MatPhase: Material phase prediction for Li-ion Battery Reconstruction using Hierarchical Curriculum Learning

[1] Anika Tabassum, [2] Nikhil Muralidhar, [1] Ramakrishnan Kannan, [1] Srikanth Allu
[1]*Oak Ridge National Laboratory*
[2] *Department of Computer Science, Stevens Institute of Technology*
Email: {tabassuma, allus , kannanr}@ornl.gov, nmurali1@stevens.edu

***Abstract*—Li-ion Batteries (LIB), one of the most efficient energy storage devices, are used extensively in many industrial applications. These batteries consist of electrodes that are put together with heterogeneous material compositions. Imaging data of these battery electrodes obtained from X-ray tomography can explain the distribution of material constituents and allow reconstructions to study electron transport pathways. Such reconstructions of material constituents help quantify various associated properties of electrodes (e.g., volume-specific surface area, porosity) which determine the performance of batteries. These images often suffer from low image contrast between multiple material constituents, hence making it difficult for humans to distinguish and characterize these constituents through visual inspection. A minor error in detecting distributions of the material constituents can lead to magnified errors in the calculated parameters of material properties (e.g., porosity). We present MatPhase, a novel hierarchical curriculum learning technique to address the complex task of estimating material constituent distribution in battery electrodes. MatPhase comprises three modules: (i) an uncertainty-aware global model trained to yield inferences conditioned upon global knowledge of material distribution, (ii) a local model to capture relatively more fine-grained (local) distributional signals, (iii) an aggregator model to appropriately fuse the local and global effects towards obtaining the final distribution. On average, MatPhase improves prediction up to 8.5% relative to other sophisticated modeling pipelines and state-of-the-arts (SOTA) object detection models employed in the performance comparison.**

***Index Terms*—Battery Segmentation, Curriculum Learning, IDK Classification, Uncertainty quantification**

I. Introduction

Lithium Ion batteries (LIB) are an advanced energy storage technology employed across many highly impactful mechanical and technological applications ranging from electric vehicles to smartphones and laptops. A critical facet of the functioning of LIB is the two electrodes (i.e., anode and cathode) that enable the storage and flow of electricity. These electrodes are typically composite materials consisting of electro-chemically active material particles (e.g., Li, Ni, Mn) and a polymeric binder (e.g., C) to bind the active material particles onto a substrate (typically a thin aluminum sheet).

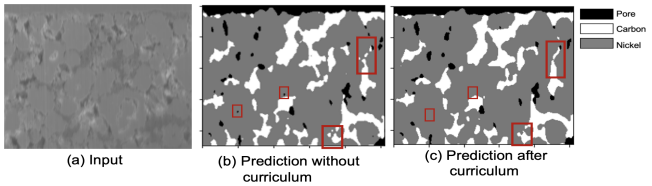

Fig. 1: Motivation for MatPhase. (a) shows a micro-scale tomography image of a portion of an electrode sample. The varying intensities over the image represent possible phase transitions among material constituents that humans cannot easily perceive. (b) Heatmap of predictions showing material constituents at the corresponding pixels obtained through a popular image segmentation model U-Net [1] (grey, white, and black pixels denote nickel, carbon, and pore, respectively). (c) Shows the final output after adopting our proposed hierarchical curriculum learning. The red rectangles indicate prediction differences between (b) and (c). Predictions at (c) are more smooth and free of artifacts.

The microstructure of these composite electrode coatings significantly influences the LIB performance. Specifically, homogeneity of the coating *thickness* across the entire electrode ensures optimal electrical conductivity. Since the primary purpose of the *binder* used in electrodes is to hold the active material in the electrodes, the distribution of the binder throughout the electrode coating determines the thickness of the coating. The lack of homogeneous distribution of the binder (and thereby the active material) can also lead to the absence of electrode coating in certain regions termed *pores*, causing degradation in battery performance.

Many approaches have been proposed to account for the binder distribution but are all heuristic in nature. Despite the advancements in imaging (e.g., X-ray microtomography), there has been noticeable difficulty in resolving the location of all phases (i.e., active materials, binder, pores) within the microstructure [2]. Since, existing imaging tools (at micro-scale) cannot distinguish spatial distributions due to the complex microstructures and various compositional materials of electrodes, identifying material constituents and phase transition in the microstructure of electrode coating from low contrast

imaging (i.e., X-ray microtomography) becomes challenging.

