Guided Gaussian Process-Based Anomaly Detectors for Uncertainty-Aware Defect Detection in New Objects

Abstract

In computer vision, detecting defects in new objects is crucial. Conventional anomaly detection methods are often unsuitable for this task. However, we can guide an anomaly detector to transform into a specialized defect detector. We present a guided Gaussian process-based anomaly detector with a strong focus on "uncertainty awareness," enabling the system to express "I don't know" when predictions are uncertain. Building on prior work by Ching-Wen Ma and Yan-Wei Lui, we replace the Gaussian mixture model with a Gaussian process classifier. Our enhancements encompass Gaussian process-based deterministic uncertainty estimation, spectral normalization, and increased induced points. Experimental validation on an open-source fruit dataset demonstrates our approach's effectiveness. We achieve a 95.2% prediction accuracy, an improvement over the previous 93.4%. Furthermore, our methodology accelerates inference speed by 4.2 times compared to the previous algorithm. Qualitative t-SNE analysis further provides additional support for our findings.

1 INTRODUCTION

Deep learning-enhanced computer vision technologies have achieved remarkable success in various recognition tasks, paving the way for applications in autonomous vehicles, smart manufacturing, and precision agriculture, among others Krizhevsky et al. [2012], Sun et al. [2020], Rawat and Wang [2017]. However, the black-box nature of AI models raises concerns about their generalization capabilities and safety Papyan et al. [2020], Hui et al. [2022], Kothapalli et al. [2022]. A significant challenge arises when test samples diverge from training samples, leading to potential overconfidence in predictions Hein et al. [2019], Guo et al. [2017]. Addressing this, various strategies have been proposed to detect out-of-distribution samples Hendrycks and Gimpel [2016], Fang et al. [2022], Yang et al. [2021], Ren et al. [2019], Wu et al. [2022], enabling systems to recognize their predictive boundaries Meinke and Hein [2019] or anomalies Pang et al. [2021], Chalapathy and Chawla [2019], and thereby refrain from potentially detrimental decisions.

For instance, in the realm of autonomous vehicles, it's imperative that the system recognizes road conditions beyond its training data and adopts a cautious approach Mohseni et al. [2019, 2021, 2022], Hendrycks et al. [2021], Hendrycks and Mazeika [2022]. Similarly, in electronic manufacturing's Automatic Optical Inspection (AOI), undetected defects due to unseen conditions can lead to subpar product deliveries Ma and Lui [2023], Dai et al. [2020], Abd Al Rahman and Mousavi [2020], Huang et al. [2020], Ling and Isa [2023]. A truly safe system should be aware of its limitations, signaling unpredictability based on its self-awareness, and paving the way for collaboration with external tools or human intervention.

Building on the work of Ma et al. Ma and Lui [2023], which discussed multi-object defect detection scenarios, we introduce the concept of "Guided Anomaly Detection". Our method enhances the original approach by substituting the Gaussian mixture model with an advanced Gaussian process classifier, as depicted in Figure 1. We delve into the nuanced distinction between defects and anomalies, highlighting that while all defects can be considered anomalies, the converse isn't true. A prime example is fruit inspection: a square-shaped apple is anomalous but not defective. Intriguingly, deep learning feature extractors can be guided to transition from anomaly to defect detection.

While the concept of "guided anomaly detection" might echo the sentiments of Ma et al. Ma and Lui [2023], our paper distinguishes itself by enhancing performance and speed. We optimize the method from van Amersfoort et al. [2021] using a two-stage optimization procedure, employ-

Submitted to the 40th Conference on Uncertainty in Artificial Intelligence (UAI 2024). **To be used for reviewing only**.

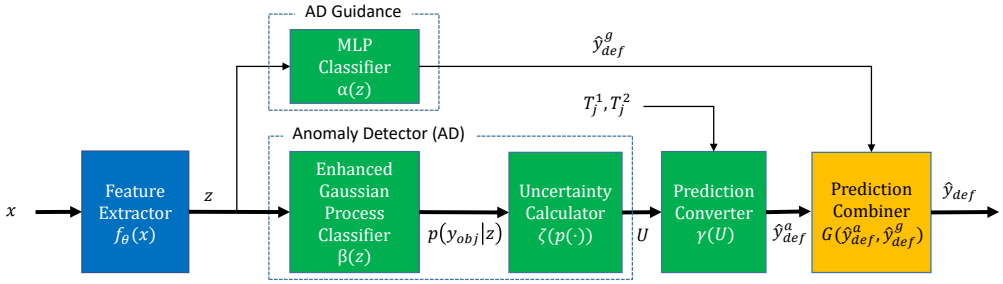

Figure 1: The inference block diagram of the proposed method. The prediction converter $\gamma(\cdot)$, using optimized thresholds T_j^1 and T_j^2, translates this uncertainty U into the secondary defect prediction $\hat{y}^g_{def}$. Consequently, two heterogeneous predictions, $\hat{y}^a_{def}$ and $p(y_{obj}|z)$, are harmonized into two homogeneous predictions, $\hat{y}^a_{def}$ and $\hat{y}^g_{def}$, facilitating their amalgamation to yield the final prediction $\hat{y}_{def}$.

ing a larger set of inducing points. Additionally, we empirically underscore the importance of spectral normalization in the feature extractor, which prevents "feature collapse"van Amersfoort et al. [2021], a phenomenon detrimental to guidance, especially when generalizing to new objects.

