Novel Methodology for Vision Backbone Network Fine-Tuning and Continual Learning in Optical Inspection Tasks

Kody Haubeil[1], Tarek Yahia[1], Alexander Suer[1], David Siegel[2], Donald Davis[3], Xiaodong Jia[1]

[1] *Center for Intelligent Metrology & Sensing, Department of Mechanical & Materials Engineering, University of Cincinnati, Cincinnati, Ohio, 45219, United States of America*

haubeiky@mail.uc.edu, yahiatk@mail.uc.edu, suerad@mail.uc.edu, jiaxg@ucmail.uc.edu

[2] *Predictronics Corp., Norwood, Ohio, 45212, United States of America*

siegel@predictronics.com

[3] *Kent Displays, Inc, Kent, Ohio, 44240, United States of America*

ddavis@kentdisplays.com

ABSTRACT

This paper presents a novel methodology for fine-tuning vision backbones (or foundation models) and enabling continual learning to ensure reliable prediction performance after AI model deployment. The proposed network architecture is designed to adapt to new and previously unseen defect classes through few-shot learning. The methodology is demonstrated using two practical optical inspection applications: 1) Liquid crystal films produced on a high-throughput roll-to-roll manufacturing line and 2) Wafer map images from real-world semi-conductor manufacturing process. Experimental results show that the model achieves high prediction accuracy at test time and is capable of continuously learning from new data. Additionally, the model provides calibration scores, offering insights into prediction uncertainty. In summary, the proposed framework delivers a practical AI-based solution for optical inspection, combining high accuracy, interpretability, and continual learning. It eliminates the need for handcrafted image features and significantly reduces human intervention in defect detection and labeling.

1. INTRODUCTION

Optical inspection with automatic data handling plays a critical role in modern manufacturing by enabling rapid, accurate, and non-contact evaluation of product quality. It significantly enhances production efficiency by accurately detecting surface defects, dimensional inaccuracies, and assembly errors. Integrating automated data handling in optical inspection further streamlines the process by organizing and analyzing vast amounts of inspection data instantly, allowing for predictive maintenance, process optimization, and quality assurance. This not only reduces human error and inspection time but also supports data-driven decision-making, ultimately leading to higher product reliability, lower defect rates, and reduced manufacturing costs.

Traditional pattern recognition algorithms have been widely used in optical inspection systems for defect identification (Lo & Lin, 2024). These methods typically rely on handcrafted features extracted from image data, such as edges, textures, shapes, or statistical properties, using techniques like Sobel filters, Gabor wavelets, histogram of oriented gradients (HOG), and Hough transformation (Gonzales & Woods, 2002). Once features are extracted, classifiers such as k-nearest neighbors (KNN), support vector machines (SVM), and decision trees are employed to categorize image regions as defective or non-defective. While these algorithms are computationally efficient and interpretable, their performance is often limited by their dependence on domain-specific feature engineering and sensitivity to variations in lighting, scale, and orientation. As a result, although effective in controlled environments, traditional methods often struggle with generalizing to complex, real-world inspection tasks (Zhu et al., 2021) (Shih et al., 2023).

Recent breakthroughs in deep learning, particularly the emergence of pre-trained foundation models, have significantly advanced AI applications in optical inspection. These models, such as Vision Transformers (ViT) (Vaswani et al., 2017) or ResNet50 (Koonce, 2021), trained on massive and diverse datasets, provide powerful feature representations that can be adapted to various defect detection tasks with minimal labeled data. Their versatility enables robust performance across different materials, lighting conditions, and defect types, making AI-driven inspection more scalable and accessible. This shift

significantly reduces the need for extensive dataset curation and domain-specific training.

Despite recent advances, several technical challenges remain in leveraging AI for optical inspection tasks(Cui & Wang, 2022). One major issue is data shift, particularly covariate shift, where the input distribution $p(x)$ changes between training and deployment, potentially degrading model performance if the inference model $p(y/x)$ fixed. Another critical challenge is open-set recognition, where previously unseen defect types or new classes appear during testing, making it difficult for traditional models to generalize. Addressing these challenges requires robust domain adaptation techniques and models capable of detecting and adapting to novel or out-of-distribution inputs.

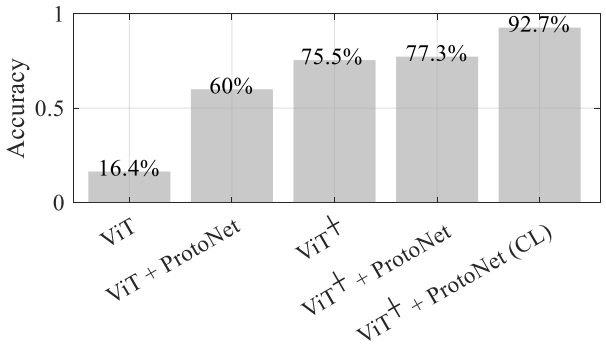


Figure 1. The overall prediction accuracy in case study 1 given by different deep neural network design. The ViT is the pre-trained foundation model without any fine-tuning. The ViT† is the updated backbone network by using supervised fine tuning. ProtoNet is a few-shot learner that is included to learning from new data continuously and adapt to unseen label classes. The results indicates that the proposed Backbone + Few-Shot Learner + Continual Learning method yield the best prediction performance.

To address data shifts in optical inspection, this paper proposes a novel methodology that integrates AI model fine-tuning during an offline phase and online learning after deployment. In the offline phase, the approach involves supervised fine-tuning (SFT) of a pre-trained backbone network, a few-shot learner to enable adaptation to new defect classes, along with a model calibration strategy to evaluate the reliability of the model's predicted probabilities across classes. During the online phase, prediction confidence is used to identify uncertain predictions, which are flagged for expert review and labeling. The newly labeled samples are then used to monitor model performance degradation and update the few-shot learner accordingly. This framework ensures the model can maintain high prediction accuracy while adapting to novel and unseen defects. Furthermore, the use of calibration scores enhances the interpretability of the AI system by indicating when predictions can be trusted. A new optical inspection application is utilized to test the proposed method, and the prediction results are shown in Figure 1.

