Peak Area Detection Network for Directly Learning Phase Regions from Raw X-ray Diffraction Patterns

Dipendra Jha*, Aaron Gilad Kusne†, Reda Al-Bahrani*, Nam Nguyen†,
Wei-keng Liao*, Alok Choudhary*, and Ankit Agrawal*
*Department of Electrical Engineering and Computer Science, Northwestern University
†Materials for Energy and Sustainable Development Group, National Institute of Standards and Technology
*{dkj755, rav650, wkliao, choudhar, ankitag}@eecs.northwestern.edu,
†{aaron.kusne@nist.gov, namuunguyen@gmail.com}

***Abstract*—X-ray diffraction (XRD) is a well-known technique used by scientists and engineers to determine the atomic-scale structures as a basis for understanding the composition-structure-property relationship of materials. The current approach for the analysis of XRD data is a multi-stage process requiring several intensive computations such as integration along 2θ for conversion to 1D patterns (intensity-2θ), background removal by polynomial fitting, and indexing against a large database of reference peaks. It impacts the decisions about the subsequent experiments of the materials under investigation and delays the overall process. In this paper, we focus on eliminating such multi-stage XRD analysis by directly learning the phase regions from the raw (2D) XRD image. We introduce a peak area detection network (PADNet) that directly learns to predict the phase regions using the raw XRD patterns without any need for explicit preprocessing and background removal. PADNet contains specially designed large symmetrical convolutional filters at the first layer to capture the peaks and automatically remove the background by computing the difference in intensity counts across different symmetries. We evaluate PADNet using two sets of XRD patterns collected from SLAC and Bruker D-8 for the Sn-Ti-Zn-O composition space; each set contains 177 experimental XRD patterns with their phase regions. We find that PADNet can successfully classify the XRD patterns independent of the presence of background noise and perform better than the current approach of extrapolating phase region labels based on 1D XRD patterns.**

I. Introduction

In materials science and crystallography, X-ray diffraction (XRD) is a widely used experimentation technique to probe materials at the atomic level. XRD analysis is used by scientists and engineers to understand atomic-scale crystal structures and predict their properties [1]–[5]. XRD patterns not only provide the geometrical information about the crystal structure, they are also used to determine the possible flaws in materials [6]. High throughput experimental techniques developed over the last few decades have accelerated the exploration of material properties. Combinatorial methods allow experimentalists to synthesize hundreds or thousands of materials at a time, with each sample varying by synthesis and processing parameters [7]. Composition spreads are one example, where a wafer is generated containing hundreds of samples, each varying in composition. Once such a wafer is generated, the properties of each sample can be rapidly measured using scanning microscopy techniques [8]. As a result, over the course of hours, XRD data can be collected for hundreds or thousands of materials.

Currently, human experts analyze the XRD patterns using domain knowledge such as peak shape and location; they are correlated with the sample composition and known phases to identify the phases in the measured sample. The current approach for the analysis of XRD patterns is a multi-stage process composed of multiple computationally intensive steps. First step is to convert the raw 2D XRD pattern to an intensity-2θ (1D) pattern by mapping the raw XRD image to the χ vs 2θ space and then integrating the intensity peaks along the 2θ axis [9]. XRD patterns are often noisy due to a collection of issues including background radiation, detector noise, and low count of incident X-rays. In addition, other background issues may be introduced by the sample-detector configuration, resulting in significantly varying measurement background from sample to sample [10]. The presence of highly irregular background makes the peak searching procedure complicated. Hence, next the background signal is removed from the 1D patterns by fitting background curve [11]–[13]. This is followed by indexing the peaks against an existing database of reference peaks and correlating with the sample composition to identify the phases in the measured sample using available software which often requires verification by a domain expert.

Over the last decade, machine learning has been used to accelerate the process of indexing using 1D XRD patterns [14]–[16]. Clustering has been used to sort samples into groups of materials that share the same constituent phases - thus reducing the number of samples required to index for unique phases [17], [18]. When plotted against the synthesis and processing parameters used to generate the samples, these clusters describe geometric "phase regions" - regions of the generative space where materials are expected to share the same constituent phases. Additionally, once a subset of samples have been sorted into phase regions, classification has been used to extrapolate these phase region labels to the rest of the samples [16]. Recently, Park et al. [19] used a CNN to classify 1D XRD patterns into space group, extinction-group and crystal-system classification. They used 150000 powder XRD patterns calculated from the structure solutions of entries in the Inorganic Crystal Structure Database (ICSD) using DFT.

Time is a limiting factor when collecting and analyzing X-ray diffraction data. For typical lab systems, low beam intensity means measuring each sample can take tens of minutes

978-1-7281-2009-6/$31.00

to hours. Additionally, when performing X-ray diffraction at the beamline, measurements only take tens of seconds, but total time is often limited to hours or days. Accelerating the computational time required in data analysis can directly impact the measurement experiment since it directly impacts the decisions. Hence, our goal is to build a predictive model for eliminating the current computationally intensive process of multi-stage XRD analysis process. We focus on directly learning the phase regions from the raw (2D) XRD patterns through classification using deep learning to make the overall process automatic and faster.