Our contributions includes 1.) Introduction of the "guided anomaly detection" concept, enabling defect detection tailored for new objects with meta-cognitive capabilities. 2.) Performance and speed enhancements by transitioning from a Gaussian mixture model-based detector to a Gaussian process-based one. 3.) Enhancement of the uncertainty estimation method proposed in van Amersfoort et al. [2021] through a two-stage procedure and an increased number of inducing points. 4.) Empirical exploration of the impact of spectral normalization on the feature extractor, with both qualitative and quantitative analyses underscoring its significance.

2 RELATED WORK

Deep Learning in Vision-Based Defect Detection: Vision-based defect detection primarily aims to identify defects within images, which could represent objects or object surfaces Ren et al. [2022], Czimmermann et al. [2020]. The advent of deep learning has revolutionized this domain, with neural networks now being widely used for feature extraction and predictions Tulbure et al. [2022], Bhatt et al. [2021].

A phenomenon termed 'neural collapse' has been discussed in the literature, suggesting that deep neural networks tend to reduce input images to specific, output-related features Papyan et al. [2020], Hui et al. [2022], Kothapalli et al. [2022]. However, excessive feature collapse can be detrimental, and spectral normalization has been proposed as a remedy van Amersfoort et al. [2021]. In our work, we focus on images representing objects, such as fruits or electronic devices, and employ spectral normalization to enhance generalization to new objects.

Anomaly and Out-of-Distribution Detection in Vision Anomaly detection and out-of-distribution (OOD) detection share similarities, with both aiming to identify data instances that deviate significantly from typical instances Pang et al. [2021], Chalapathy and Chawla [2019]. Ensuring the safety of intelligent systems is paramount, and anomaly detection plays a crucial role in this aspect Mohseni et al. [2019, 2021]. Systems equipped with the capability to detect anomalies can make informed decisions, enhancing overall safety. In our research, we leverage the inherent characteristics of anomalies to imbue our system with uncertainty awareness.

Gaussian Process Classification and Reliable Uncertainty Estimation Many deep neural network-based image classification methods exhibit overconfidence when faced with OOD samples Hein et al. [2019]. Calibration techniques are essential to adjust the confidence levels of these networks Guo et al. [2017]. Gaussian process classification (GPC) emerges as a promising approach, offering reliable prediction confidence Gibbs [1998], Hensman et al. [2015].

Efficient Gaussian Process Classification for Large Datasets Deep kernel learning combines the strengths of deep neural networks and GPC, facilitating the processing of large datasets Bradshaw et al. [2017], Wilson et al. [2016]. To enhance computational efficiency, one-pass uncertainty estimation algorithms have been developed Liu et al. [2020], Ulmer [2021]. Notably, GPC methods consistently yield high uncertainty estimates for OOD data, as demonstrated in Liu et al. [2021]. Inspired by the work in van Amersfoort et al. [2021], which employed GPC and spectral normalization, we aim to enhance the GMM-based guided anomaly detection approach with GPC.

3 METHODOLOGY

3.1 TASK DESCRIPTION

We are given a dataset of N samples, each associated with one of K unique objects, denoted as $\{x_n, n = 1, 2, ..., N\}$.

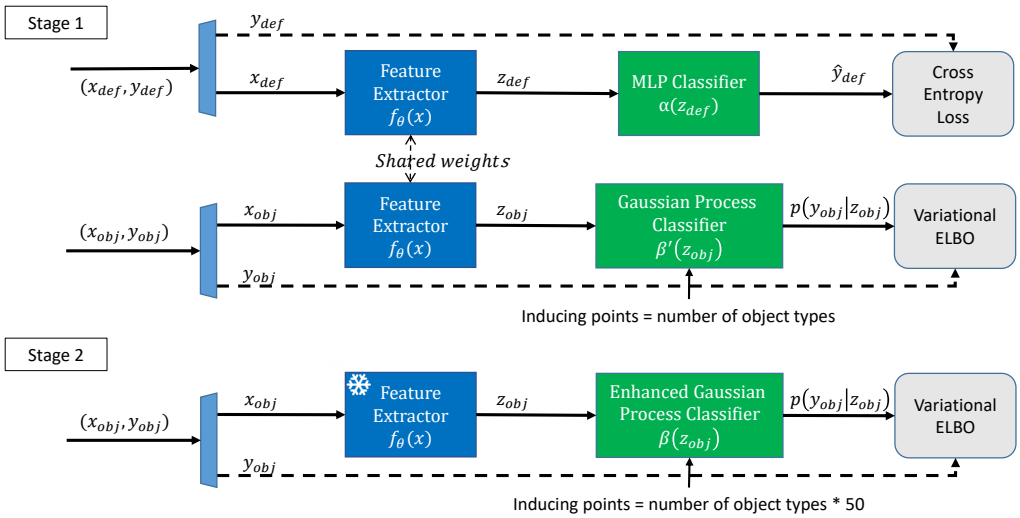

Figure 2: Illustration of the two-stage training procedure. In stage 1, the feature extractor $f_\theta(\cdot)$ is trained concurrently with the defect classifier $\alpha(\cdot)$ using $(x_{\text{def}}, y_{\text{def}})$ and with the preliminary object classifier $\beta'(\cdot)$ using $(x_{\text{obj}}, y_{\text{obj}})$. In stage 2, $\beta'(\cdot)$ is discarded, and a new classifier $\beta(\cdot)$ is trained with an expanded set of induced points for enhanced performance.

