 0.1 and 5 s per step and Cu$K\alpha_{1/2}$ or Cu$K\alpha_{1/2}/\beta$-radiation.

TABLE 2 | Parameters for the experimental samples. All samples were recorded with exposure times of 0.1, 1, 2, and 5 s using Cu$K\alpha_{1/2}$ and Cu$K\beta$-radiation.

Sample Index	1	2	3	4	5	6
wt-% CuO	0	25	50	70	75	100
wt-% Fe_3O_4	100	75	50	30	25	0

2.6.1 | Quantification

Two instances of **ComaNet** were trained to analyze the experimental data, one for the Cu$K\alpha_{1/2}/\beta$ and Cu$K\alpha_{1/2}$ emission spectra, respectively. The models were trained for 20 epochs on 2 million XRD patterns with a 60/40 training/validation split, similar to the synthetic parameter study. Patterns are augmented using Gaussian noise with a maximum standard deviation of $0.05 h_{\text{ref}}$, and the maximum background height was set to $3 h_{\text{ref}}$, according to the background amplitudes in the experimental XRD patterns (see Figure 11). The models' validation $\mathcal{Q}$ converged at 0.971 and 0.968 for the Cu$K\alpha_{1/2}$ and Cu$K\alpha_{1/2}/\beta$-model, respectively. For the full data generation and training parameters, please refer to the GitHub repository.[7]

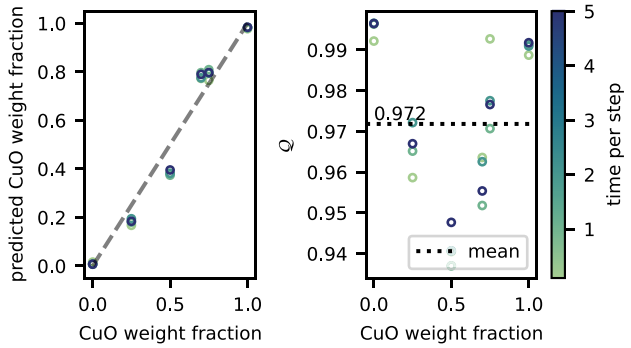

FIGURE 12 | Experimental results on the Cu$K\alpha$-XRD patterns.

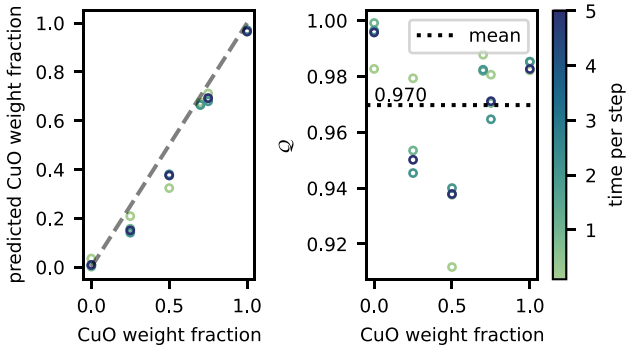

FIGURE 13 | Experimental results on the Cu$K\alpha_{1/2}/\beta$-XRD patterns.

Figure 12 shows **ComaNet**'s performance on experimental Cu$K\alpha_{1/2}$ patterns. On average, we achieve a mean $\mathcal{Q}$ of 0.972, with the lowest $\mathcal{Q}$ for the noisiest patterns at 0.937.

Figure 13 shows a similar relationship for Cu$K\alpha_{1/2}/\beta$-radiation. We observe mean $\mathcal{Q}$ of 0.97, and 0.91 in the worst case. This suggests that our models generalize to experimental data, and are able to predict volume fractions even for low-exposure XRD patterns.

From this, we can conclude that our method performs well on experimental data. While there is a slight drop in performance for noisier patterns, reducing exposure times may be an acceptable tradeoff. This entails a reduction of exposure times ranging from 60% to 98%, depending on how much precision is required.

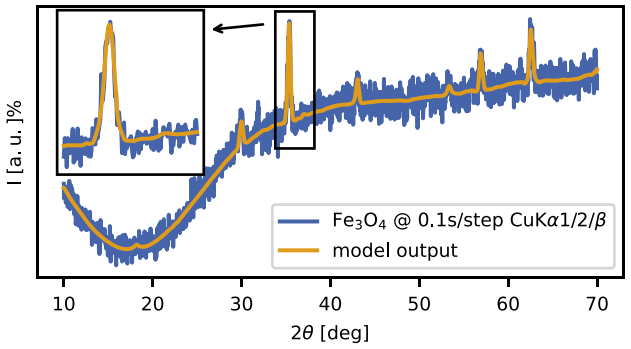

FIGURE 14 | Example application of **PatraNet** to an experimental XRD pattern. The 100% Fe_3O_4 sample was recorded for 0.1 s per step using Cu$K\alpha_{1/2}/\beta$-radiation. Our model produces a plausigle and legible version of the experimental pattern.