In this paper, we introduce a Peak Area Detection Network (PADNet) to directly learn to predict the phase regions from the raw XRD patterns without any need for preprocessing or background noise removal. PADNet is a specially designed CNN that contains large symmetrical convolutional filters with filter size of 50×50 in the first layer. These filters are initialized using either 1 or -1 across different symmetries and they compute the difference in intensity counts across different symmetries which enable them to capture the peak areas and automatically remove the background noise. To evaluate the proposed approach, we experiment using two sets of XRD patterns from SLAC [8] and Bruker D-8 [20]; each of them contains 177 XRD patterns from a Sn-Ti-Zn-O thin-film, composition-spread, combinatorial library sample with eight phase regions as the labels. The XRD patterns from SLAC contain significant irregular background which varies by sample, while the ones from Bruker contain comparably low background which, as a function of 2θ, does not significantly vary from sample to sample. To our knowledge there does not exist any algorithm for removing background noise for the raw 2D XRD image; hence, we also explore some novel background removal techniques based on minimum and mean convolutional filters.

We evaluate the performance of PADNet using 10-fold cross validation. PADNet achieves an overall classification accuracy of 84% for the multi-class labeling task, with the SLAC model performing slightly better than the Bruker model. Our results demonstrates that PADNet can successfully predict the phase regions from the raw 2D XRD patterns independent of presence of background noise. We also compared our approach against the recent approach of phase region classification using 1D XRD patterns from Bunn et al. [16]; PADNet significantly outperformed the AdaBoost classifier for both datasets.

II. Background and Motivation

A. Background

X-ray diffraction is an atomic scale probing technique for determining the crystal structure of materials [2]–[5]. The crystal structure causes the beam of incident X-rays to diffract into many specific directions; a 3D image representing the density of electrons in the crystal can be constructed by measuring the angles and intensities of the diffracted intensity patterns. An X-ray diffraction image is a plot of the intensity of X-rays scattered at different angles by a materials sample, as measured by a 2D detector, with each pixel measuring the number of incident X-rays. The atomic-scale structures of materials can be determined using the XRD technique [1].

The XRD pattern from a material composed of periodic atomic structures is composed of multiple sharp spots known as Bragg diffraction peaks; the positions and intensities of these peaks determine the phase of the materials - the specific chemistry and atomic arrangement. For instance, quartz, cristobalite and glass are all different phases of SiO_2; they are chemically identical but the atoms are arranged differently, the XRD pattern is distinct for each phase. A phase map represents the physical conditions at which thermodynamically distinct phases occur and coexist. The constituent phases in the phase map represent the different crystal lattice structures for varying material composition. Scott [9] provides more details about X-ray powder diffraction.

B. Motivation

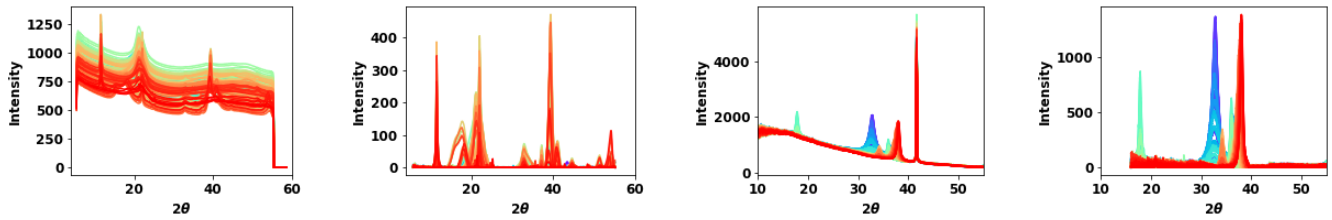
Our current work is motivated by the success of convolutional neural networks for image classification [21], [22]. In our previous work, we have shown the efficiency of deep neural networks in learning crystal orientations directly from electron diffraction patterns [23]. Recently, Park et al. applied convolutional neural networks for classification of crystal structure using 1D XRD patterns. Since deep neural networks are supposed to require large training datasets and the experimentally measured XRD patterns are limited, they used 150,000 1D XRD patterns calculated from the structure solutions of every entry from the Inorganic Crystal Structure Database [24]. However, our previous work has shown that deep neural networks can be leveraged even with relatively smaller datasets and perform better than traditional machine learning techniques [25].

Directly using the raw 2D XRD patterns is also beneficial from the perspective of information content. The conversion from 2D raw patterns to 1D intensity-2θ patterns results in loss of important information due to their limited representation. The peak characteristics such as peak height, peak width, presence of secondary peaks (peak doublets), are very critical to correctly understand the materials structure. For instance, the peak broadening can be used to quantify the average crystalline size of nanoparticles, lines on the 2D raw pattern represent polycrystalline structure, and points on the 2D raw pattern represent very well ordered crystalline structure. Such fine grained differentiation is very critical to understand the true structure of materials. However, these facts are ignored because during the conversion to 1D, such information is lost.