The rest of the paper is organized as follows. Section 2 describes the inspection problem and the related works. Section 3 elaborates on the proposed methodology. Section 4 shows the results and discussions. Conclusions are given in Section 5.

2. Problem Statement and Related Works

2.1. Problem Statement

The objective of this study is to develop an AI-powered optical inspection system capable of automatically identifying part defects in Figure 2. Currently, human inspectors are required due to challenges related to lighting and complex data processing. Specifically, defective areas often occupy less than 2% of the entire part, making them difficult for AI algorithms to detect. Additionally, reflections and ambient light further complicate the detection of such small defects. Automating this inspection process is critical, as liquid crystal film products in Figure 2 are produced through high-volume roll-to-roll manufacturing, making it impractical to manually inspect every part. According to the manufacturer, the accuracy of human inspectors is approximately 80%, primarily due to fatigue and visual strain. Additionally, while classes 2 and 3 in Figure 2 illustrate common defects, there are also rare defect types in Table 1 that must be incorporated into the model. However, image data for these rare defects is extremely limited, presenting a challenge for effective training. Due to the disclosure restrictions, the images for those rare defects cannot be provided.

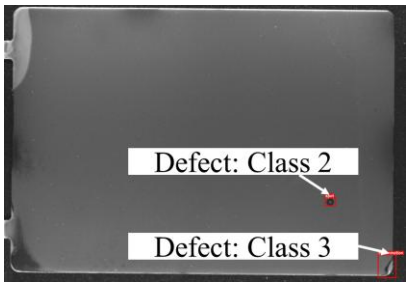


Figure 2. Defect classes on liquid crystal films

Table 1 The prediction class labels for this research task

Class 1	Healthy
Class 2	Common Defects (included in the training set)
Class 3	Common Defects (included in the training set)
Class 4	Rare Defects (Unseen to the training set)
Class 5	Rare Defects (Unseen to the training set)

To address these technical challenges, this study developed a machine vision system incorporating a state-of-the-art deep neural network architecture with the following capabilities. 1) Fine-grained sub-region scanning to

accurately identify small defective areas; 2) Adaptability to new and previously unseen defect classes, enabling the system to incorporate emerging defect types into the detection model—while classes 2 and 3 represent common defects, it is essential to account for novel classes that may appear during production; 3) Sustained prediction accuracy through online learning, allowing the model to continually improve and adapt in real time.

2.2. Literature Review

Backbone network retraining and fine-tuning are foundational strategies used for adapting pretrained vision models to application-specific tasks. Previous studies have shown that full backbone retraining performs well when large amounts of labeled data are available, especially in complex settings such as manufacturing (Kornblith et al., 2019). As an alternative, supervised fine-tuning (SFT) focuses on selectively updating specific layers or parameters of pretrained models to improve efficiency and avoid overfitting. This approach has gained popularity in recent literature, especially for tasks with limited data or deployment constraints. Layer-wise learning rate decay (LLRD) is a common approach, where earlier layers receive smaller gradient updates while deeper layers adapt more aggressively to the target task (Clark et al., 2020). Discriminative fine-tuning extends this idea by assigning different learning rates to different layers based on their sematic relevance to the downstream task (Howard & Ruder, 2018). Additional SFT methods such as BitFiT (Zaken et al., 2021) and LoRA (Low-Rank Adaptation) (Hu et al., 2022) have shown promising results in resource-constrained environments. Vision backbones including ViT-B/16 (Dosovitskiy et al., 2020), MobileNetv3 (Howard et al., 2019), and EfficientNet (Tan & Le, 2019) have been utilized in these settings due to their balance of accuracy and deployment efficiency. Collectively, these techniques have become state-of-the-art for deploying foundation models on application specific tasks with minimal overfitting and high generalization (Elharrouss et al., 2024).

Few-Shot Learning (FSL) has concurrently become an essential method to address novel class generalization with limited labeled samples. Prototypical networks remains a foundational component of FSL by computing class centroids in an embedding space and classifying queries via distance metrics (Snell et al., 2017). Matching Networks builds upon this by employing a learned attention mechanism over the support set, producing query-dependent embeddings through context-aware matching (Vinyals et al., 2016). Relation Networks introduce a learnable non-linear comparator that models interactions between support-query pairs, improving performance on more complex visual tasks (Sung et al., 2018). Gradient-based meta-learning methods such as Model-Agnostic Meta-Learning (MAML) take a different approach, training models through inner-loop optimization so they can quickly adapt to new tasks with just a few gradient steps (Finn et al., 2017). Variants like ANIL (Almost No Inner Loop) (Raghu et al., 2019) and Meta-SGD (Li et al., 2017) refine this framework by adjusting which layers or learning rates are trainable during the inner loop. More recent methods include Meta-Baseline, which applies a straightforward yet effective normalization and linear classifier strategy on top of a pretrained backbone (Chen et al., 2020), and FEAT (Few-shot Embedding Adaptation with Transformer), which uses self-attention to dynamically adapt the supper-query relationship in the embedding space (Ye et al., 2020). When integrated with a fine-tuned backbone, FSL methods benefit from more semantically aligned features, improving both accuracy and robustness in few-shot classification tasks under real-world constraints (Triantafillou et al., 2019).

In vision-based classification, modern deep neural networks often produce mis-calibrated probability estimates (Wenger et al., 2020) Standard post-hoc fixes apply simple parametric mappings. For example, Platt scaling (Böken, 2021) uses logistic regression on model scores and Guo et al. showed that a single ‘temperature’ scalar on the softmax scores can greatly improve calibration on vision tasks (Guo et al., 2017). These methods are easy to implement but are limited by their fixed functional forms. Bayesian Binning into Quantiles (BBQ) (Naeini et al., 2015) is a non-parametric Bayesian calibration approach. BBQ creates multiple equal frequency histogram-binning models of the classifier’s scores and scores each binning using a Bayesian likelihood. It then averages the calibrated probabilities from all these models according to their posterior weights. By marginalizing over bin configurations, BBQ produces a flexible calibration map that is not constrained to a sigmoid or scalar-temperature shape.