Among these objects, a subset is classified as old, while the remainder are new, resulting in the equation $K = K_{\text{old}} + K_{\text{new}}$. The K_{old} old objects consist of both normal and defective samples, labeled as $x_{\text{old,normal}}$ and $x_{\text{old,defective}}$, respectively. On the other hand, the K_{new} new objects exclusively contain normal samples, denoted as $x_{\text{new,normal}}$. Our primary goal is to detect defects across a diverse range of test samples, which encompass $x_{\text{old,normal}}$, $x_{\text{old,defective}}$, $x_{\text{new,normal}}$, and any potential $x_{\text{new,defective}}$ samples.

Importantly, while $x_{\text{new,defective}}$ samples are not present during training, they might emerge during testing. If a test sample does not correspond to any of the K objects, or if the system's predictive accuracy is unsatisfactory, it should demonstrate uncertainty-awareness by outputting an 'Unknown' label, indicating that the test sample extends beyond the system's trained knowledge base."

3.2 MODEL ARCHITECTURE

Refer to Figure 1 for the inference block diagram and Figure 2 for the training procedure. The following items elucidate these blocks and their respective functionalities:

1. **Feature Extractor** $f_\theta(\cdot)$

   The feature extractor is built upon MobileNetV3 Large. Given a sample x, the average pooling layer of the MobileNetV3 Large produces an output feature vector z. The feature extractor $f_\theta(\cdot)$ is regularized by spectral normalization to enhance its generalization to new object samples, x_{new}.

2. **MLP Classifier** $\alpha(\cdot)$

   The multi-layer perceptron (MLP) classifier produces the output denoted as $\hat{y}^g_{def}$, representing an auxiliary prediction of the defectiveness of the input sample x. During training, $\alpha(\cdot)$ acts as a guide, directing the anomaly detector towards defect detection. This is achieved by backpropagating the classification loss to $f_\theta(\cdot)$. With $\alpha(\cdot)$'s guidance, $f_\theta(\cdot)$ ensures defective samples are distinctly separated from normal ones in the space of z.

3. **Enhanced Gaussian Process Classifier** $\beta(\cdot)$

   The anomaly detector is an object type classifier. It is inspired by the Deterministic Uncertainty Estimation (DUE) proposed in van Amersfoort et al. [2021]. The enhanced Gaussian Process classifier (GPC) $\beta(\cdot)$ produces the output $p(y_{obj}|z)$, which is obtained by applying the softmax likelihood to a multivariate normal distribution generated by a Gaussian process regression algorithm. During training (Figure 2), the Evidence Lower Bound (ELBO) is utilized to update $f_\theta(\cdot)$ and $\beta'(\cdot)$ through variational optimization.

   In the first training stage, $\alpha(\cdot)$, $\beta'(\cdot)$, and $f_\theta(\cdot)$ are alternately updated, using knowledge from $x_{old,normal}$, $x_{old,defective}$, and $x_{new,normal}$ to guide the anomaly detector. The second training stage enhances DUE by freezing $f_\theta(\cdot)$ and increasing the number of inducing points by 50x to further boost performance.

4. **Uncertainty Calculator** $\zeta(\cdot)$

   The output of $\beta(\cdot)$, $p(y_{obj}|z)$, represents the probability of the input sample x being associated with each object category. This probability is then used to compute the uncertainty measure using the following entropy

equation:

$$U = \frac{1}{K} \sum_{i=1}^{K} -p(y_{obj=i}|z) \cdot \log(p(y_{obj=i}|z)). \quad (1)$$

5. **Prediction Converter** $\gamma(\cdot)$
   Bayesian optimization is employed to determine thresholds, T_j^1 and T_j^2, for $j = 1, ...K$. These thresholds transform the anomaly detector output U into the primary defectiveness prediction $\hat{y}_{def}^a$. The prediction follows the rule:

$$\hat{y}_{def}^a := \begin{cases} \text{unknown}, & U > T_j^2 \\ \text{defective}, & T_j^1 < U \leq T_j^2 \\ \text{normal}, & U \leq T_j^1 \end{cases} \quad (2)$$

   A more detailed explanation of the Bayesian optimization procedure can be found in the supplementary material.

6. **Prediction Combiner** $G(\cdot, \cdot)$
   After obtaining the primary defectiveness prediction $\hat{y}^a def$ and the auxiliary prediction $\hat{y}^g def$, we produce the final prediction. If $\hat{y}^a def$ and $\hat{y}^g def$ are inconsistent or if $\hat{y}^a def$ is unknown, the sample is labeled as 'unknown.' Otherwise, the final prediction $\hat{y} def$ is set equal to $\hat{y}^a def$, matching $\hat{y}^g def$.