III. Methods and Technical Solutions

A. Challenges

The primary challenge of XRD data analysis is the presence of background noise which can be highly irregular such as in the case of SLAC as shown in Figure 1. During the experiment, several factors can impact the XRD pattern captured, some of which are beyond human control. It depends on multiple effects: machine setup, air around the wafer, etc. The presence of background makes it difficult to detect the intensity peaks



(a) 1D XRD Pattern from SLAC with background

(b) 1D XRD Pattern from SLAC after background removal

(c) 1D XRD Pattern from Bruker with background

(d) 1D XRD Pattern from Bruker after background removal

Fig. 1: 1D XRD Patterns from SLAC and Bruker. The XRD patterns from SLAC contains highly irregular noise while the noise in the case of XRD patterns from Bruker is a constant function of 2θ.

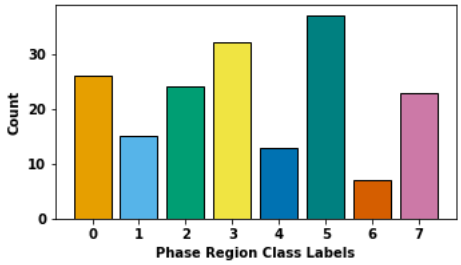


Fig. 2: Distribution of class labels for the two XRD datasets. XRD Patterns are collected for the same composition space of Sn-Ti-Zn-O from both Bruker and SLAC; hence, they refer to same samples.

which are important for obtaining the crystal information. For beamline, the resulting background is not a simple bias to subtract. Hence, background removal is a primary concern for XRD pattern analysis for the domain scientists.

Although several techniques exist for removing background in the 1D XRD samples [11]–[13], and parsed 2D XRD patterns [26], to our knowledge, there exists no technique for removal of background noise from the raw XRD patterns coming directly from the experiments. The raw 2D XRD patterns are convoluted rather than being a 2D rectangle; the background removal methods for parsed 2D XRD patterns do not work for the raw XRD patterns. For example, we implemented the Cache-efficient 2D Bruckner Filter from Baur et al. [27] which is designed for parsed 2D XRD patterns, but it did not work for the case of raw 2D XRD patterns.

Another challenge associated with this task is the limited dataset size. Our dataset contains only 177 XRD samples (and we have eight classes to learn). Since deep neural networks are supposed to require large training datasets and the experimentally measured XRD patterns are limited, Parker et al. [19] used 150,000 1D XRD patterns calculated from the structure solutions of every entry from the Inorganic Crystal Structure Database [24]. However, here our goal is to directly learn from the raw 2D XRD images coming from experiments.

B. Datasets

We leverage the XRD patterns collected from Stanford Linear Accelerator Center (SLAC) [8], and Bruker D-8 [20] at National Institute of Standards and Technology (NIST). SLAC has a high throughput system for XRD experiments [8]; it outputs a single XRD pattern for a specific range of 2θ; the configuration used gives a 2θ range of 5.365 to 58.566 for our experiments. The Bruker system was used to collect two diffraction frames centered at the 2θ values of 25 and 45, the range for the two frames were [10, 40] and [30, 60]. XRD patterns from SLAC contain more features due to the high energy of the beam, we are able to resolve XRD with greater signal to noise ratio at greatly reduced exposure times; however, the instrument is less available.

Each dataset is composed of 177 XRD patterns for the material alloy system with different compositions of Tin, Titanium, and Zinc (Oxygen is also present but not controlled) from experiments. Each XRD pattern is of size 2048×2048 containing the intensity values; hence, they are not like RGB images used for image classification such as in ImageNet [21]. In addition to the XRD patterns, the composition information for each sample is also available in the dataset. The samples were labeled by converting to 1D, clustered, then followed by human expert validation. There are eight phase region classes, some represent pure constituent phases while others represent mixed phases. As shown in Figure 2, the distribution of the dataset is not balanced. The largest class has 37 samples while the smallest class has only 7 samples. We used random split during our ten-fold cross validation; for each training set from the random split, the smallest class was present in all of the training set during 10-fold cross validation. One option could be to remove the class with data count below a certain threshold, but our dataset is already limited and some of the phase regions are mixed (combination) of other phase regions (classes); hence, learning one phase region can help in predicting the other phase region. Therefore, we decided not to drop any phase region class from our dataset.

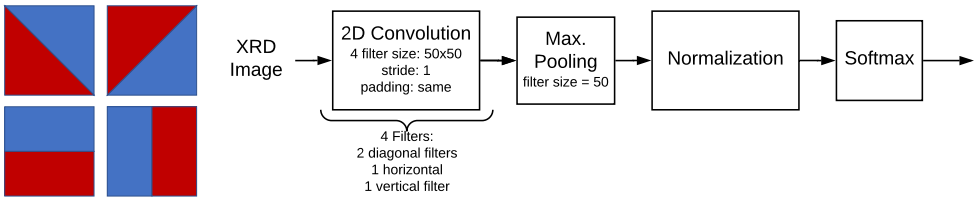
C. Peak Area Detection Network

The Peak Area Detection Network (PADNet) is a deep convolutional neural network for directly learning the phase regions from the 2D raw XRD patterns. PADNet is composed

(a) PADNet for XRD patterns from SLAC

(b) PADNet for XRD patterns from Bruker

Fig. 3: PADNet model architectures for the XRD patterns from SLAC and Bruker. Since both datasets refer to the same composition space of Sn-Ti-Zn-O and have same samples, we constrained both models to have same number of model parameters and same architecture. PADNet for Bruker is composed of two convolutional graph to handle the two XRD patterns compared to the PADNet for SLAC having one convolutional graph since SLAC outputs one XRD image.