In prognostics and health monitoring tasks, new fault classes typically appear infrequently, and models must adapt on edge devices with very limited memory and compute. Under these constraints, conventional continual learning methods struggle. Replay-based methods become impractical due to storage, privacy, and scalability limits, and parameter-isolation approaches incur large model overhead and assume clear task delineation. Moreover, gradient-based fine-tuning for a few layers can impose nontrivial computation, making it poorly suited for real-time edge use (Chen et al., 2025). Accordingly, recent work favors a frozen-backbone strategy with lightweight adapters or small task-specific heads. For example, one can freeze a pretrained backbone and insert compact adapter modules or additive weight updates that are trained on the few new examples (Stein et al., 2025). This parameter-efficient design achieves near full-finetuning performance while keeping the bulk of the network fixed. In practice, it preserves the generic features of the backbone and adds only minimal new parameters, enabling few-shot adaptation of rare classes without overwriting previous knowledge.

3. PROPOSED METHODOLOGY

3.1. Overview of the Proposed Methodology

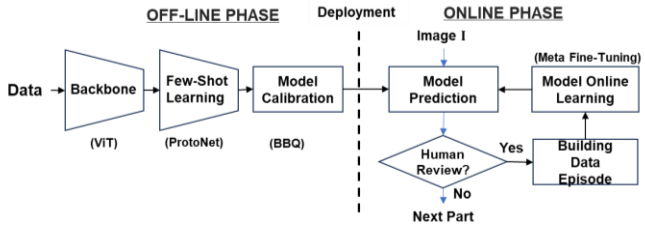


Figure 3. The proposed methodology

An overview of the proposed methodology is shown in Figure 3. The proposed methodology has an offline phase and an online phase. The offline phase focuses on getting a good prediction model using historical data. The online phase retains the model's performance after deployment by continuous learning from new data.

In the offline phase, the proposed network architecture consists of a pre-trained backbone, a few-shot learner, and a model calibration module. The pre-trained image backbone serves as a general-purpose feature encoder that transforms input images into compact feature vectors. In this study, we adopted the Vision Transformer (ViT) backbone, pre-trained on ImageNet. Depending on the computational environment (e.g., a server or an IoT device), the model size can be adjusted accordingly. Given that our model runs on an industrial PC with limited RAM and computational resources, we adopt the ViT-B/16 backbone, a variant of ViT with approximately 86M network parameters (330MB). Supervised fine-tuning (SFT) can be applied to adapt the general-purpose backbone to a specific task. Once fine-tuned, the backbone can remain fixed during the online phase to ensure stable and consistent feature extraction. In the authors' experience, SFT is essential when using a lightweight backbone, as it significantly enhances prediction accuracy. However, for traditional backbones pre-trained on extremely large datasets, SFT can often be omitted, as few-shot learners offer sufficient adaptability to tailor the backbone to specific tasks.

The few-shot learning module is crucial for enabling the model to adapt to new defect classes and maintain performance after deployment. It is common for new defect types to emerge as inspection continues, or new product variants are introduced. In this work, we implement Prototypical Networks (ProtoNet) to support few-shot learning.

The model calibration module serves as a post-processing component for the classification head. It adjusts the prediction probabilities to improve the reliability and interpretability of the model's outputs. In this study, the Bayesian Binning into Quantiles (BBQ) model calibration method is utilized. The approach involves dividing the predicted probability space into quantile-based bins and fitting a Bayesian model to estimate the true likelihood of correctness within each bin. By capturing uncertainty, BBQ produces well-calibrated probabilities even when data is sparse or imbalanced.

During the online phase, the trained model is deployed to identify part defects from a given image I. To evaluate model performance, we apply the BBQ calibration method, which adjusts predicted probabilities to better reflect true likelihoods under uncertainty. For confidence estimation, we use the distance between a sample's embedding and its nearest prototype in the feature space. Based on this confidence measure, a decision is made on whether the image requires human review. In this study, we set 90% confidence as a threshold for sample selection. If so, a human inspector will label the image and annotate the defective region. The labeled image is then added to the training set to support continual online learning and improve future model performance. Once enough labeled images are collected (e.g., 10 images), the data is used to update the few-shot learner. During the online phase, the fine-tuned backbone remains fixed to ensure consistent feature representation, while only the few-shot learner is updated to incorporate the new data and adapt to emerging defect classes.

3.2. Few-Shot Learning (FSL)

The FSL is the key to accommodating new defects and maintaining model prediction accuracy after the model is deployed. Given an input image $\boldsymbol{I}$, the fine-tuned backbone (or feature encoder) extracts a feature vector $\boldsymbol{V} \in \boldsymbol{R^D}$ from the input $\boldsymbol{I}$. The FSL in this study further maps the feature vector $\boldsymbol{V}$ to a task-specific feature vector $\boldsymbol{C} \in \boldsymbol{R^d}$. The FSL establish a mapping $\boldsymbol{f_\phi}: \boldsymbol{R^D} \rightarrow \boldsymbol{R^d}$ by using neural networks. In this study, the Prototypical Network (ProtoNet) (Snell et al., 2017) is utilized to build the few-shot learner.

Given a labelled dataset $S = \{(\mathbf{x}_i, y_i)\}_{i=1,\dots,N}$ where $\mathbf{x}_i \in R^D$ represents the feature vector given by the feature encoder (or fine-turned backbone) and $y \in \{1,2,\dots,K\}$ is the class labels. S_k denotes the subset of samples belonging to class k. The loss function for ProtoNet can be written as follows.

$$\mathbf{c}_k = \frac{1}{|S_k|} \sum_{(\mathbf{x}_i, y_i) \in S_k} f_\phi(\mathbf{x}_i) \tag{1}$$

$$L(\mathbf{x}) = \frac{\exp\left[-d\left(f_\phi(\mathbf{x}), \mathbf{c}_k\right)\right]}{\sum_{k'} \exp\left[-d\left(f_\phi(\mathbf{x}), \mathbf{c}_{k'}\right)\right]} \tag{2}$$

Where $\mathbf{c}_k$ in Eq. (1) is the centroid of class k. The distance function $d(\cdot,\cdot)$ measures the distance between the embedding of a sample and the class centroid. By minimizing the loss function, the neural network learns a new feature mapping in which examples are close to their class prototype but far from other classes.