Fig. 1(a) shows an example of a low contrast micro-scale image of an electrode. Every pixel of the image either contains carbon binder (C) or active particles, e.g., nickel (Ni), or a pore. Clearly, it is impossible to distinguish these material constituents and phase transitions between two constituents by the human eye. Thus, we leverage the power of machine learning (ML) techniques to address the problem of material phase prediction of electrode coating from micro-scale images. Specifically, we develop an ML framework to label each pixel in a micro-scale image with its constituent phase (i.e., C, Ni, or Pore). We treat this problem of pixel-wise constituent phase identification as an image segmentation task. Popular architectures (e.g., *U-Net* [1]) do not translate well to the current task. Fig. 1(b) shows the predicted material constituents for each pixel obtained by *U-Net*. In the figure, gray pixels correspond to the active material nickel (Ni), white pixels signify the carbon binder (C), and black pixels represent a pore. Although U-net effectively captures the global properties of material constitution, it fails to predict the phase boundaries (i.e., transition between two material phases) accurately. Additionally, the predictions from U-Net also lead to *artifacts* (i.e., small regions of discontinuous predictions) especially at the phase boundary (see Fig. 1(b) - red rectangles are used to highlight artifacts). Accurate prediction of the material constitution at phase boundaries is crucial as this influences the quantification of various physical properties of the electrode by domain experts. Even a minor error in predicting these phase boundaries translates to a high error in calculating the physical properties of the material.

We posit that the failure of traditional segmentation models to represent the microstructure accurately is two-fold: (a) the inability of *flat* modeling architectures to learn at different granularity across different regions of the microstructure. For example, predictions conditioned upon a coarse-grained representation might suffice in a homogeneous region of the microstructure. In contrast, a less homogeneous region (i.e., a region concentrated with many constituents) requires a relatively more fine-grained representation learning ability (b) traditional segmentation models do not possess the ability to accurately distinguish between tasks of differing degrees of difficulty (e.g., predicting pixels at phase transitions vs. predicting pixels within a single phase).

We propose a novel deep-learning framework, Material Phase Prediction (MatPhase) for LIB microstructure reconstruction. To address failure mode (a), we introduce two architectures that are designed to learn representations at varying complexities (i.e., a coarse-grained model and a fine-grained model). To address the failure mode (b), we develop a hierarchical curriculum learning framework to automatically distinguish between regions with low and high degrees of prediction difficulty. Fig. 1(c) shows our final predicted result after incorporating curriculum learning. Comparing with the pixels within corresponding red rectangles (e.g., leftmost rectangle) in Fig. 1(b), the small black dot of pore is not seen in the corresponding rectangle in Fig. 1(c). Overall, our predictions are smoother around phase boundaries without the artifacts that are apparent without using the curriculum.

Contribution: To the best of our knowledge, there is no existing technique to identify the complex phase boundaries from low contrast tomography images of electrodes, and MatPhase is the first attempt to address this challenge. Our contributions are: (i) We develop a novel uncertainty-aware framework MatPhase to predict electrode material phases from low-contrast micro-scale image samples obtained through X-ray tomography. (ii) First time in the literature, we introduce a *hierarchical curriculum learning* technique to predict the fine-grained material phase boundaries from a low-contrast image. (iii) To aid smooth fine-grained phase boundary prediction, we build a multi-class classifier which can capture both local and global representations. (iv) Through rigorous experiments, we show that our framework yield better predictions compared to any non-trivial competitors.

II. Relevant Work

Recently, multiple deep-learning based image segmentation models have been proposed for object or boundary detection in computer vision applications [1], [3]–[6]. Howard et al. proposed a boundary detection model by addressing the spatially varying intensities in velocity shock wave image data [7]. Researchers viewed the problem of identifying material constituents from electrode samples as object detection [8]. Errors induced during data binarization and how that would translate into uncertainties in the calculated parameters, such as porosity, specific surface area was demonstrated by Pietsch et al. [2]. Labonte et al. proposed to to quantify uncertainty to the segmentation for electrode material [9].

Curriculum learning is a strategy to train a machine learning model ordered from easier dataset to harder dataset mimicking the learning order as human to solve different tasks [10]. Existing strategies of curriculum involve Gaussian Kernel-base feature embedding [11], hierarchical loss function [12], or style transfer module [13]. To accelerate inference along with prediction accuracy, Wang et al. proposed a framework 'I don't know' (IDK) that systematically selects a subset of instances of a pre-trained deep learning model [14]. Their proposed IDK framework is a rule-based function that selects instances if the entropy loss of the pre-trained model is within a user-defined threshold. For MatPhase no external module is employed and curriculum is learnt implicitly as part of the framework . To the best of our knowledge, our proposed hierarchical curriculum learning has not been adopted in any application for the task of material phase prediction.