(a) Slope Filters

(b) Peak Area Detection Component

Fig. 4: Peak Area Detection Component with Slope Filters: This component contains slope filters which help in peak detection by measuring the difference in slope across different symmetries. The slope filters has two regions - blue representing -1 and red representing 1. Since they are symmetrical, they can effectively detect the high slope areas containing peaks.

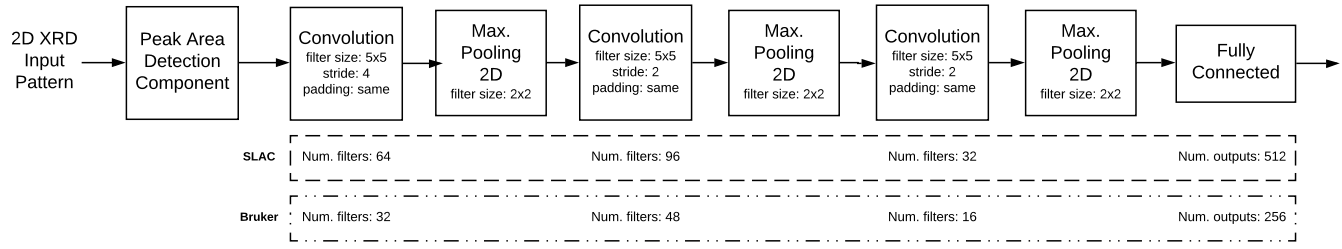


Fig. 5: Convolutional Graph: The component of the CNN network for the raw 2D XRD pattern in the input.

of three components - a convolutional graph with a peak area detection component for each input XRD pattern, a dense network for vector composition in the input and a final classifier network containing dense layers for classification. The PADNet architecture for the XRD patterns from both sources are shown in Figure 3.

The first layer of the convolutional graph is composed of four large convolutional filters with filter size $f = 50$. These filters are initialized in a special symmetrical manner as follows:

$$F1_{i,j} = \begin{cases} -1, & \textit{if } i < j \\ 1, & \textit{otherwise} \end{cases} \quad (1)$$

$$F2_{i,j} = \begin{cases} -1, & \textit{if } i + j > f \\ 1, & \textit{otherwise} \end{cases} \quad (2)$$

$$VF_{i,j} = \begin{cases} -1, & \textit{if } i < f/2 \\ 1, & \textit{otherwise} \end{cases} \quad (3)$$

$$HF_{i,j} = \begin{cases} -1, & \textit{if } j < f/2 \\ 1, & \textit{otherwise} \end{cases} \quad (4)$$

$F1$ and $F2$ are two diagonal filters symmetrical about the diagonals. HF is a filter symmetrical about the horizontal and VF is symmetrical about the vertical. These filters are illustrated visually in Figure 4. Due to their symmetries with opposite signs on the two sides, these filters F measure the difference in intensity counts and the background is automatically implicitly subtracted at each point in Equation 5, where I is the input XRD pattern and we refer to these filters as slope filters.

$$I_{i,j} = \sum (I_{i+x,j+y}) \cdot F_{x,y} \qquad \text{for} - f/2 \leq x, y \leq f/2 \quad (5)$$

The symmetry with opposite sign also means that the value computed on opposite symmetries around the peak will have opposite signs. We are interested in the peak area. Hence, we take their absolute values as follows:

$$I_{i,j} = |I_{i,j}| \tag{6}$$

The value of slope measured at the actual peak should be zero since the intensity counts across a peak should be symmetrical. Hence, to detect the area around a peak including the peak itself, we apply a maximum filter with a filter size $f = 50$ as follows:

$$I_{i,j} = \max(I_{i+x,j+y}) \qquad \text{for } -f/2 \leq x, y \leq f/2 \tag{7}$$

Next, we normalize the outputs from each filter using batch normalization to make the mean zero and variance 1 for proper learning in the next convolutional layers. After the batch normalization, we apply a softmax activation function so that the network puts more emphasis on the points with high slopes and hence, high intensity counts. The softmax function is defined as follows:

$$softmax(I_{i,j}) = \frac{e^{I_{i,j}}}{\sum e^{I_{i,j}}} \tag{8}$$

Figure 4 illustrates the specially designed network component for the peak area detection. The output from the peak area detection component is fed into the next convolutional layer of the convolution graph component. Figure 5 illustrates the convolutional graph of the CNN network used for the two datasets. Since Bruker outputs two XRD patterns, the Bruker model contains two convolutional graphs for each input pattern, but with half number of filters compared to SLAC model. In this way, both SLAC and Bruker models have equal number of trainable parameters, and we can fairly compare their performance with each other using our domain intuition.