3.3 TRAINING PROCEDURE

Stage 1, Initial Training: In the initial stage of training, we concurrently optimize two models:

1. The feature extractor, denoted as $f_\theta(\cdot)$, alongside the defect classifier, $\alpha(\cdot)$, using the training sample pairs $(x_{\text{def}}, y_{\text{def}})$.
2. The same feature extractor, $f_\theta(\cdot)$, alongside a Gaussian process classifier, $\beta'(\cdot)$, using the sample pairs $(x_{\text{obj}}, y_{\text{obj}})$.

For both models, the samples x_{def} and x_{obj} are drawn randomly from the dataset $\{x_n, n = 1, 2, ..., N\}$. The labels y_{def} and y_{obj} represent the defectiveness and object type of the samples, respectively.

Stage 2, Enhanced Training: In the subsequent stage, we fix the feature extractor $f_\theta(\cdot)$ and replace the Gaussian process classifier $\beta'(\cdot)$ with enhanced Gaussian process classifier $\beta(\cdot)$. The enhanced classifier is introduced and trained with an expanded set of induced points (50 times the original) to further enhance its performance.

3.4 BENEFITS OF REPLACING GMM WITH GPC IN ANOMALY DETECTION

In Ma and Lui [2023], an anomaly detector based on the Gaussian Mixture Model (GMM) is employed. However, as highlighted in Liu et al. [2021], the Gaussian process model emerges as the superior choice for uncertainty estimation. We observed several advantages in replacing GMM with GPC: 1) enhanced computational efficiency, 2) performance gains, and 3) simplified optimization of the thresholds for the prediction converter. A quantitative analysis, backed by experimental results, further elucidates the extent of these improvements.

3.5 PERFORMANCE IMPLICATIONS OF MODIFYING INDUCING POINTS IN DUE

The original DUE van Amersfoort et al. [2021] employs a number of inducing points equivalent to the number of classes. In the context of our anomaly detector, this corresponds to the number of object types. Intuitively, one might assume that increasing the number of inducing points would bolster performance. However, this isn't necessarily the case. A significant contribution of our work lies in devising a procedure that enhances the DUE by augmenting the number of inducing points.

Our approach bifurcates the training procedure into two stages. In the first stage, the number of inducing points is set equal to the number of object types (classes). Upon convergence, the second stage commences. Here, the feature extractor remains fixed, while the number of inducing points is amplified by a factor of 50. By retraining the DUE under these conditions, we obtain its enhanced version. The efficacy of this approach will be substantiated through experimental results.

3.6 INFLUENCE OF SPECTRAL NORMALIZATION ON NEW OBJECT DEFECT DETECTION

Throughout our research, we integrated Spectral Normalization as described in van Amersfoort et al. [2021]. By employing Spectral Normalization, we ensure the smoothness of the feature space which is beneficial for generalization to new objects. Consequently, it significantly elevates the predictive accuracy of our algorithm, especially when dealing with new objects. It's imperative to note that omitting Spectral Normalization would inevitably diminish the precision of our algorithm's predictions related to new objects.

4 EXPERIMENTS AND RESULTS

4.1 DATASET

We utilize the open-source dataset features fresh and rotten fruits, available at `https://data.mendeley.com/datasets/bdd69gyhv8/1` Sultana et al. [2022]. The dataset comprises:

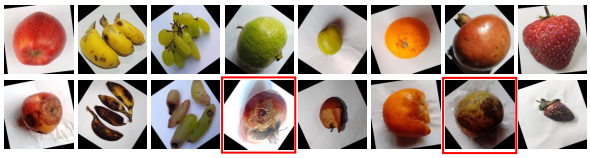

Figure 3: Examples of the input images. The top row displays fresh fruits, while the bottom row showcases rotten ones. Images of pomegranate and guava, highlighted with a red border, are excluded during the training phase for qualitative analysis experiments.

- 8 distinct fruit classes.
- Each class contains images of both fresh and rotten fruits, leading to 16 subclasses in total.
- Each subclass has 600 images.

Examples of these images are showcased in Figure 3. In our experiments, rotten fruits are treated as defective objects, fresh fruits are viewed as normal objects, 6 out of the 8 fruit classes are labeled as old objects, and the remaining 2 classes are labeled as new objects.

During the training phase, rotten fruits from new objects are intentionally excluded. However, they are reintroduced during the testing phase. Our guided anomaly detection method aims to detect rotten fruits from both old and new object samples. When uncertain, outputs an “Unknown".

4.2 EXPERIMENT CONFIGURATION AND OBJECTIVES

We employed the MobileNetV3 large model Koonce and Koonce [2021], pre-trained on the ImageNet dataset Deng et al. [2009], as our primary feature extractor $f_\theta(\cdot)$. The training was conducted over 50 epochs with an initial learning rate of 0.05. This rate was decayed at epochs 20, 30, and 40. Stochastic Gradient Descent (SGD) served as our optimization algorithm, with momentum and weight decay parameters set to 0.9 and 0.0001, respectively. All input images were standardized to a resolution of 224 $\times$ 224 pixels. The experiments were executed on a single NVIDIA Tesla V100 GPU.