III. Background and Preliminaries

We state our problem as a downstream task of material phase prediction. Each pixel of a low contrast micro-scale image of an electrode is classified as one of 'n' individual material constituents.

Problem III.1 (Material Phase Prediction)**.** ***Given****, a set of labeled datasets* $\mathcal{D} = (\mathcal{X}, \mathcal{Y})_{i=1}^{N}$*, and a set of classes* $\mathcal{C}$*. Each* $\mathbf{X}_i \in \mathcal{X}$ *is an image of* $n \times m$ *pixels.* $\mathbf{Y}_i \in \mathcal{Y}$ *consists of*

$n \times m$ *instances*[1] *for the corresponding pixels in* $\mathbf{X_i}$*, where each instance belongs to a class* $c \in C$*.* ***Predict****,* $\hat{\mathbf{Y}}$ *for an unlabeled* $\mathbf{X}$*.*

Hierarchical Curriculum Learning: A learning task often requires a set of multiple curricula $\mathcal{S}$ (i.e., ordered learning strategies) to enable a model to learn effective representations. It is possible that these multiple curricula could each govern a subset of the overall dataset $\mathcal{D}$. Additionally, the set of curricula $\{s_1, .., s_n\} \in \mathcal{S}$ themselves have (latent) *dependencies*. In such a scenario, we introduce the paradigm of *hierarchical curriculum learning* (HCL) as a solution to address learning in the aforementioned context.

We define the goal of hierarchical curriculum learning (HCL) as being the implementation of multiple complementary curriculum strategies in a hierarchical fashion to order data by relative *hardness* (i.e., from easy to hard). Specifically, assume $\mathcal{S} = \{s_1, s_2\}$ where $|\mathcal{S}|$ (i.e., number of curricula) may be derived by expert knowledge. Further, let us assume $h(s_1) < h(s_2)$ where $h(\cdot)$ is a hardness measure also specified by a domain expert. s_1 may for instance be considered the curriculum geared toward learning global properties of the problem while s_2 may be the curriculum for learning (nuanced) local variations (thus $h(s_1) < h(s_2)$).

In this context, automatic-HCL comprises a learning framework which automatically learns to address the task of interest by first *partitioning* $\mathcal{D}$ into $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2\}$ such that $\mathcal{D}_1 \subseteq \mathcal{D}$, $\mathcal{D}_2 \subseteq \mathcal{D}$, $\mathcal{D}_1 \cap \mathcal{D}_2 = \emptyset$ and $h(\mathcal{D}_1) < h(\mathcal{D}_2)$. The partitioning step may be viewed as a *coarse-grained* curriculum that initially restricts learning to the easier instances in the dataset. This partitioning step not only learns effective representations for $\mathcal{D}_1$ but is also designed to yield *fine-grained* curricula for training models on the harder $\mathcal{D}_2$ data. Hence, the *coarse* and *fine-grained* curricula together comprise an automatic-HCL framework for effective learning of the task of interest.

IV. Our Framework

This section describes our framework MatPhase, which predicts material phases from low-contrast images not easily perceived by humans. MatPhase first predicts the material phases through a global model (learning global data features). This is followed by a data partitioning step governed by the performance of global model on $\mathcal{D}$, which forms the coarse-grained curriculum. The partitioning step concludes with the global model which learns the properties of easier data partition ($\mathcal{D}_1$), while yieliding a measure of hardness for $\mathcal{D}_2$, which forms the fine-grained curriculum. We design a local model to be trained on $\mathcal{D}_2$, governed by the fine-grained curriculum to learn local data features.