The dense network for composition input is composed of two fully connected layers with 256 outputs in each layer. The output from the convolutional graph is concatenated with output from the dense network for composition input and fed into a final classification network that learns to predict the crystal phase label. The final classification network is composed of two layers with 256 outputs in the penultimate and 8 outputs in the last layer. ReLU [28] is used as the activation function. Batch normalization [29] is used after each layer for the faster convergence. Since there are eight phase labels in our datasets, the last fully connected layer in the classification network has softmax activation with eight outputs.

D. 2D Background Removal from Raw XRD Pattern

One of the domain constraints before performing any analysis is how to remove the background from the XRD patterns so that the peaks can be easily detected. Hence, we explored some of the commonly used techniques used for background removal and smoothing for background removal from the raw 2D XRD patterns. The raw 2D XRD patterns are either in GFRM or TIF format, we will refer them as I. These XRD patterns contain intensity values for different values of χ vs 2θ and have a size of 2048×2048. Generally, resizing is done to reduce the computation required; but, we do not perform any resizing as that can lead to information loss.

First, we apply a minimum 2D filter of size $f \times f$ to the raw input image I as follows:

$$MF_{i,j} = \min(I_{i+x,j+y}) \text{ for } -f/2 \leq x, y \leq f/2 \tag{9}$$

MF represents the background obtained by fitting a minimum filter. This can be subtracted from the raw pattern I to obtain the pattern with background removal IM.

After applying the minimum filter, we found that the output pattern IM contains some edges and corners. Next, following the smoothing techniques for 1D XRD patterns such as in [11], we apply a convolutional filter of size $f \times f$ to smooth the background as follows:

$$CS_{i,j} = \sum (IM_{i+x,j+y} \cdot F_{x,y}) \qquad \text{for } -f/2 \leq x, y \leq f/2 \tag{10}$$

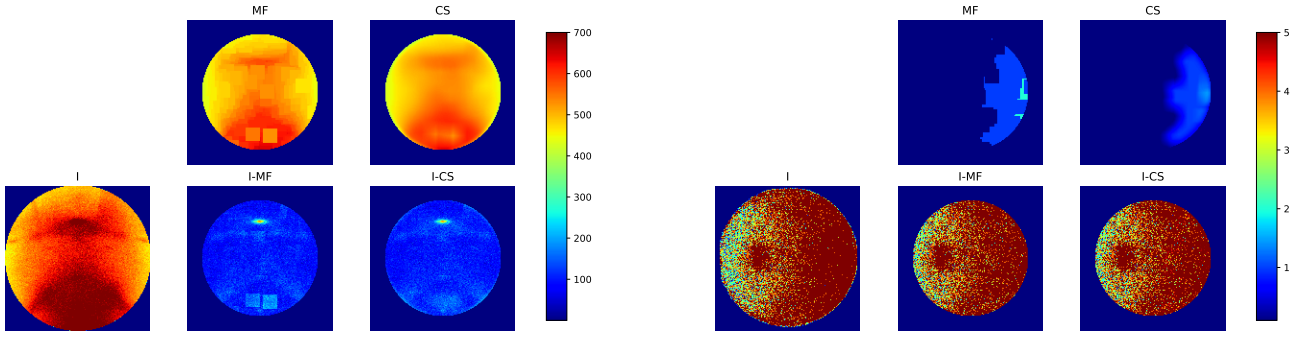
where F is a constant mean filter containing the same value at each position that sum up to one. Since the XRD pattern is large in size 2048×2048, fitting a polynomial using least square can be very expensive without scaling down the size which will impact the quality leading to loss of information. Hence, we applied the convolution mean filter. The smooth background CS can be subtracted from the image I to obtain the XRD pattern without background- IC. We will used both IM and IC as the input XRD pattern evaluate our model performance.

Time Complexity: The time complexity for the background removal for 2D raw XRD patterns is $\mathcal{O}(h \cdot w \cdot f^2)$ where h is the height and w is the width of the input XRD pattern I and f is the filter size.

IV. Empirical Evaluation

A. Experimental Settings

We have used python and the TensorFlow [30] deep learning framework to implement the deep neural network models. For machine learning algorithms, we used their implementations from Scikit-learn [31]. The models were trained using NVIDIA Titan X GPUs. We learn to predict the phase label for understanding the crystal structure. As the dataset is small, we performed a ten-fold cross-validation and aggregated the results. Generally each fold had 160 and 18 samples in the training and test set respectively. The data splitting used for cross-validation is the same across all experiments. We experimented with different types of preprocessing such as normalizing and image whitening, but none of them worked well. Hence, we do not use any kind of preprocessing or feature engineering other than the background removal as stated. We performed a detailed hyperparameter search and architecture search for the PADNet model for both cases, but limited the two PADNet models for both datasets to same



(a) SLAC

(b) Bruker

Fig. 6: Background and Processed XRD images: I is the original XRD pattern. The first row of subfigures represent the background and the second row of subfigures shows the XRD patterns after background removal using the two techniques. We used a filter size of 200 for both the minimum filter and the convolutional mean filter for both cases. For SLAC, the raw XRD pattern is similar to the background images using the two techniques; this illustrates that SLAC image contains high background noise. For Bruker, the pattern after background removal look similar to the raw pattern since the background is very small. This concurs with the domain expertise, thereby suggesting that the proposed background detection module is working as expected.