The primary objectives of our experiments were to:

1. Demonstrate that the Guided Anomaly Detection (GAD) approach outperforms its non-guided counterpart (AD).
2. Show that the Gaussian Process Classifier (GPC) based GAD is superior to the Gaussian Mixture Model (GMM) based GAD, as referenced in Ma and Lui [2023].
3. Validate that the uncertainty measure U effectively indicates sample membership to a specific class, and can be translated into a defectiveness prediction.
4. Confirm that spectral normalization acts as a preventive measure against model overfitting, ensuring the model’s applicability to new defective object samples.
5. Illustrate the implementation of uncertainty-awareness by producing an "unknown" prediction in uncertain scenarios.
6. Record the computation time of our approach and compare it with previous methods to highlight efficiency improvements.

4.3 QUALITATIVE ANALYSIS

Evaluation Methods:

We utilized t-SNE visualizations [Van der Maaten and Hinton, 2008] to create 2-D representations of the feature embedding z. By overlaying the test samples on top of the training samples, we aimed to observe:

1. Whether normal samples cluster on a per-class basis.
2. Whether defective samples are either positioned at the boundaries of these clusters or are distant from them.

and then,

1. The performance of the guidance mechanism in ensuring separability.
2. The representation of uncertainty in the feature space.
3. The impact of spectral normalization on the feature distribution.

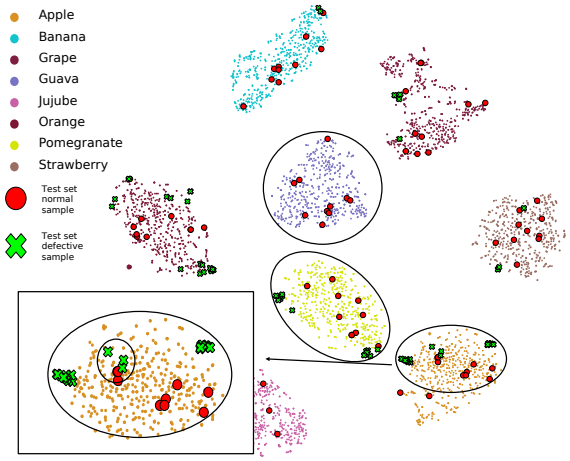

Figure 4: The t-SNE plot of the feature embedding z without employing the defect detector guidance. Some normal and defective samples are not separable. The image of the apple is highlighted with a circle for reference.

The Guidance Phenomenon:

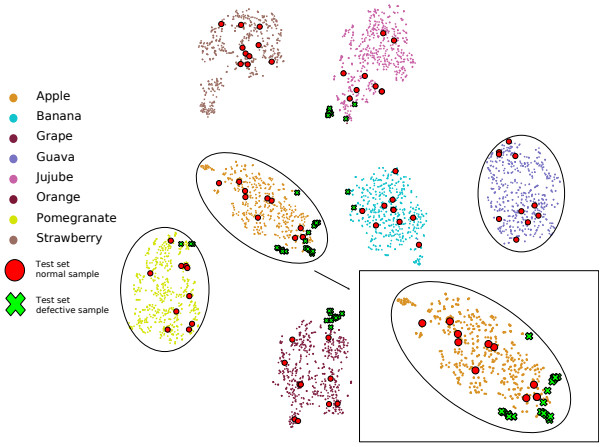

Figure 5: The t-SNE plot of the feature embedding z with defect detector guidance employed. Both normal and defective samples are separable. The image of the apple is highlighted with a circle for reference.

Initially, we trained the feature extractor $f_\theta(\cdot)$ and anomaly detector $\beta(\cdot)$ without the guidance of the defect detector $\alpha(\cdot)$, as depicted in the central pipeline of Figure 2. After training, we visualized the feature embedding z for all training and test samples using t-SNE, as depicted in Figure 4. This visualization reveals that some defective samples, located within clusters, mix with normal samples, rendering them inseparable.

Upon introducing guidance from the defect detector $\alpha(\cdot)$, Figure 5 demonstrates that defective samples are pushed to cluster boundaries, maximizing the separability between normal and defective samples. While some defective samples may still appear inseparable from normal samples, these instances are relatively few. We will further explore this observation through quantitative analysis in Section 4.4.

Uncertainty U for Uncertainty-awareness Detection:

Although the samples are visually separable, we should examine if the uncertainty measure U serves as an index for the separation purpose. We plotted the t-SNE map of z with uncertainty coloring. In Figure 6, all normal samples were shown as circles without a border line. The test samples, including both normal and defective samples, are shown as circles with a border line and crosses with a border line. We observe that normal samples are clustered and form exactly 8 islands in that K, the number of classes of old and new objects, is 8. Obviously, the normal samples are with 'low uncertainty colors'. The defective test samples are located on the border of the islands with 'high uncertainty' colors. Thus, we can use the uncertainty measure U for normal, defective, and unknown separation. When uncertainty is low, it is a normal sample; when uncertainty is high, it is a defective sample or unknown.