A. Uncertainty-aware Global Segmentation Model (UGSM)

We initially view Problem III.1 as an image segmentation task using U-Net [1] due to its precise object localization and adaptability to be trained with a few samples. However, to achieve smooth prediction (as mentioned in Sec. I Fig. 1), we pursue U-Net as a global model instead. Our idea is to leverage a course-grained curriculum which learns the global representations of the model. Then, partition the incorrectly predicted instances (i.e., harder data $\mathcal{D}_2$) from the correctly predicted instances ($\mathcal{D}_1$). Hence, we aim for an uncertainty-aware global segmentation model (*UGSM*). Our assumption **A1** is: high uncertainty tends to incorrect predictions. We pursue Monte-Carlo Dropout (MCD) based uncertainty estimation due to its easy adaptability by any deep-learning model [15]. For *UGSM*, we incorporate a dropout layer (dropout probability 0.5) after each convolution block of *U-Net* [16]. During inference, *UGSM* samples T predictions. For each input $\mathbf{X}$, the final output of UGSM consists of two pairs: (i) predicted class per pixel, i.e, $\hat{\mathbf{Y}}_G \in \mathcal{R}^{n \times m}$, (ii) an uncertainty map $\mathbf{U}_G \in \mathcal{R}^{n \times m}$. Suppose, *UGSM* samples a set of logits $\mathcal{P} = \{\mathbf{P}^i\}_{i=1}^{T}$, where $\mathbf{P}^i \in \mathcal{R}^{|C| \times n \times m}$. Now if $\mathbf{F} = \frac{1}{T} \sum_{\mathbf{P}^i \in \mathcal{P}} \mathbf{P}^i$. $\hat{\mathbf{Y}}_G = \arg_{\max} \text{softmax}(\mathbf{F})$. For uncertainty map $\mathbf{U}_G$, we compute the entropy each instance. Suppose, p_c^i is the probability that UGSM labels instance i as class c. We calculate $u^i = -\sum_{c \in C} p_c log(p_c)$.

B. Automatic Hierarchical Curriculum Learning (HCL)

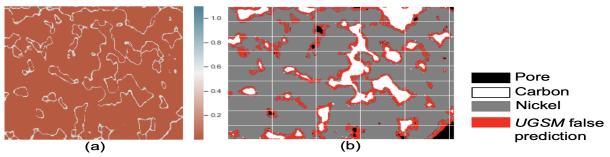

Fig. 2: Motivation for coarse-grained HCL: (a) Uncertainty map by UGSM (b) Heatmap of the material phases of the input of Fig. (a). Red pixels are incorrect predictions of UGSM. All other pixels are correct predictions (black indicates pore, white as C, and grey as Ni). The density of red instances in Fig. (b) is higher than the pixels with high uncertainty value in Fig. (a), indicating that UGSM also assigns low uncertainties to a significant portion of *incorrect* predictions invalidating assumption A1.

As mentioned in Sec. III, our automatic *HCL* consists of coarse-grained curriculum s_1 and a fine-grained curriculum s_2. We view the incorrectly predicted instances of UGSM as harder data $\mathcal{D}_2$. In this context, our curriculum s_1 is the process of automatically partitioning $\mathcal{D}$ into $\mathcal{D}_1$ and $\mathcal{D}_2$, where $\mathcal{D}_1$ is the set of correctly predicted instances of UGSM. If assumption **A1** was to hold, s_1 could be a simple rule-based classifier with a pre-set uncertainty threshold that identifies $\mathcal{D}_2$ using the uncertainty map $\mathbf{U}_G$ obtained by UGSM. Fig. 2 (a) shows example of an uncertainty map of UGSM for an image $\mathbf{X}$. The high values (white colored pixels) denote high uncertainty. We observe that majority of the instances with high uncertainty lie at the *phase boundaries*. Fig. 2(b) shows the heatmap of the ground-truth label (material phases) for the same input $\mathbf{X}$. The red pixels represent *incorrect* predictions of UGSM. We notice that the number of incorrect classifications (Fig. 2(b) red pixels) is much higher than the number of instances with high uncertainty (Fig. 2(a)). This implies that

[1]We employ the word 'pixel' and 'instance' interchangeably throughout the paper

UGSM assigns low uncertainties to a significant portion of instances that it predicts incorrectly. This implies that our simplistic assumption **A1** is a necessary but not a sufficient condition for identifying incorrect predictions in the context of the current model pipeline. Our goal through curriculum s_1 is hence to segregate (i.e., coarse-grained curriculum) all these red instances as $\mathcal{D}_2$.

Coarse-grained curriculum (HCL-IDK): We intend to automatically learn the representations of $\mathcal{D}_1$ and $\mathcal{D}_2$ by UGSM for partitioning. Fig. 3 shows the heatmaps of the UGSM logits, $\mathbf{F}$ of each class $c \in \mathcal{C}$ (pore, carbon, and nickel) for the input $\mathbf{X}$ in Fig. 2. For e.g., consider Fig. 3(b) which is the heatmap of the logits of class carbon. The logits for all the pixels that contains class carbon (C) bear values within range $(2, 6)$. Also, the logits of the pixels across the C boundaries bear similar values, i.e., all values within range $(0.5, 1)$, (pixels colored in white). Comparing with Fig. 2(b), we see these white pixels or pixels of carbon boundaries (at Fig. 3(b)) match with the incorrectly predicted red pixels around carbon (Fig. 2(b)). We observe similar behavior for the heatmaps of logits of class pore and nickel. Hence, we can make an assumption **A2**: incorrectly predicted instances bear similar weights in the UGSM logits. For **A2**, we motivate to view the incorrect instances as "I don't know" (*IDK*) class. We leverage a Feed-forward classifier for s_1 to identify the *IDK* instances. The goal of the classifier is to learn an effective latent representations for *IDK* through UGSM features, i.e. F and $\mathbf{U}_G$. For each instance j, the input of the classifier is a vector $\mathbf{idk}^j \in \mathcal{R}^4$ which consists of the features, i.e., $F^j \in \mathcal{R}^3$ and u^j. Let, $\mathcal{Y}_{IDK}$ be the set of all instances that classifier identifies as *IDK* and $\mathcal{Y}'_{IDK}$ be the complement of $\mathcal{Y}_{IDK}$. Hence, our easy data $\mathcal{D}_1$ comprises of the instances present in $\mathcal{Y}'_{IDK}$, and hard data $\mathcal{D}_2$ comprises of the instances in $\mathcal{Y}_{IDK}$.