2.6.2 | Translation

Quantitative evaluation of **PatraNet**'s capabilities on experimental data is not possible, as background profiles change between Cu$K\alpha_{1/2}$ and Cu$K\alpha_{1/2}/\beta$-radiation. However, we can show the model's output for exemplary patterns. Figure 14 shows an example translation from our experimental dataset. The model produces a legible version of the pattern. Background, peak positions and intensities are approximately reproduced, like in the synthetic tests, but Cu$K\beta$-peaks and noise are removed.

3 | Conclusion

We present **HiVE**, a novel system for processing XRD patterns recorded without the need for monochromators in XRD instruments. Our volume fraction estimation model's prediction quality is presented for a large range of noise levels. Using our analysis of synthetic and experimental XRD patterns, we conclude that our method may be used in high-throughput scenarios, where low exposure times cause high noise in XRD patterns. Furthermore, show that our model is able to analyze XRD patterns produced by polychromatic emission spectra, removing the need for monochromators from XRD devices.

ComaNet can be made extremely robust against noise, allowing operation on XRD patterns where peaks are barely discernible by eye. To restore interpretability for these types of XRD patterns, we introduce **PatraNet** to "translate" XRD patterns to conventional emission spectra and exposure times.

Due to our model's sizes and our data generation pipeline's speed, it is possible to simulate training data and generate new models for specific problems within minutes. We therefore don't require a "god-model" capable of analyzing arbitrary XRD patterns. Instead, when a material system needs to be analyzed, we propose training a lightweight network specifically for that task. The model can be trained and verified without handling the combinatorial explosion which arises from the analysis of arbitrary compositions.

In preliminary work [40], we have shown using synthetic data that our approach can in general work for more complex material mixtures for a monochromatic emission spectrum. Here, we show that the approach works on synthetic and experimental data for a binary mixture and multiple emission lines. Future work should continue building up complexity by testing more complex mixtures of components using polychromatic emission spectra.

Acknowledgments

Open access funding enabled and organized by Project DEAL. This work was funded by the program Material Systems Engineering of the

Helmholtz Association (43.31.02). This group also receives funding from the Post Lithium Storage Cluster of Excellence (PoLiS).

Open Access funding enabled and organized by Projekt DEAL.

Funding

This study was supported by Helmholtz-Gemeinschaft (43.31.02).

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

Code for dataset generation and neural network training is available at https://github.com/hawo-hoefer/yaxs, https://github.com/hawo-hoefer/patranet-training and https://github.com/hawo-hoefer/cuo-fe3o4-xrd-analysis. Where experimental data was used for testing, it has been made available in the corresponding repository.

Endnotes

[1] https://github.com/hawo-hoefer/yaxs

[2] https://github.com/hawo-hoefer/yaxs

[3] https://github.com/hawo-hoefer/cuo-fe3o4-xrd-analysis

[4] https://github.com/hawo-hoefer/patranet-training

[5] It is to be noted that there are a small number of patterns in our 400 000 pattern test dataset (approx. 1.73%) whose $M_{\max}$ exceeds 0.2. These catastrophic errors occur at narrow peaks, where impurity peaks overlap with regular peaks and at the boundaries of the input XRD pattern.

[6] Preliminary testing on $\mathcal{C}_{\alpha_{1/2},\beta}$ to $\mathcal{C}_{\alpha_1}$-translation shows similar results, but we have chosen to omit it for brevity.

[7] https://github.com/hawo-hoefer/cuo-fe3o4-xrd-quantification

References

1. H. Hwang, S. M. Choi, J. Oh, et al., "Integrated Application of Semantic Segmentation-Assisted Deep Learning to Quantitative Multi-Phased Microstructural Analysis in Composite Materials: Case Study of Cathode Composite Materials of Solid Oxide Fuel Cells," *Journal of Power Sources* 471 (2020): 228458.

2. H. Kim, J. Han, and T. Y.-J. Han, "Machine Vision-Driven Automatic Recognition of Particle Size and Morphology in SEM Images," *Nanoscale* 12, no. 37 (2020): 19461.

3. L. Rettenberger, N. J. Szymanski, Y. Zeng, et al., "Uncertainty-Aware Particle Segmentation for Electron Microscopy at Varied Length Scales," *npj Computational Materials* 10, no. 1 (2024): 124.