Spectral Normalization for Enhanced Generalization in

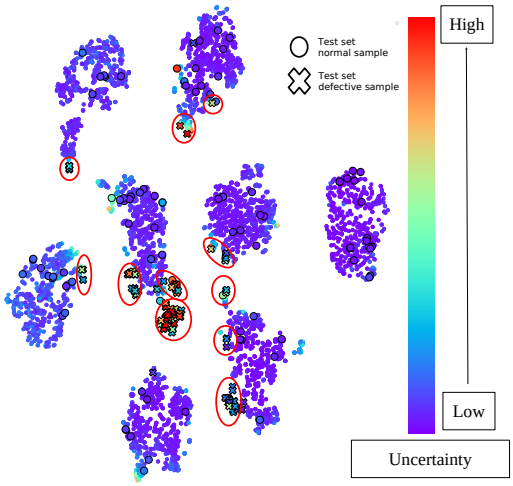

Figure 6: The t-SNE plot of z, the output of the feature extractor of the GAD. The color serves as an index of uncertainty. When uncertainty is low, it is a normal sample; when uncertainty is high, it is a defective sample or unknown. Thus, uncertainty U is a good tool enabling defect detection.

New Object Detection::

The guided anomaly detection model is initially trained on a dataset comprising of $x_{old,normal}$, $x_{old,defective}$, and $x_{new,normal}$. Without regularization, the model may overfit to this seen data and lose its ability to generalize to unseen data $x_{new,normal}$. To demonstrate this phenomenon, we conducted a simulation without spectral normalization and report the qualitative analysis. Figure 7 illustrates that without spectral normalization, all defective samples form a new cluster far from the true clusters. In the supplementary material, we present the uncertainty map. The uncertainty measure U is very low, rendering it ineffective as a defect detection index.

With the application of spectral normalization regularization, Figure 5 reveals that defective samples are positioned at the border of the object clusters. Some samples are slightly distant from the object clusters, and those distant ones exhibit high uncertainty. High uncertainty results in a 'Unknown' label assignment. As a result, with spectral normalization regularization, the uncertainty measure shows promise as a defect detection index.

4.4 QUANTITATIVE ANALYSIS AND ABLATION STUDY

We conducted settings for quantitative analysis and ablation study. Every setting involves 3 randomly initialized experiments. We randomly selected 2 out of 8 classes as new objects for each setting. The data split was also carried out randomly. As a result, there were 3 independent experiments

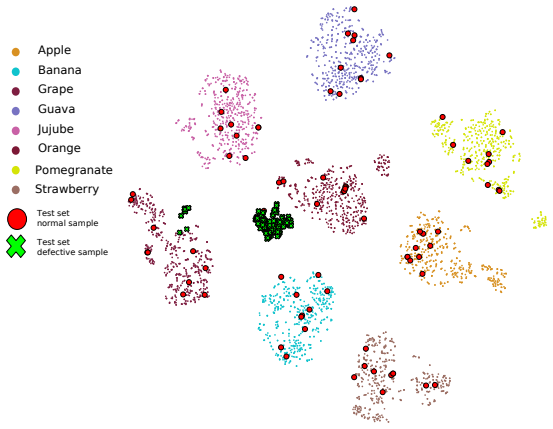

Figure 7: The t-SNE plot of the output of the feature extractor when the guidance is employed but the spectral normalization regularization is not employed. The defective samples forms a solid cluster. Indicating overfit and ineffective uncertainty measure.

for each setting.

Evaluation Metrics: For fair comparisons with Ma and Lui [2023], we also utilize overkill, leakage, and unknown rates as our measurement standards. The harmonic score takes all these three standards into account in a balanced manner as defined in Ma and Lui [2023]. These rates are expressed as ratios to the overall number of test samples. We report the results of different settings to see

1. How much does the GPC based GAD out perform GMM based GAD?
2. How much performance does our enhanced GPC gain?
3. How much does spectral normalization affect our method?

Gaussian Process Classifier vs. Gaussian Mixture Model:

Table 1 displays the overkill, leakage, unknown, overkill plus leakage rates, and the harmonic score over all test samples. The test samples include both normal and defective samples for both old and new objects. We see that the GMM based GAD (aka deepDG3, GMM-GAD) performs worse than our GPC based guided anomaly detector(GPC-GAD). The harmonic score of GMM-GAD is 0.93403. The harmonic score of GPC-GAD is 0.95187. This is a 1.93% improvement. The unknown rate of GMM-GAD is about 18.47%. The unknown rate of GPC-GAD is about 12.84%. This is a 30.5% improvement. The overkill plus leakage rate of GMM-GAD is about 2.967%. The overkill plus leakage rate of GPC-GAD is about 2.4%. This is a 19.1% improvement. Moreover, the computation efficiency is improved as discussed in Section 4.5.

Guided Anomaly Detection vs. non-Guided Anomaly Detection:

From Table 1, we can also calculate the improvement of Guided Anomaly Detection over non-Guided Anomaly Detection. The harmonic score of GPC-AD is 0.93764. The harmonic score of GPC-GAD is 0.95187. This is a 1.49% improvement. Moreover, GPC-AD has more leakages and less unknowns than GPC-GAD. We attribute this result to the fact that anomaly is not the same concept as defect. Directly taking anomaly as defect causes poorer defect detection performance.