Fine grained curriculum (HCL-USI): For learning the set of hard instances in $\mathcal{D}_2$, we adopt curriculum s_2. Here, s_2 is a fine-grained curriculum that quantifies the hardness of every instance in $\mathcal{D}_2$. We employ the uncertainty map $\mathbf{U}_G$ of UGSM as the hardness function. Hence, our s_2 curriculum progresses by training with increasingly hard instances governed by the instance-wise uncertainty of UGSM (HCL-USI).

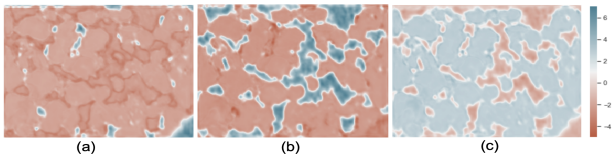

Fig. 3: Heatmap of the prediction logits of UGSM for the same input image of Fig. 2. Each Fig. corresponds to the logits of a class (a) Pore, (b) C, and (c) Ni. These heatmaps have high logit values (blue pixels) that corresponds to the predicted pixel class. Each heatmap also shows similar weights (pixels in white) around the class boundaries. These density of these white pixels mostly match with the incorrectly predicted red pixels in Fig. 2 (b). This motivates to view the incorrectly predicted pixels as "I dont Know" class.

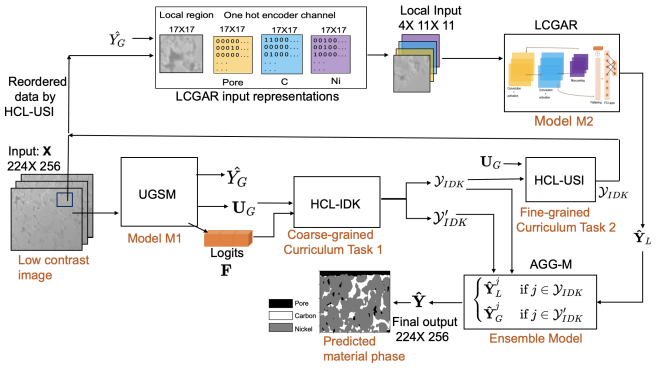

Fig. 4: Overview of our proposed framework MatPhase. HCL-IDK and HCL-USI) is designed to capture both global and local representations for smooth phase predictions.

C. Local Classification with Global Annotated Region (LCGAR)

The main goal of LCGAR is to provide smooth predictions for the *IDK* instances in $\mathcal{D}_2$. We aim to capture both the local and global features of the target instance for this task. We motivate for a classifier that accepts multiple features, i.e., (i) a local image of $\mathbf{X}$ surrounding the target *IDK* instance, (ii) predicted class labels of UGSM, i.e., $\hat{\mathbf{Y}}_G$, (iii) UGSM logits, i.e., $\mathbf{F}^j$ for the target instance j. By training our model with global features from UGSM, we intend to learn latent local representations of target instances. The architecture of LCGAR is a CNN. To capture rich information about the arrangement of the surrounding for the target pixel of interest, we concatenate UGSM logits (F^j) after flattening. We use LeakyRelu as the activation.