4. M. P. Schilling, S. Schmelzer, J. E. U. Gómez, A. A. Popova, P. A. Levkin, and M. Reischl, "Grid Screener: A Tool for Automated High-Throughput Screening on Biochemical and Biological Analysis Platforms," *IEEE Access* 9 (2021): 166027.

5. H. M. Rietveld, "A Profile Refinement Method for Nuclear and Magnetic Structures," *Journal of Applied Crystallography* 2, no. 2 (1969): 65.

6. D. Jha, K. Narayanachari, R. Zhang, et al., "Enhancing Phase Mapping for High-Throughput X-Ray Diffraction Experiments using Fuzzy Clustering," in ICPRAM, (2021), 507–514.

7. V. Stanev, V. V. Vesselinov, A. G. Kusne, G. Antoszewski, I. Takeuchi, and B. S. Alexandrov, "Unsupervised Phase Mapping of X-Ray Diffraction Data by Nonnegative Matrix Factorization Integrated with Custom Clustering," *npj Computational Materials* 4, no. 1 (2018): 43.

8. Y. Zhou, B. Wu, J. Wang, and H. Wang, "Effect of Signal-to-Noise Ratio on the Automatic Clustering of X-Ray Diffraction Patterns from Combinatorial Libraries," *Materials Genome Engineering Advances* 2, no. 1 (2024): e27.

9. Z. Zhao, Y. Jin, P. Shi, et al., "An Improved High-Throughput Data Processing Based on Combinatorial Materials Chip Approach for Rapid Construction of Fe–Cr–Ni Composition-Phase Map," *ACS Combinatorial Science* 21, no. 12 (2019): 833.

10. C. Long, J. Hattrick-Simpers, M. Murakami, et al., "Rapid Structural Mapping of Ternary Metallic Alloy Systems Using the Combinatorial Approach and Cluster Analysis," *Review of Scientific Instruments* 78, no. 7 (2007): 072217.

11. Y. Iwasaki, A. G. Kusne, and I. Takeuchi, "Comparison of Dissimilarity Measures for Cluster Analysis of X-Ray Diffraction Data from Combinatorial Libraries," *npj Computational Materials* 3, no. 1 (2017): 4.

12. H. Xing, B. Zhao, Y. Wang, et al., "Rapid Construction of Fe–Co–Ni Composition-Phase Map by Combinatorial Materials Chip Approach," *ACS Combinatorial Science* 20, no. 3 (2018): 127.

13. J. Schuetzke, S. Schweidler, F. R. Muenke, et al., "Accelerating Materials Discovery: Automated Identification of Prospects from X-Ray Diffraction Data in Fast Screening Experiments," *Advanced Intelligent Systems* 6, no. 3 (2024): 2300501.

14. J.-W. Lee, W. B. Park, J. H. Lee, S. P. Singh, and K.-S. Sohn, "A Deep-Learning Technique for Phase Identification in Multiphase Inorganic Compounds Using Synthetic XRD Powder Patterns," *Nature Communications* 11, no. 1 (2020): 86.

15. F. Oviedo, Z. Ren, S. Sun, et al., "Fast and Interpretable Classification of Small X-Ray Diffraction Datasets Using Data Augmentation and Deep Neural Networks," *npj Computational Materials* 5, no. 1 (2019): 60.

16. J. Schuetzke, N. J. Szymanski, and M. Reischl, "Validating Neural Networks for Spectroscopic Classification on a Universal Synthetic Dataset," *npj Computational Materials* 9, no. 1 (2023): 100.

17. J. Schuetzke, A. Benedix, R. Mikut, and M. Reischl, "Enhancing Deep-Learning Training for Phase Identification in Powder X-Ray Diffractograms," *IUCrJ* 8, no. 3 (2021): 408.

18. N. J. Szymanski, C. J. Bartel, Y. Zeng, Q. Tu, and G. Ceder, "Probabilistic Deep Learning Approach to Automate the Interpretation of Multi-Phase Diffraction Spectra," *Chemistry of Materials* 33, no. 11 (2021): 4204.

19. B. Zhao, S. Wolter, and J. A. Greenberg, "Application of Machine Learning to X-Ray Diffraction-Based Classification," in *Anomaly Detection and Imaging with X-Rays (ADIX) III*, vol. 10632 (SPIE, 2018), 20–25.

20. C. Royse, S. Wolter, and J. A. Greenberg, "Emergence and Distinction of Classes in XRD Data via Machine Learning," in *Anomaly Detection and Imaging with X-Rays (ADIX) IV*, vol. 10999, (SPIE, 2019), 63–70.