architectures for a fair comparison between the two sources since both datasets are for the same composition space. Since the dataset is limited, we used early stopping with patience of 30 and also used L2 regularization with regularization coefficient of 0.0001 to avoid overfitting. For training our models, we used a learning rate of 0.001 and Adam as the optimizer. Since we are dealing with a multi-class classification problem, we used the softmax cross entropy as the loss function and the evaluation metric is prediction accuracy which represents the total number of samples correctly classified by the model across all class in the dataset. The evaluation is carried out by training and testing the models on the raw XRD pattern I and the XRD patterns after removing background using the two methods- MF and CS.

B. Background Removal from 2D Raw XRD Patterns

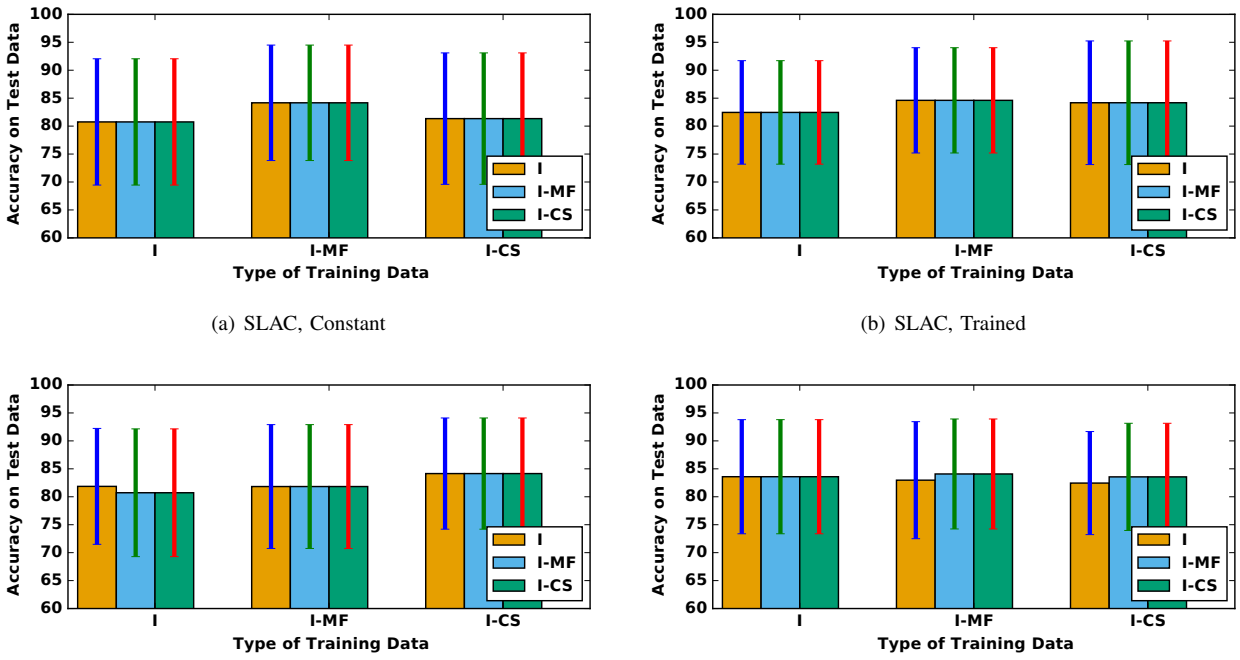
Figure 6 presents the results from background removal using the two techniques. In the case of SLAC, the background is very high and varies within a sample. The background removal using minimum filter method demonstrates that the background obtained using this technique contain some edges and patches. The mean convolution filter removes them by smoothening using the mean of a window of size $f \times f$ where we used $f = 200$. There exist a trade-off between the size of filter and loss of peaks. If the filter size is small, it leads to loss of peaks. If the filter size is large, the background signal is still present in the output. Also, large value for filter size makes it computationally expensive since the computations required are directly proportional to the square of filter size f. The convolutional operation to compute CS background with $f = 200$ took around 7 minutes for each image on a single core of a 2.3 GHz CPU. We implemented the convolution mean filter operation using TensorFlow [30] to run on Tesla Titan X GPU, this reduced the operation time by $7\times$. We experimented with several values of f, $f = 200$ worked best for our experimentation here.

C. Performance using PADNet

Figure 7 illustrates the efficiency of using the peak area detection network for learning phase regions from raw 2D XRD patterns. For a thorough evaluation, we trained different models on the original raw 2D XRD images with and without background removal using the two methods MF and CS; hence, there are three types of training datasets - I, $I - MF$ and $I - CS$. To evaluate the efficiency of PADNet trained on different input types, we also evalaute them using the three types of inputs for each model - I, $I - MF$ and $I - CS$. Next, the peak detection component in the convolutional graph of each model can be either held constant or trained using backpropagation so that the network can learn the slope filter parameters itself. Therefore, Figure 7 demonstrates the performance of all types of models using different possible combinations of type of training data, type of test data and the configuration of slope filters.