In this study, FSL is leveraged for three main purposes: (1) tailoring the network to specific classification tasks during the offline phase; (2) maintaining model performance through continual learning from new samples; (3) accommodating new defect classes with minimal supervision. The training procedure of the ProtoNet is as follows:

Table 2. The ProtoNet training algorithm using 3-way 5-shot learning as an example.

ProtoNet Training Algorithm (3-Way 5-Shot):	
1	**Initialization**: Training set $S = \{(\mathbf{x}_i, y_i)\}_{i=1,\dots,N}$, where $y_i \in \{1,2,\dots,K\}$.
2	**Building Data Episode**: *2.1* Randomly select 3 subclasses (3-Way) from all K classes. *2.2* Randomly select 15 samples from the training set to build the ***Support Set***. The support set has 5 samples for each selected subclass (5-Shot). *2.3* Randomly select samples from the training set to build the ***Query Set***. The ***Query Set*** has no overlapping with the ***Support Set***.
3	Minimizing the loss function: *3.1 C*ompute the prototype from the ***Support Set*** based on Eq. (1). *3.2* Minimize the loss function Eq. (2) by using the ***Query Set***.
4	Repeat step 2 and 3 until the network is fully trained.

3.3. Model Calibration and Model Performance Metrics

Model calibration aims to adjust a model's predicted probabilities so that they better reflect the true likelihood of prediction outcomes. In quality inspection applications, understanding model calibration is as important as the prediction itself. Common calibration methods include Histogram Binning (or Hist**)**, which divides the predicted probabilities into fixed width bins and adjusts the prediction for each bin based on the observed frequency of correct predictions, providing a simple yet effective way to reduce miscalibration. Platt Scaling (or Platt), which fits a logistic regression model to the model's output probabilities; Isotonic Regression (IsoReg), a non-parametric method that fits a piecewise-constant function; and Bayesian Binning into Quantiles (BBQ), which combines quantile binning with Bayesian inference to improve robustness. Calibration is typically performed on a validation dataset after model training and is particularly useful for deep learning models, which are often overconfident in their predictions. Proper calibration enhances model interpretability, supports better decision-making and model reliability.

Model performance evaluation in classification tasks often involves multiple metrics to comprehensively assess both accuracy and reliability. Accuracy and Area Under the Curve (AUC) measure how well a model classifies data, with accuracy focusing on overall correctness and AUC assessing the model's ability to distinguish between classes. However, these metrics do not evaluate how reliable the predicted probabilities are. Expected Calibration Error (ECE) and Maximum Calibration Error (MCE) address this by evaluating the alignment between predicted probability and actual outcomes. While ECE captures the average calibration error across bins, MCE highlights the worst-case miscalibration.

Expected Calibration Error (ECE) quantifies the average difference between predicted probabilities and actual outcomes across all predictions, capturing how well the probability scores reflect true likelihoods. Maximum Calibration Error (MCE) complements this by reporting the worst-case deviation among all prediction bins, highlighting the most severe miscalibration. These metrics are written as follows:

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{N} |acc(B_m) - conf(B_m)| \quad (3)$$

$$MCE = \max_{m \in \{1,\dots,M\}} |acc(B_m) - conf(B_m)| \quad (4)$$

Where N is the total number of samples, M is the number of bins. B_m is the set of sampls falls in the bin m, $acc(B_m)$ shows the average prediction accuracy in sample bin B_m and $conf(B_m)$ is the average uncalibrated prediction probability given by the classification head of the deep neural network. Normally, uncalibrated probability is a real number $c \in [0,1]$. To evaluate calibration performance, the range of ccc is often discretized into $M = 10$ equal-width bins, which are then used to construct a reliability diagram that visually compares predicted probability with actual accuracy.

3.4. Online Learning

During the online phase, model performance is tracked regularly (e.g., hourly or daily) and the model parameters for the few-shot learner are updated when the prediction accuracy drops below 95% or when a new class is identified.

First, model performance tracking is conducted by randomly selecting a subset of AI-labeled images with prediction confidence ≤ 90% for human inspection during each monitoring period. These low-confidence images are reviewed and re-labeled by human experts. The validated images, along with their expert annotations, are then added to the training set to update the few-shot learner. The backbone feature encoder remains unchanged during this update.

Second, the few-shot learner is updated according to the training algorithm outlined in Table 2. The triggering condition for the model parameter update is prediction accuracy ≤ 95% or a new class is identified. In this study, a new defect class is added if more than 10 samples of that

class are observed. This augmented class will be utilized for few-shot training.

To emphasize the importance of newly added images during model updates, higher weights are assigned to these samples to increase their likelihood of being selected for the support set and query set in the training algorithm.

3.5. Hardware Setup and Data Description

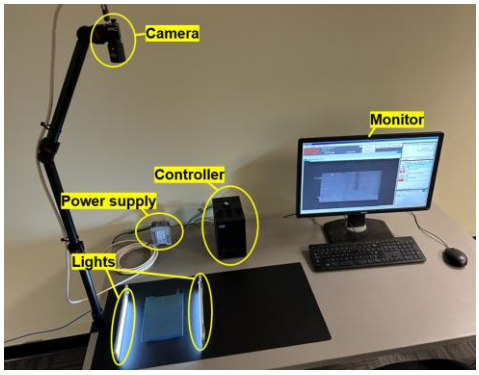


Figure 4. The hardware setup for the optical inspection.

Figure 4 illustrates the hardware setup for the inspection task. The vision system consists of a camera, optical lens, power supply, and a camera controller with built-in image processing capabilities. Due to hardware constraints on the camera controller, a moderately sized model is required.