D. MatPhase Framework

Fig. 4 provides an overview of MatPhase. We first train the uncertainty-aware global segmentation model (UGSM). For each image, $\mathbf{X} \in \mathcal{X}$, UGSM provides a pixel-wise prediction map ($\hat{\mathbf{Y}}_G$) and an uncertainty map ($\mathbf{U}_G$) associated with the prediction. Our coarse-grained curriculum HCL-IDK is trained to partition all the instances in $\hat{\mathbf{Y}}_G$ into two sets, $\mathcal{Y}_{IDK}$ and $\mathcal{Y}'_{IDK}$. Next, the fine-grained curriculum HCL-USI orders $\mathcal{Y}_{IDK}$ by increasing prediction uncertainty obtained from the UGSM model. Next, our local model LCGAR is trained on both the global and local features to learn local context of $\mathcal{Y}_{IDK}$ to yield corresponding predictions $\hat{\mathbf{Y}}_L$. Finally, the ensemble model *AGG-M* aggregates $\hat{\mathbf{Y}}_G$ and $\hat{\mathbf{Y}}_L$ to provide spatially smoother predictions $\hat{\mathbf{Y}}$. Here, $\hat{\mathbf{Y}} = \hat{\mathbf{Y}}^j_G$, if $j \in \mathcal{Y}_{IDK}$, else $\hat{\mathbf{Y}} = \hat{\mathbf{Y}}^j_L$.

V. Experiments

Our framework is implemented in PyTorch. All models are trained using 3 V100 Tesla GPUs. For every X-Ray Computed Tomography (XCT) imaging of electrode cross-sections, we obtain the ground-truth labels by utilizing cross-sectioning with Focused-Ion beam (FIB/SEM) experiments [17]. Code and datasets are available for research purposes [2].

[2]Code repository: github.com/srikanthallu/BatteryAnalytics

Research Questions: Our goal is to demonstrate that MatPhase can capture better predictions than any other state-of-the-art (SOTA) image segmentation models. We are interested to analyze whether our automatic HCL framework is useful for modeling smooth phase predictions. Specifically we want to address: **Q1.** Is our automatic curriculum learning (HCL) effective for modeling phase boundaries? **Q2.** Does incorporating global information aid local modeling? **Q3.** How does MatPhase predict over SOTA? **Q1** and **Q2** are aligned with justifying the effectiveness of MatPhase. With **Q3**, we plan to showcase the performance of MatPhase comparing with SOTA image segmentation models.

Measure of success: We evaluate the quality of MatPhase predictions on a downstream task of Problem III.1 for low contrast image data. We choose four evaluation metrics: (i) F1-score. (ii) *Pixel accuracy (ACC)* for k best performing and worst performing predictions to evaluate smoothness of material phase predictions. (iii) *Mean intersection over union (mIU)* from the standard practice of image segmentation models [4], [5]. (iv) *Frequency weighted intersection over union (fIU)* to evaluate in presence of class imbalance from the standard practice for image segmentation models [5].

Baselines: We compare MatPhase with SOTA image segmentation models. Specifically, to provide pixel to pixel predictions and to compare with different variants of UGSM, we use (1) DeepLabV3 [18] (2) MANet [19](3) *FCN* [5], (4) *SegNet* [4], (5) *U-Net* [1], (6) *U-Net++* [20]. (7) *MCD-U-Net* [16]: Monte-Carlo dropout based U-Net, similar to our global model UGSM. (8) *Local-U-Net*: Replace our local model LCGAR with a U-Net model [1]. For training, we mask the cross-entropy loss values (with zeros) for the instances that are not classified by HCL-IDK. (9) *ResNet-18* [21]: Our aim is to check with a pixel by pixel classifier model using same the input representation of LCGAR architecture. We leverage a *ResNet-18* model, setting the bounded local region $h = 11$ for every instance in the image. Note that, we also try with ResNet-34, ResNet-50 and only show the record which yield the best result. (10) *Adapted-LCGAR*: Predict every instances of image with pre-trained LCGAR.

A. Effectiveness (Q1-Q2)

Next, we aim to demonstrate the effectiveness of MatPhase, by analyzing the importance of (i) coarse-grained (HCL-IDK) and fine-grained (HCL-USI) curriculum of HCL, (ii) input representations of LCGAR, which incorporates global data features of to aid local modeling.

Effectiveness of HCL: For **Q1**, we want to evaluate the importance of curriculum learning. Hence, we describe the models varying the implementation of HCL learning strategies: (i) *UGSM* : We obtain the predictions from our pre-trained UGSM without any partitioning steps governed by curriculum HCL-IDK. (ii) *MatPhase w/o HCL-USI* : The model is trained directly using the hard data partitioned by HCL-IDK, which further is not learnt by HCL-USI. For testing, we replace our local model with *MatPhase w/o HCL-USI* and use the same framework MatPhase.

TABLE I: Effectiveness of HCL for different variants of MatPhase in terms of F1-score per class, accuracy (%) on top 5 best performing, and lowest performing test sets, *mIU*, and *fIU*. For all metrics, high score yields the better result. MatPhase outperforms all the models (best values in bold).