For SLAC, we observe a cons istent performance across all input types used during evaluation for both types of configuration of slope filters and for all types of training data. The predictions made for input raw XRD pattern is completely independent of background removal which illustrates that the model can be directly used to predict the phase region labels from the raw input XRD pattern measured from experiment. The performance of the model slightly improves if the background is removed from the data.

We observe similar performance for Bruker; the performance is almost consistent for all types of test XRD patterns when the peak area detection component is held constant. If



(a) SLAC, Constant

(b) SLAC, Trained

(c) Bruker, Constant

(d) Bruker, Trained

Fig. 7: Performance with peak area detection network using a ten-fold cross validation.The uniform performance across all test input pattern types exhibits the efficacy of PADNet for phase region classification directly from raw 2D XRD pattern.

the peak area detection component is allowed to be trained, the performance is slightly lower for raw XRD patterns if the model is trained using XRD patterns with background removed.

The results illustrates that PADNet can be used for directly making prediction of phase region labels from the raw XRD patterns without any need for background removal. The performance is specially interesting in the case of highly irregular background present in the case of SLAC (Figure 1) where the performance is completely independent of the background removal process.

D. Comparison with Current Approach

We compared the performance of PADNet against the current approach of phase region classification using 1D XRD patterns. For dataset from Bruker, since there are two XRD patterns for different range of 2θ, we combined them together. Next, we subtracted the background from the 1D XRD patterns for both datasets using the envelope function in MATLAB as shown in Figure 1. Finally, we applied traditional machine learning approaches to both datasets.

We followed the recent approach of training an AdaBoost classifier from Bunn et al. [16]. We performed an elaborate grid search for hyperparameter tuning of AdaBoost. For AdaBoost classifier, we used Decision Tree Classifiers with varying depth from 2 to 10 as the estimator. For learning rate, we used the values in [1, 0.1, 0.001] and for number of estimators, we used [5, 10]. For SLAC, we obtained $34.78 \pm 14.04\%$ accuracy and for Bruker we obtained 83.80 ± 14.50 accuracy using a 10-fold cross validation. While using composition, we achieved an accuracy of $70.80 \pm 15.80\%$ on XRD patterns from SLAC and an accuracy of 84.03 ± 22.33 on XRD patterns from Bruker.

We also analyzed the performance of other types of classifiers such as Logistic Regression, Naive Bayes, DecisionTree Classifier and SGD Classifier on the two datasets with and without composition information. We performed extensive grid search for hyperparameter tuning for all of them. In general, the use of composition results into slight improvement of performance. All these models had slightly worse performance compared to AdaBoost on the two datasets, performing significantly poorly on dataset from SLAC compared to the dataset from Bruker. One reason behind this might be that, for SLAC, the 1D XRD input contains only 931 intensity values compared to XRD patterns from Bruker having 2501 intensity values. The more information present in the input, the better the models perform on the dataset. Another conjecture is that these models perform bad on SLAC because of high noise present in the raw XRD pattern which can lead to loss of information while converting them to 1D.

Our PADNet performs significantly better than the current approach of indexing using 1D XRD patterns on SLAC and slightly better than AdaBoost and other classifier on 1D XRD patterns from Bruker. PADNet also exhibits lower deviation in the performance which shows that it can make more robust predictions. The current approach analysis of 1D XRD patterns is a computationally intensive process which involves multiple steps- converting to 1D by integrating along 2θ axis,

background removal and comparison to reference database or applying a machine learning based predictive model. PADNet can provide a fast approach as the prediction takes less than one second (on a Tesla Titan X GPU) for directly predicting phase region from 2D XRD pattern.

V. Conclusions and Future Works

We designed a peak area detection network for predicting phase regions directly from raw 2D XRD patterns from real experiments. The classification results using the peak area detection network demonstrated their invariance to the presence of background in the input XRD pattern during evaluation. This illustrates that PADNet can be directly used to predict the phase regions from the raw 2D XRD patterns without any background removal almost without any impact in prediction performance. This is the first application of deep learning on the raw 2D XRD patterns from real experiments. Since deep learning works better with big training data, our approach should provide better performance if applied on larger datasets. It would hopefully pave the way for future works tapping the potential of deep learning for this and related problems. We hope this would foster the adoption of deep learning techniques for rapid and automated analysis of X-ray diffraction images, and more broadly in the field of materials science and imaging. There exists a lot of scope for future research to understand the efficacy of the proposed methods such as automating the filter size selection, evaluating these on other XRD patterns, and incorporating them in a real time system for XRD analysis.

Acknowledgment

This work is supported in part by the following grants: NIST award 70NANB14H012, NSF award CCF-1409601; DOE awards DE-SC0014330, DE-SC0019358.

References

[1] M. M. Woolfson and M. M. Woolfson, *An introduction to X-ray crystallography*. Cambridge University Press, 1997.