Enhancement of GPC with 50x inducing points:
Table 2 displays the improvement of using stage 2 training with 50x inducing points. We observe that all indexes are improved. Overall, the harmonic score is improved from 0.94 to 0.95, which is equivalent to 1.06% improvement.

Spectral Normalization in Old and New Object:
Table 3 and Table 4 together display that the harmonic scores of the new object test set are worse than those of the old object test set. While for the old object test set, the spectral normalization shows negligible improvement. For the new object test set, the spectral normalization shows improvement from 0.89 to 0.91 which is equivalent to 1.74% improvement. For old objects, both normal and defective samples are available in the training stage. Hence, feature collapse does not deteriorate the performance. When a new object is considered, the defective new object samples are out of distribution. Feature collapse might prevent the model from generalizing to these unseen data. The spectral normalization avoids feature collapse for these unseen data, resulting in performance improvement.

4.5 COMPUTATION EFFICIENCY

Enhanced Training Speed:

- *Previously proposed GMM-GAD (aka deepGD3 Ma and Lui [2023]:* Approximately 8.57 ± 1.36 hours (with Bayesian optimization over 50 iterations taking 16.86 ± 0.9 hours).
- *Our Method:* Approximately 6.834 ± 2.21 hours (with Bayesian optimization over 200 iterations taking 15.81 ± 1.79 hours).

Improved Inference Speed:

- *Previously proposed GMM-GAD's (aka deepGD3 Ma and Lui [2023]'s) Detection Time:* 25.51 ± 3.66 minutes.
- *Our Method's Detection Time:* 5.947 ± 2.229 minutes, resulting 4.2 times speed up.

5 CONCLUSION

Anomaly detection and defect detection are distinct tasks. Nonetheless, it is possible to guide an anomaly detector

Table 1: Comparison of the performances of GMM-GAD, GPC-AD, and GPC-GAD.

Architecture	Overkill rate (%)↓	Leakage rate (%)↓	Unknown rate (%)↓	Overkill+Leakage rate (%)↓	Harmonic score↑ Ma and Lui [2023]
Base line, deepGD3 (GMM based GAD) Ma and Lui [2023]	2.967 (±0.12%)	0 (±0%)	18.47 (±10.6%)	2.967 (±1.2%)	0.93403
GPC based Anomaly Detector (GPC-AD)	3.452 (±1.20%)	14.132 (±2.87%)	2.158 (±2.21%)	17.58 (±3.49%)	0.93764
Ours, GPC based Guided Anomaly Det. (GPC-GAD)	1.187 (±0.65%)	1.214 (±0.79%)	12.84 (±1.91%)	2.40 (±1.43%)	**0.95187**

Table 2: Enhancement of GPC with 50x inducing points trained at stage 2.

Architecture	Overkill rate (%)↓	Leakage rate (%)↓	Unknown rate (%)↓	Overkill+Leakage rate (%)↓	Harmonic score↑ Ma and Lui [2023]
Number of inducing points = K (stage 1 only)	1.268 (±0.44%)	2.481 (±2.15%)	15.13 (±3.21%)	3.749 (±1.71%)	0.94083
Number of inducing points = $50 \times K$	1.187 (±0.65%)	1.214 (±0.79%)	12.84 (±1.91%)	2.4 (± 1.43%)	**0.95189**

Table 3: Evaluating the performance of different spectral normalization coefficients on new object test set, which include $x_{new,normal}$ and $x_{new,defective}$.

Architecture	Overkill rate (%)↓	Leakage rate (%)↓	Unknown rate (%)↓	Overkill + Leakage rate (%)↓	Harmonic score↑
SN coeff=1	2.017 (±0.62%)	4.313 (±3.51%)	25.36 (±1.89%)	6.331 (±4.13%)	0.9046
SN coeff=3	1.79 (±1.26%)	4.17 (±0.98%)	27.61 (±6.12%)	5.960 (±1.52%)	0.90006
SN coeff=5	1.603 (±0.84%)	4.751 (±3.01%)	21.2 (±5.06%)	6.354 (±3.42%)	**0.91556**
SN coeff=7	2.885 (±0.34%)	3.263 (±2.49%)	24.15 (±4.05%)	6.148 (±2.23%)	0.90831
without SN	0.859 (±0.49%)	1.586 (±2.48%)	32.28 (±9.91%)	2.444 (±2.83%)	0.89995

Table 4: Evaluating the performance of different spectral normalization coefficients on old object test set, which include $x_{old,normal}$ and $x_{old,defective}$.