Table 3. Hardware Specifications

Operating System	Windows 7 Embedded
CPU	Intel Celeron, dual core
Memory	8 GB RAM, 32 GB ROM

Table 3 summarizes the hardware platform used for optical-inspection data acquisition and on-device deployment. The selected industrial controller-class PC lacked a discrete GPU; consequently, all inference was executed on the CPU. Within these constraints, the Vision Transformer base model with 16-pixel patches (ViT-B/16) was adopted as the backbone because it offers a favorable balance between representational capacity and computational efficiency for resource-limited environments. ViT-B/16 comprises 12 transformer encoder layers with multi-head self-attention, yielding reliable accuracy without exceeding the compute and memory budgets of a CPU-only system. To further minimize overhead during online adaptation, the Prototypical Network (ProtoNet) adapter was implemented as a lightweight module containing a single transformer block, preserving real-time responsiveness while enabling class-prototype refinement.

The number of data samples used for model training and validation is summarized in Table 4. During the offline phase, the training data contains only three classes, with class 1 representing the healthy condition. In contrast, the online phase includes data from a total of five classes, two of which (Classes 4 and 5) are previously unseen by the trained neural network. To simulate a real-world operating environment, the online training samples are fed sequentially to the model as a data stream. The objective is to adapt the pre-trained network to these unseen classes using FSL.

Table 4. Flex Electronics Data Description

Class	Training		Validation	Total
	Offline	Online		
1(Healthy)	800	100	30	930
2	500	50	30	580
3	500	20	30	550
4	N/A	20	10	30
5	N/A	20	10	30
Total	1800	210	110	2120

4. Results and Discussions

4.1. Case Study 1: Flex Electronics

4.1.1. Offline Phase: Model Training

The ResNet50 (Koonce, 2021) and ViT-B/16 (Vaswani et al., 2017) that are pre-trained on ImageNet-1k are utilized as a backbone feature encoder. The supervised fine-tuning for the pretrained backbone is attempted. The detailed model prediction performance for the ViT-B/16 after SFT (or ViT^{+}) are provided in Table **6** and the benchmarking between the two models can be found in Table 7. All the results in this subsection are generated by using the training (offline) and validation set in Table 4**Error! Reference source not found.**. Table 5 shows the SFT training parameters for ViT^{+}.

Table 5. SFT training parameters

Learning Rate	1e-4
Optimizer	AdamW
Number of Epochs	10
Image Size	224x224
Batch Size	64
Patch Size	16
Trainable ViT Layers	7

Table 6 compares four model calibration methods: Histogram Binning (Hist), Platt Scaling (Platt) (Böken, 2021), Isotonic Regression (IsoReg) (Zadrozny & Elkan, 2002), and Bayesian Binning into Quantiles (BBQ) (Naeini et al., 2015). The evaluation is based on four performance metrics: class prediction accuracy (Acc), Area Under the Curve (AUC), Expected Calibration Error (ECE), and Maximum Calibration Error (MCE). The backbone network used in this study is ViT-B/16, a variant of the Vision

Transformer (ViT) architecture introduced by Meta AI for image classification tasks. It consists of 12 Transformer encoder blocks, among which 7 blocks are made tunable for supervised fine-tuning (SFT).

Table 6. The model performance for ViT†

Class 1				
	Hist	**Platt**	**IsoReg**	**BBQ**
Acc	.900	**.930**	.870	**.930**
AUC	.890	.890	**.910**	.860
ECE	**.056**	.074	.071	.064
MCE	.618	.189	.841	**.136**
Class 2				
	Hist	**Platt**	**IsoReg**	**BBQ**
Acc	**.970**	.900	.900	.900
AUC	.970	.970	**.980**	.970
ECE	.025	.108	.034	**.021**
MCE	.171	.656	.517	**.065**
Class 3				
	Hist	**Platt**	**IsoReg**	**BBQ**
Acc	**.930**	**.930**	**.930**	**.930**
AUC	.950	.950	**.970**	.950
ECE	.046	.079	**.041**	.054
MCE	.268	.290	.478	**.202**

The results in Table 6 demonstrate promising performance from ViT†, with all three classes achieving satisfactory prediction accuracy. Among the four calibration methods, all yielded comparable results. For subsequent analysis, we adopt BBQ as the post-processing method, given its strong balance between robustness and predictive performance.

Table 7 shows more comprehensive benchmarking among different model architectures. Comparison between the untuned backbone feature encoder (i.e., ResNet+BBQ and ViT+BBQ) with the fine-tuned backbone (ResNet†+BBQ and ViT†+BBQ) clearly indicates the SFT can significantly improve the model performance. This is because the backbone feature encoder is fairly lightweight and the training data in this inspection is significantly different with ImageNet. Therefore, the SFT is a necessary step in this analysis. Further comparison of the untuned backbone and fine-tuned backbone with ProtoNet indicate that the SFT is required to improve the prediction accuracy in this task even with the ProtoNet concatenated to the backbone.

Comparison between the ViT†+BBQ and the ViT†+ProtoNet+BBQ shows quite similar model prediction performance. This is because a large number of training samples for class 1,2, and 3 are available and the merits of ProtoNet are not fully demonstrated. Moreover, the separability of these classes is not so difficult. This is supported by the feature embedding distribution in Figure 5. The middle figure already shows great separability among the class 1-3 after backbone SFT.

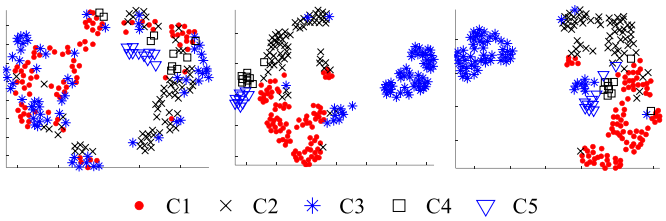


Figure 5. A visualization of scatter plots for the feature

Table 7. The benchmarking of different network design for the offline AI model training