[2] H. P. Klug and L. E. Alexander, "X-ray diffraction procedures: for polycrystalline and amorphous materials," *X-Ray Diffraction Procedures: For Polycrystalline and Amorphous Materials, 2nd Edition, by Harold P. Klug, Leroy E. Alexander, pp. 992. ISBN 0-471-49369-4. Wiley-VCH, May 1974.*, p. 992, 1974.

[3] D. M. Moore and R. C. Reynolds, *X-ray Diffraction and the Identification and Analysis of Clay Minerals*. Oxford university press Oxford, 1989, vol. 322.

[4] D. L. Bish and J. E. Post, *Modern powder diffraction*. Mineralogical Society of America Washington, DC, 1989, vol. 20.

[5] B. Cullity, "Elements of xrd diffraction, addition-wesley," *Reading, MA*, 1978.

[6] J.-S. Chung and G. E. Ice, "Automated indexing for texture and strain measurement with broad-bandpass x-ray microbeams," *Journal of applied physics*, vol. 86, no. 9, pp. 5249–5255, 1999.

[7] H. Koinuma and I. Takeuchi, "Combinatorial solid-state chemistry of inorganic materials," *Nature materials*, vol. 3, no. 7, p. 429, 2004.

[8] J. Gregoire, D. Van Campen, C. Miller, R. Jones, S. Suram, and A. Mehta, "High-throughput synchrotron x-ray diffraction for combinatorial phase mapping," *Journal of synchrotron radiation*, vol. 21, no. 6, pp. 1262–1268, 2014.

[9] "Mit cmse x-ray diffraction facility," 2019. [Online]. Available: http://prism.mit.edu/xray/training/analysis.html

[10] Y. M. Mos, A. C. Vermeulen, C. J. Buisman, and J. Weijma, "X-ray diffraction of iron containing samples: The importance of a suitable configuration," *Geomicrobiology Journal*, vol. 35, no. 6, pp. 511–517, 2018.

[11] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Analytical chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[12] R. E. Dinnebier, "Fwhm optimized polynomial smoothing filters: A practical approach," *Powder Diffraction*, vol. 18, no. 3, pp. 199–204, 2003.

[13] S. Brückner, "Estimation of the background in powder diffraction patterns through a robust smoothing procedure," *Journal of Applied Crystallography*, vol. 33, no. 3-2, pp. 977–979, 2000.

[14] M. Tatlier, "Artificial neural network methods for the prediction of framework crystal structures of zeolites from xrd data," *Neural Computing and Applications*, vol. 20, no. 3, pp. 365–371, 2011.

[15] C. J. Gilmore, G. Barr, and J. Paisley, "High-throughput powder diffraction. i. a new approach to qualitative and quantitative powder diffraction pattern analysis using full pattern profiles," *Journal of applied crystallography*, vol. 37, no. 2, pp. 231–242, 2004.

[16] J. K. Bunn, J. Hu, and J. R. Hattrick-Simpers, "Semi-supervised approach to phase identification from combinatorial sample diffraction patterns," *JOM*, vol. 68, no. 8, pp. 2116–2125, 2016.

[17] J. R. Hattrick-Simpers, J. M. Gregoire, and A. G. Kusne, "Perspective: Composition–structure–property mapping in high-throughput experiments: Turning data into knowledge," *APL Materials*, vol. 4, no. 5, p. 053211, 2016.

[18] 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*, vol. 3, no. 1, p. 4, 2017.

[19] W. B. Park, J. Chung, J. Jung, K. Sohn, S. P. Singh, M. Pyo, N. Shin, and K.-S. Sohn, "Classification of crystal structure using a convolutional neural network," *IUCrJ*, vol. 4, no. 4, pp. 486–494, 2017.

[20] X-ray diffractometer — nist.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[22] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning." in *AAAI*, 2017, pp. 4278–4284.

[23] D. Jha, S. Singh, R. Al-Bahrani, W.-k. Liao, A. Choudhary, M. De Graef, and A. Agrawal, "Extracting grain orientations from ebsd patterns of polycrystalline materials using convolutional neural networks," *Microscopy and Microanalysis*, vol. 24, no. 5, pp. 497–502, 2018.

[24] G. Bergerhoff, R. Hundt, R. Sievers, and I. Brown, "The inorganic crystal structure data base," *Journal of chemical information and computer sciences*, vol. 23, no. 2, pp. 66–69, 1983.

[25] D. Jha, L. Ward, A. Paul, W.-k. Liao, A. Choudhary, C. Wolverton, and A. Agrawal, "Elemnet: Deep learning the chemistry of materials from only elemental composition," *Scientific reports*, vol. 8, no. 1, p. 17593, 2018.

[26] M. A. Bauer, A. Biem, S. McIntyre, and Y. Xie, "A pipelining implementation for parsing x-ray diffraction source data and removing the background noise," in *Journal of Physics: Conference Series*, vol. 256, no. 1. IOP Publishing, 2010, p. 012017.

[27] M. A. Bauer, A. Biem, S. McIntyre, N. Tamura, and Y. Xie, "High-performance parallel and stream processing of x-ray microdiffraction data on multicores," in *Journal of Physics: Conference Series*, vol. 341, no. 1. IOP Publishing, 2012, p. 012025.

[28] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.

[29] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[30] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[31] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.