21. W. B. Park, J. Chung, J. Jung, et al., "Classification of Crystal Structure Using a Convolutional Neural Network," *IUCrJ* 4, no. 4 (2017): 486.

22. B. D. Lee, J.-W. Lee, J. Ahn, S. Kim, W. B. Park, and K.-S. Sohn, "A Deep Learning Approach to Powder X-Ray Diffraction Pattern Analysis: Addressing Generalizability and Perturbation Issues Simultaneously," *Advanced Intelligent Systems* 5, no. 9 (2023): 2300140.

23. N. J. Szymanski, S. Fu, E. Persson, and G. Ceder, "Integrated Analysis of X-Ray Diffraction Patterns and Pair Distribution Functions for Machine-Learned Phase Identification," *npj Computational Materials* 10, no. 1 (2024): 45.

24. P. Hosein and J. Greasley, "An Optimization-Based Supervised Learning Algorithm for PXRD Phase Fraction Estimation," *Materials Today Communications* 36 (2023): 106423.

25. J. Greasley and P. Hosein, "Exploring Supervised Machine Learning for Multi-Phase Identification and Quantification from Powder X-Ray Diffraction Spectra," *Journal of Materials Science* 58, no. 12 (2023): 5334.

26. S. Y. Park, B.-K. Son, J. Choi, H. Jin, and K. Lee, "Application of Machine Learning to Quantification of Mineral Composition on Gas Hydrate-Bearing Sediments, Ulleung Basin, Korea," *Journal of Petroleum Science and Engineering* 209 (2022): 109840.

27. J.-W. Lee, W. B. Park, M. Kim, S. P. Singh, M. Pyo, and K.-S. Sohn, "A Data-Driven XRD Analysis Protocol for Phase Identification and Phase-Fraction Prediction of Multiphase Inorganic Compounds," *Inorganic Chemistry Frontiers* 8, no. 10 (2021): 2492.

28. D. Kim, J. Choi, D. Kim, and J. Byun, "Predicting Mineralogy by Integrating Core and Well Log Data Using a Deep Neural Network," *Journal of Petroleum Science and Engineering* 195 (2020): 107838.

29. Z. Zhou, C. Li, X. Bi, et al., "A Machine Learning Model for Textured X-Ray Scattering and Diffraction Image Denoising," *npj Computational Materials* 9, no. 1 (2023): 58.

30. B. H. Toby and R. B. Von Dreele, "GSAS-II: The Genesis of a Modern Open-Source All Purpose Crystallography Software Package," *Journal of Applied Crystallography* 46, no. 2 (2013): 544.

31. S. P. Ong, W. D. Richards, A. Jain, et al., "Python Materials Genomics (pymatgen): A Robust, Open-Source Python Library for Materials Analysis," *Computational Materials Science* 68 (2013): 314.

32. B. O'Connor, D. Li, and B. Hunter, "The Importance of the Specimen Displacement Correction in Rietveld Pattern Fitting with Symmetric Reflection-Optics Diffraction Data," *Advances in X-Ray Analysis* 44 (2001): 96.

33. B. D. Cullity, *Elements of X* (Wesley Mass, 1978), 127–131.

34. H. Mimura, Y. Neo, and T. Aoki, "Novel Light Sources using Micro Field Emitters," in *Eighth International Conference on Correlation Optics*, vol. 7008, (SPIE, 2008), 260–268.

35. H. H. Otto, "X-Ray Powder Diffraction: Why Not Use CuKβ Radiation?," *Journal of Analytical Sciences, Methods and Instrumentation* 8, no. 3 (2018): 37.

36. A. Paszke, S. Gross, F. Massa, et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," *Advances in Neural Information Processing Systems* 32 (2019).

37. O. Ronneberger, P. Fischer, and T. Brox, Medical Image Computing and Computer-Assisted Intervention–MICCAI. 2015: 18th International Conference, (Springer, 2015), 234–241.

38. M. Tripathi, "Facial Image Denoising Using AutoEncoder and UNET," *Heritage and Sustainable Development* 3, no. 2 (2021): 89.

39. D. Yoo, N. Kim, S. Park, A. S. Paek, and I. S. Kweon, *Computer Vision–ECCV 2016: 14th European Conference*, (Springer, 2016), 517–532.

40. H. H. Höfer, A. Orth, S. Schweidler, B. Breitung, J. Aghassi-Hagmann, and M. Reischl, "Quantitative Convolutional Neural NetworkBased Multi-Phase XRD Pattern Analysis," *Current Directions in Biomedical Engineering* 10 (2024); 307–310.

Supporting Information

Additional supporting information can be found online in the Supporting Information section.