Network	Class 1			Class 2			Class 3		
	Acc	AUC	ECE	Acc	AUC	ECE	Acc	AUC	ECE
ResNet + BBQ	.800	.690	.090	.370	.650	.075	.133	.400	.102
ViT + BBQ	.230	.420	**.045**	.000	.640	.035	.833	.730	.104
ResNet† + BBQ	.867	**.950**	.081	.870	.900	.155	.867	.940	.048
ViT† + BBQ	.933	.890	.074	.900	**.970**	.108	.933	.950	.079
ResNet + ProtoNet + BBQ	.033	.672	.067	.233	.551	**.005**	.900	.537	**.020**
ViT + ProtoNet + BBQ	.000	.516	.056	.433	.717	.042	.900	.611	.030
ResNet† + ProtoNet + BBQ (Proposed)	.900	.940	.108	.900	.918	.187	.867	**.954**	.037
ViT† + ProtoNet + BBQ (Proposed)	**.933**	.847	.065	**.933**	.947	.022	**.933**	.946	.051
	Class 4			Class 5			Overall		
	Acc	AUC	ECE	Acc	AUC	ECE	Acc	AUC	ECE
ResNet + BBQ	.000	.300	.094	.000	.680	.154	.355	.542	.103
ViT + BBQ	.000	.820	.000	.000	.880	.000	.291	.699	.037
ResNet† + BBQ	.000	.510	.091	.000	.560	.091	.709	.774	.093
ViT† + BBQ	.000	.840	.000	.000	.400	.000	.755	.811	.052
ResNet + ProtoNet + BBQ	.000	.646	.000	.000	.510	.000	.318	.583	**.018**
ViT + ProtoNet + BBQ	.000	.371	.000	.000	.520	.000	.364	.547	.027
ResNet† + ProtoNet + BBQ (Proposed)	.000	.500	.091	.000	.500	.091	.727	.762	.103
ViT† + ProtoNet + BBQ (Proposed)	.000	.764	.003	.000	.805	.003	**.764**	**.862**	.029

* ViT† and ResNet† denote the backbone feature extractor after supervised fine-tuning

* The prediction performance for classes 4 and 5 is bad as these defect types were unseen by the trained model, see Table 4

vectors given by (*left*) original ViT backbone; (*middle*) ViT^{+} backbone after supervised fine-tuning. (*right*) ViT + ProtoNet. The scatter plot is generated by the dimension reduction algorithm UMAP.

The prediction performance for classes 4 and 5 in Table 7 is not satisfactory so far, as these defect types were not included in the offline training. In the online learning phase, we will demonstrate improved performance in these two classes by fine-tuning the ProtoNet using online training samples. In the online phase, we assume that data is fed sequentially to the deployed model. Samples with a prediction confidence of ≤ 90% will be flagged for human labeling to support continual learning.

4.1.2. Online Phase: Continual Learning

After the offline AI model training, the online training set (Table **4**) is fed into the trained AI model for sample selection. The prediction confidence scores, generated by the ViT^{+}+ProtoNet+BBQ model, are shown in Figure 6. Samples with prediction confidence below 90% are selected for human labeling, while those with confidence scores equal to or above 90% are automatically labeled by the AI model. Table **9** summarizes the prediction accuracy for both groups. For the unselected samples (confidence ≥ 90%), the overall prediction accuracy reaches 98.35%, consistent with the expectation that high-confidence predictions are reliable. In contrast, the accuracy for the selected samples (confidence < 90%) is 48.31%, providing clear evidence that the model's confidence score is a trustworthy indicator of prediction reliability.

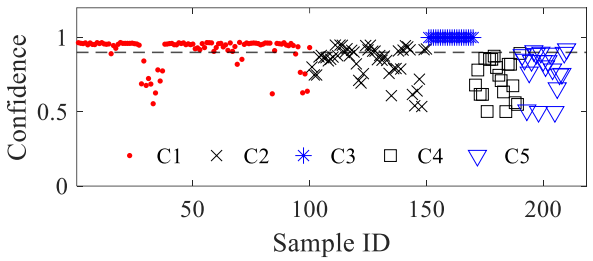
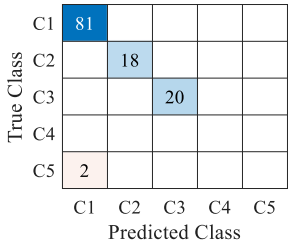
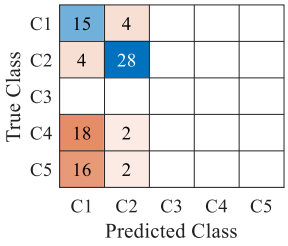


Figure 6. Prediction confidence plot for the online training set. The samples with confidence score <90% are selected for human labeling.

Table 9. The prediction outcome on the online training set

Unselected Samples (Confidence>=90%)	**Selected Samples (Confidence<90%)**
121 out of 210 AI generated labels are accepted	89 out of 210 are selected for human labelling
	
Overall Accuracy 98.35%	Overall Accuracy 48.31%

The selected samples and expert-provided labels in Table 9 are used to update the ProtoNet by introducing two new classes, C4 and C5, into the label set. In the continual learning setting, the feature encoder (ViT^{+} or ResNet^{+}) remains fixed, while only the ProtoNet is updated using the training algorithm described in Table 2. Figure **7** illustrates the improvement in prediction confidence before and after continual learning. Since C4 and C5 were previously unseen by the trained model, the initial confidence for these classes was low. However, after continual learning, the prediction confidence for C4 and C5 increases significantly, demonstrating the effectiveness of the continual learning.

Table 8 highlights the performance improvements achieved through continual learning. Compared to the best offline model, ViT^{+}+ProtoNet+BBQ, the prediction accuracy for the newly introduced classes C4 and C5 has significantly increased. As a result, the overall prediction accuracy improved from 76.4% to 92.7%.