Model	F1	ACC(%) Best 5	ACC(%) Lowest 5	mIU	fIU
UGSM	Pore: **0.86** C: 0.82 Ni: 0.94	91.5 ± 0.04	90.5 ± 0.6	0.77	0.844
MatPhase w/o HCL-USI	Pore: **0.86** C: **0.85** Ni: **0.95**	**93.2** ± .07	91.7 ± 1.7	**0.803**	0.870
MatPhase	Pore: **0.86** C: **0.85** Ni: **0.95**	**93.2** ± 0.02	**91.9** ± 1.3	0.802	**0.871**

We compare the performance of MatPhase, which is governed by both the curriculum HCL-IDK and HCL-USI against the models mentioned above in Table I. We evaluate multiple metrics, mean F1-score per class across all the test datasets, average pixel accuracy rate of best 5 and worst 5 predictions, and mean IoU. For accuracy, the values ($\pm$) denote the percentage of deviation among the 5 predictions. We observe, MatPhase outperforms *UGSM* across all metrics.

Effectiveness of LCGAR input: For **Q2**, our goal is to capture how incorporating global features of UGSM to LCGAR aid modeling predictions. We use different variants of LCGAR varying the input representations. We train each model based on the same target instances used in MatPhase, only changing the input representation. (i) *Only-Image*: Consider LCGAR input as local input image surrounding target instances as mentioned in Sec. IV-C. We do not consider UGSM logits $\mathbf{F}^j$. (ii) *Image+UGSM-Y*: Input includes input image and the UGSM predictions $\hat{\mathbf{Y}}_G$ surrounding the target instances. We do not consider UGSM logits $\mathbf{F}^j$. (iii) *Image+UGSM-Emb*: Consider LCGAR input as local input image surrounding target instances. Also include logits $\mathbf{F}^j$ of target instance j.

TABLE II: Effectiveness of incorporating global information for modeling LCGAR. Evaluation metrics analogous to Table I. MatPhase outperforms all the models.

Model	F1	ACC (%) Best 5	ACC (%) Lowest 5	mIU	fIU
Only-Image	Pore: 0.69 C: 0.7 Ni: 0.92	87.6 ± 0.08	82.2 ± 1.9	0.65	0.781
Image+ UGSM-Y	Pore: 0.84 C: 0.84 Ni: **0.95**	**93.3** ± 0.06	89.8 ± 0.028	0.79	0.867
Image + UGSM-Emb	Pore: 0.69 C: 0.7 Ni: 0.92	87.6 ± 0.08	82.2 ± 0.019	0.65	0.781
MatPhase	Pore: **0.86** C: **0.85** Ni: **0.95**	93.2 ± 0.02	**91.9** ± 1.3	**0.802**	**0.871**

Table II shows the performance of LCGAR for different

input representations. MatPhase uses all the global features as mentioned in Sec. IV-C. MatPhase clearly outperforms all the models. On average using global features, F1- score of MatPhase increase to 0.12 (0.17 for pore, 0.15 for C, and 0.03 for Ni) comparing with *Only-Image* which uses no global features. Comparing with *Image+UGSM-Y*, the low *mIU* and *fIU* indicates that MatPhase predicts better phase boundaries, hence lead to smooth prediction.

B. Comparison with baselines (Q3)

We show the performance of MatPhase against SOTA image segmentation models and some non-trivial competitors. Our goal is to demonstrate that achieving smooth predictions for material phases from low-contrast images is challenging and cannot be easily captured by SOTA.

TABLE III: Performance of MatPhase comparing with the baselines in terms of evaluation metrics analogous to Table I.

Model	F1			mIU	fIU
	Pore	C	Ni		
DeepLabV3 [18]	0.85	**0.86**	0.87	0.76	0.83
MANet [19]	0.81	0.79	0.93	0.74	0.82
FCN [5]	0.66	0.17	0.88	0.45	0.64
SegNet [4]	0.81	0.78	0.94	0.74	0.83
U-Net [1]	0.85	0.82	0.94	0.77	0.84
U-Net++ [20]	0.76	0.53	0.78	0.54	0.59
MCD-U-Net [16]	**0.86**	0.82	0.94	0.78	0.84
Local-U-Net	**0.86**	0.82	0.94	0.78	0.84
ResNet-18 [21]	**0.86**	0.83	**0.95**	0.79	0.86
Adapted-LCGAR	0.85	0.84	**0.95**	0.79	0.86
MatPhase	**0.86**	0.85	**0.95**	**0.80**	**0.87**

Table III shows MatPhase performance compared with baselines (see Sec. V). We observe MatPhase outperforms all the baselines. On average, our F1-score improves 0.5 for pore, 0.14 for C, and 0.2 for Ni. w.r.t. *mIU* and *fIU*, on average MatPhase performance improves 8.5% and 6% than the baselines. The best and the lowest accuracy (*ACC* $(\%)$) values are in the appendix[3]. Results indicate the effectiveness of hierarchical curriculum learning (HCL) and incorporating local and global information for predicting target instances as they provide quality predictions (smooth and free of *artifacts*).