Table 8. The prediction accuracy on the validation set after online continual learning

Network	Class 1			Class 2			Class 3		
	Acc	AUC	ECE	Acc	AUC	ECE	Acc	AUC	ECE
ResNet^{+}+ProtoNet+BBQ+CL (Proposed)	.933	.985	.039	.900	.957	.090	.933	.983	.038
ViT^{+}+ProtoNet+BBQ+CL (Proposed)	.933	.960	.016	.967	.968	.011	.933	.960	.017
ViT^{+}+RealtionNet+BBQ+CL	.933	.919	.140	.933	.912	.103	.967	.959	.074
ViT^{+}+BBQ (Baseline)	.933	.890	.074	.900	.970	.108	.933	.950	.079
	Class 4			Class 5			Overall		
	Acc	AUC	ECE	Acc	AUC	ECE	Acc	AUC	ECE
ResNet^{+}+ProtoNet+BBQ+CL (Proposed)	.700	.993	.034	.900	.998	.009	.900	**.983**	.038
ViT^{+}+ProtoNet+BBQ+CL (Proposed)	.900	.985	.006	.800	.970	.005	**.927**	.969	**.011**
ViT^{+}+RealtionNet+BBQ+CL	.800	.914	.096	.667	.911	.079	.860	.923	.050
ViT^{+}+BBQ (Baseline)	.000	.840	.000	.000	.400	.000	.755	.811	.052

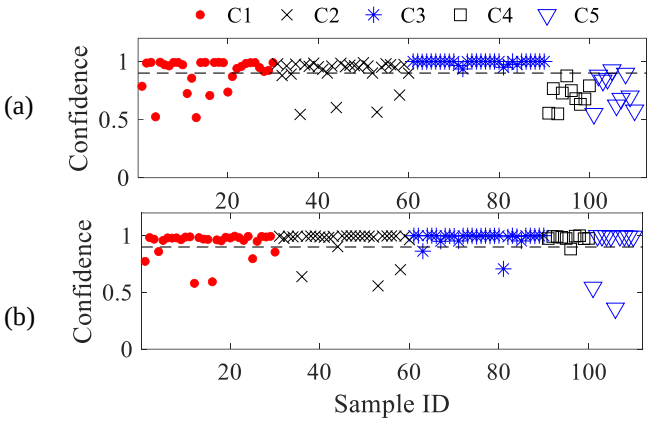


Figure 7. (a) The model prediction confidence on the validation dataset (see Table 4) before continual learning. Prediction confidence on C4 and C5 is low. (b) The model prediction confidence after continual learning. The prediction confidence in all classes is improved.

4.2. Case Study 2: Semi-Conductor Wafer Map

To validate repeatability and domain transfer displayed in case study 1, the proposed framework is evaluated on the WM811-k wafer-map dataset using the same five-label data structure: Healthy, Edge Ring, Center, Scratch, and Random. To mirror similar factory conditions to the previous case study, the dataset is intentionally imbalanced: common defects are abundant in the offline training split, whereas rare defects are absent offline and appear only in the online pool, simulating "unseen-at-deployment" classes.

It was established in the previous case study that fine-tuned ViT backbone outperforms its pretrained variant and ResNet baselines. Therefore, in case study 2, we focus on three configurations that directly test the claims of our framework:

1. ViT┼ + BBQ (baseline): strong offline backbone;
2. ViT┼ + ProtoNet + BBQ: adds a few-shot learner without rare-class support;
3. ViT┼ + ProtoNet + BBQ + CL: the online continual-learning setting, where low-confidence/novel samples are labeled and used in 3-way, 5-shot episodes to adapt the few-shot learner while keeping the backbone fixed.

Table 10. WM811-k Data Description

Class	Training		Validation	Total
	Offline	Online		
None	2000	1000	550	3550
Edge Ring	2000	1000	550	3550
Center	2000	1000	550	3550
Scratch	0	500	225	725
Random	0	500	225	725
Total	6000	4000	2100	12100

Table 10 summarizes the WM811-k data structure. The offline split contains only the known classes (None, Edge Ring, Center). The online split supplies the few-shot supports for rare classes (Scratch, Random) during CL episodes.

Reported in Table 11 are per-class and overall accuracy on the validation data. As expected, ViT┼ + BBQ performs strongly on the common classes but fails to correctly classify the defective classes that were unseen offline. After attaching the ProtoNet few-shot learner (ViT┼ + ProtoNet + BBQ), performance on the common classes is maintained, however, adaptation to the rare defective remains poor. Once meta-learning is introduced (ViT ┼ +ProtoNet +BBQ+CL) and the few-shot learner is updated with low confidence samples from the online training episode, rare-class accuracy improves substantially while preserving performance on common classes.

5. CONCLUSIONS

This paper presents a novel methodology for vision backbone fine-tuning and continual learning in optical inspection tasks. The approach is demonstrated through a applications in liquid crystal film and semi-conductor wafer map defect detection. A vision inspection testbed—comprising a camera system, computational hardware, and a human–machine interface—was developed to support the inspection process. The proposed AI algorithms are deployed to an industrial PC to enable automated data processing. Through benchmarking various network architectures, the study arrives at the following key findings:

1) Vision backbones can achieve high prediction accuracy when fine-tuned with supervised learning. These models are suitable for deployment on resource-constrained hardware.

2) Continual learning using few-shot learners (e.g., ProtoNet) is essential for adapting to unseen defect classes

Table 11. The prediction accuracy on the validation set for the WM-811K dataset

Network	Accuracy					
	None	Edge Ring	Center	Scratch	Random	Overall
ViT┼+BBQ (baseline)	.980	.996	.969	.080	.000	.780
ViT┼+ProtoNet +BBQ	.989	.995	.947	.000	.000	.767
ViT┼+ProtoNet +BBQ+CL (3-way 5-shot FSL)	.933	.985	.936	.733	.920	**.925**

and mitigating performance degradation over time.

3) Calibration techniques are important for evaluating the model reliability and prediction confidence. Human review and annotation of low-confidence samples is crucial for online model monitoring and performance maintenance.

ACKNOWLEDGEMENT

This material is based on research sponsored by Office of the Under Secretary of Defense for Research and Engineering, Strategic Technology Protection and Exploitation, and Defense Manufacturing Science and Technology Program under agreement number W15QKN-19-3-0003. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government.

REFERENCES

Böken, B. (2021). On the appropriateness of Platt scaling in classifier calibration. *Information Systems*, *95*, 101641.

Chen, J., He, J., Chen, F., Lv, Z., & Tang, J. (2025). Forward-Only Continual Learning. *arXiv preprint arXiv:2509.01533*.

Chen, Y., Wang, X., Liu, Z., Xu, H., & Darrell, T. (2020). A new meta-baseline for few-shot learning. *arXiv preprint arXiv:2003.04390*, *2*(3), 5.

Clark, K., Luong, M.-T., Le, Q. V., & Manning, C. D. (2020). Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Cui, P., & Wang, J. (2022). Out-of-distribution (OOD) detection based on deep learning: A review. *Electronics*, *11*(21), 3500.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., & Gelly, S. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Elharrouss, O., Akbari, Y., Almadeed, N., & Al-Maadeed, S. (2024). Backbones-review: Feature extractor networks for deep learning and deep reinforcement learning approaches in computer vision. *Computer Science Review*, *53*, 100645.

Finn, C., Abbeel, P., & Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. International conference on machine learning,

Gonzales, R. C., & Woods, R. E. (2002). Digital Image Processing, 2-nd Edition. In: Prentice Hall.

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. International conference on machine learning,

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., & Vasudevan, V. (2019). Searching for mobilenetv3. Proceedings of the IEEE/CVF international conference on computer vision,

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2022). Lora: Low-rank adaptation of large language models. *ICLR*, *1*(2), 3.

Koonce, B. (2021). ResNet 50. In *Convolutional neural networks with swift for tensorflow: image recognition and dataset categorization* (pp. 63-72). Springer.

Kornblith, S., Shlens, J., & Le, Q. V. (2019). Do better imagenet models transfer better? Proceedings of the IEEE/CVF conference on computer vision and pattern recognition,

Li, Z., Zhou, F., Chen, F., & Li, H. (2017). Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:1707.09835*.

Lo, C.-M., & Lin, T.-Y. (2024). Automated optical inspection based on synthetic mechanisms combining deep learning and machine learning. *Journal of Intelligent Manufacturing*, 1-15.

Naeini, M. P., Cooper, G., & Hauskrecht, M. (2015). Obtaining well calibrated probabilities using bayesian binning. Proceedings of the AAAI conference on artificial intelligence,

Raghu, A., Raghu, M., Bengio, S., & Vinyals, O. (2019). Rapid learning or feature reuse? towards understanding the effectiveness of maml. *arXiv preprint arXiv:1909.09157*.

Shih, Y., Kuo, C.-C., & Lee, C.-H. (2023). Low-Cost Real-Time Automated Optical Inspection Using Deep Learning and Attention Map. *Intelligent Automation & Soft Computing*, *35*(2).

Snell, J., Swersky, K., & Zemel, R. (2017). Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems*, *30*.

Stein, K., Mahyari, A. A., Francia III, G., & El-Sheikh, E. (2025). Adaptive Additive Parameter Updates of Vision Transformers for Few-Shot Continual Learning. *arXiv preprint arXiv:2504.08982*.

Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H., & Hospedales, T. M. (2018). Learning to compare: Relation network for few-shot learning. Proceedings of the IEEE conference on computer vision and pattern recognition,

Tan, M., & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. International conference on machine learning,

Triantafillou, E., Zhu, T., Dumoulin, V., Lamblin, P., Evci, U., Xu, K., Goroshin, R., Gelada, C., Swersky, K., & Manzagol, P.-A. (2019). Meta-dataset: A dataset of datasets for learning to learn from few examples. *arXiv preprint arXiv:1903.03096*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*.

Vinyals, O., Blundell, C., Lillicrap, T., & Wierstra, D. (2016). Matching networks for one shot learning. *Advances in Neural Information Processing Systems*, *29*.

Wenger, J., Kjellström, H., & Triebel, R. (2020). Non-parametric calibration for classification. International Conference on Artificial Intelligence and Statistics,

Ye, H.-J., Hu, H., Zhan, D.-C., & Sha, F. (2020). Few-shot learning via embedding adaptation with set-to-set functions. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition,

Zadrozny, B., & Elkan, C. (2002). Transforming classifier scores into accurate multiclass probability estimates. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining,

Zaken, E. B., Ravfogel, S., & Goldberg, Y. (2021). Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*.

Zhu, H., Huang, J., Liu, H., Zhou, Q., Zhu, J., & Li, B. (2021). Deep-learning-enabled automatic optical inspection for module-level defects in LCD. *IEEE Internet of Things Journal*, *9*(2), 1122-1135.

BIOGRAPHIES

Kody Haubeil received his B.S. degree in mechanical engineering from Otterbein University, Westerville OH, in 2024. He is currently pursuing his M.S. and Ph.D. degrees in mechanical engineering at the University of Cincinnati. His research interests include computer vision, machine learning, industrial A.I., and human-robot collaboration.

Tarek Yahia is a Ph.D. Mechanical Engineering student at University of Cincinnati, Cincinnati, OH. He received his B.S. degree in Industrial Engineering from University of South Florida, Tampa, FL in 2024. His research interests include computer vision, image & signal processing, machine learning, and human-robot collaboration.

Alexander Suer received his B.S. and M.S. in mechanical engineering from the University of Cincinnati, Cincinnati, OH, USA. He is pursuing his Ph.D. degree in mechanical engineering with the University of Cincinnati. His research includes computer vision, language processing, physics informed neural networks, semiconductor PHM, and signal processing.

David Siegel is the Chief Technology Officer at Predictronics Corporation. He received his B.S. (2007), M.S. (2009), and Ph.D. (2013) degrees in Mechanical Engineering from the University of Cincinnati, Cincinnati, OH. His research efforts include advanced diagnostic methods for industrial robots, health monitoring systems for railway applications, failure prediction tools for machine tool bearings, and intelligent maintenance systems for military ground vehicles.

Donald Davis received the B.A. degree in Physics from Thiel College in Greenville, PA and his M.S. in Physics at Oklahoma State University, Stillwater, OK. Mr. Davis has 35 years' experience in advanced manufacturing including military laser systems, liquid crystal displays, MEMS and nano-embossed plastic films and is currently the Director of Process Engineering at Kent Displays, Inc. in Kent, OH.

Xiaodong Jia received the B.S. degree in engineering thermo-dynamics from Central South University, Changsha, China, in 2008, the M.S. degree in mechanical engineering from Shanghai Jiao Tong University, Shanghai, China, in 2014, and the Ph.D. degree in mechanical engineering from the University of Cincinnati, Cincinnati, OH, USA, in 2018. He is currently an Assistant Professor with the Department of Mechanical and Materials Engineering, University of Cincinnati. His research interests include prognostics and health management, data mining, and ML.