VI. Discussion and Conclusion

With the collaboration between computer scientists and battery researchers, for the first time, we present MatPhase, a novel framework to predict material phases, especially phase boundaries, from low contrast images of an electrode. Our extensive study against multiple non-trivial competitors (deep-learning-based models) and SOTA show that, on average, MatPhase increases the performance by 8.5%. We also show that incorporating HCL and the local model together, yields a maximum performance improvement of 7% overall. Our framework is practically useful to battery researchers for understanding electrode material phase distribution from noisy data. We envision using the framework to study the cross-sections of cycled electrodes and understand various degradation mechanisms that impact the loss of capacities.

Acknowledgements: This work is partially supported by the U.S. DOE, DOE Office of Electricity Dr. Imre Gyuk, Office of Science, Office of ASCR, SciDAC program, under the "RAPIDS Institute". This research used resources of the ORNL CADES, which is supported by the DOE Office of Science DE-AC05-00OR22725.

[3] Appendix:github.com/srikanthallu/BatteryAnalytics

References

[1] O. Ronneberger *et al.*, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, ser. LNCS, vol. 9351. Springer, 2015, pp. 234–241.

[2] P. Pietsch *et al.*, "Determining the uncertainty in microstructural parameters extracted from tomographic data," *Sustainable Energy & Fuels*, vol. 2, no. 3, pp. 598–605, 2018.

[3] K. He *et al.*, "Mask r-cnn," in *Proceedings of the IEEE ICCV*, 2017, pp. 2961–2969.

[4] V. Badrinarayanan *et al.*, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[5] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE CVPR*, 2015, pp. 3431–3440.

[6] Z. Jiang *et al.*, "Machine-learning-revealed statistics of the particle-carbon/binder detachment in lithium-ion battery cathodes," *Nature communications*, vol. 11, no. 1, pp. 1–9, 2020.

[7] M. Howard *et al.*, "A locally adapting technique for boundary detection using image segmentation," *arXiv preprint arXiv:1707.09030*, 2017.

[8] S. R. Daemi *et al.*, "Visualizing the carbon binder phase of battery electrodes in three dimensions," *ACS Applied Energy Materials*, vol. 1, no. 8, pp. 3702–3710, 2018.

[9] T. M. LaBonte *et al.*, "We know where we don't know: 3d bayesian cnns for uncertainty quantification of binary segmentations for material simulations." Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), Tech. Rep., 2020.

[10] X. Wang *et al.*, "A survey on curriculum learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[11] S. Sinha *et al.*, "Curriculum by smoothing," *Proc. of NeurIPS*, 2020.

[12] P. Goyal and S. Ghosh, "Hierarchical class-based curriculum loss," 2021.

[13] Z. Liu *et al.*, "Style curriculum learning for robust medical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2021, pp. 451–460.

[14] X. Wang *et al.*, "Idk cascades: Fast deep learning by learning not to overthink," *arXiv preprint arXiv:1706.00885*, 2017.

[15] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059.

[16] T. DeVries and G. W. Taylor, "Leveraging uncertainty estimates for predicting segmentation quality," *arXiv preprint arXiv:1807.00502*, 2018.

[17] S. Vierrath *et al.*, "Morphology of nanoporous carbon-binder domains in li-ion batteries—a fib-sem study," *Electrochemistry Communications*, vol. 60, pp. 176–179, 2015.

[18] L.-C. Florian and S. H. Adam, "Rethinking atrous convolution for semantic image segmentation," in *Conference on CVPR. IEEE/CVF*, vol. 6, 2017.

[19] T. Fan *et al.*, "Ma-net: A multi-scale attention network for liver and tumor segmentation," *IEEE Access*, vol. 8, pp. 179 656–179 665, 2020.

[20] Z. Zhou *et al.*, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep learning in medical image analysis and multimodal learning for clinical decision support*. Springer, 2018, pp. 3–11.

[21] K. He *et al.*, "Deep residual learning for image recognition," in *Proceedings of the IEEE CVPR*, 2016, pp. 770–778.