Architecture	Overkill rate (%)↓	Leakage rate (%)↓	Unknown rate (%)↓	Overkill + Leakage rate (%)↓	Harmonic score↑
SN coeff=1	1.446 (±0.45%)	0 (±0%)	9.99 (±1.86%)	1.446 (± 0.45%)	0.96353
SN coeff=3	1.335 (±0.22%)	0.072 (±0.06%)	14.34 (±6.85%)	1.407 (± 0.18%)	0.95081
SN coeff=5	1.05 (±1.26%)	0 (±0%)	10.004 (±3.56%)	1.05 (± 1.26%)	0.96479
SN coeff=7	1.299 (±0.28%)	0.036 (±0.06%)	7.47 (±2.01%)	1.335 (± 0.22%)	**0.97158**
without SN	0.542 (±0.11%)	0 (±0%)	9.142 (±3.76%)	0.542 (± 0.11%)	**0.96908**

for the purpose of defect detection. Various strategies exist for constructing anomaly detectors, and among them, the Gaussian process classifier emerges as a favorable choice for introducing uncertainty-awareness, as highlighted by Liu et al. Liu et al. [2021]. To adapt the Gaussian process classifier for large datasets, we employ inducing points. Contrary to prior beliefs that increasing the number of inducing points offers limited benefits van Amersfoort et al. [2021], our method demonstrates performance improvements with a larger set of these points. Our system's uncertainty-awareness prediction ability extends to new objects, even when defective samples are absent during the training phase. The incorporation of spectral normalization as a network regularization technique further ensures that our method generalizes effectively to unseen defective samples in new objects. While our experiments primarily utilize fruit data to showcase our method's efficacy, its potential applications span industrial, medical, and agricultural settings, to name a few.

In terms of future research directions, we envision enhancing the defect detection algorithm to also encompass defect classification. Additionally, we aim to address the data imbalance issue and explore applications in real industrial settings.

Code and Data Availability: Upon the paper's acceptance, we will release the code and data on GitHub for public access.

References

M Abd Al Rahman and Alireza Mousavi. A review and analysis of automatic optical inspection and quality monitoring methods in electronics industry. *Ieee Access*, 8: 183192–183271, 2020.

Prahar M Bhatt, Rishi K Malhan, Pradeep Rajendran, Brual C Shah, Shantanu Thakar, Yeo Jung Yoon, and Satyandra K Gupta. Image-based surface defect detection using deep learning: A review. *Journal of Computing and Information Science in Engineering*, 21(4):040801, 2021.

John Bradshaw, Alexander G de G Matthews, and Zoubin Ghahramani. Adversarial examples, uncertainty, and transfer testing robustness in gaussian process hybrid deep networks. *arXiv preprint arXiv:1707.02476*, 2017.

Raghavendra Chalapathy and Sanjay Chawla. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019.

Tamás Czimmermann, Gastone Ciuti, Mario Milazzo, Marcello Chiurazzi, Stefano Roccella, Calogero Maria Oddo, and Paolo Dario. Visual-based defect detection and classification approaches for industrial applications—a survey. *Sensors*, 20(5):1459, 2020.

Wenting Dai, Abdul Mujeeb, Marius Erdt, and Alexei Sourin. Soldering defect detection in automatic optical inspection. *Advanced Engineering Informatics*, 43: 101004, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Zhen Fang, Yixuan Li, Jie Lu, Jiahua Dong, Bo Han, and Feng Liu. Is out-of-distribution detection learnable? *Advances in Neural Information Processing Systems*, 35: 37199–37213, 2022.

Mark N Gibbs. *Bayesian Gaussian processes for regression and classification*. PhD thesis, Citeseer, 1998.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 41–50, 2019.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Dan Hendrycks and Mantas Mazeika. X-risk analysis for ai research. *arXiv preprint arXiv:2206.05862*, 2022.

Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Unsolved problems in ml safety. *arXiv preprint arXiv:2109.13916*, 2021.

James Hensman, Alexander Matthews, and Zoubin Ghahramani. Scalable variational gaussian process classification. In *Artificial Intelligence and Statistics*, pages 351–360. PMLR, 2015.

Weibo Huang, Peng Wei, Manhua Zhang, and Hong Liu. Hripcb: a challenging dataset for pcb defects detection and classification. *The Journal of Engineering*, 2020(13): 303–309, 2020.

Like Hui, Mikhail Belkin, and Preetum Nakkiran. Limitations of neural collapse for understanding generalization in deep learning. *arXiv preprint arXiv:2202.08384*, 2022.

Brett Koonce and Brett Koonce. Mobilenetv3. *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pages 125–144, 2021.

Vignesh Kothapalli, Ebrahim Rasromani, and Vasudev Awatramani. Neural collapse: A review on modelling principles and generalization. *arXiv preprint arXiv:2206.04041*, 2022.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

Qin Ling and Nor Ashidi Mat Isa. Printed circuit board defect detection methods based on image processing, machine learning and deep learning: A survey. *IEEE Access*, 2023.

Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in Neural Information Processing Systems*, 33:7498–7512, 2020.

Yehao Liu, Matteo Pagliardini, Tatjana Chavdarova, and Sebastian U Stich. The peril of popular deep learning uncertainty estimation methods. *arXiv preprint arXiv:2112.05000*, 2021.

Ching-Wen Ma and Yanwei Lui. DeepGD3: Unknown-aware deep generative/discriminative hybrid defect detector for PCB soldering inspection. In Robin J. Evans

and Ilya Shpitser, editors, *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pages 1326–1335. PMLR, 31 Jul–04 Aug 2023. URL https://proceedings.mlr.press/v216/ma23a.html.

Alexander Meinke and Matthias Hein. Towards neural networks that provably know when they don't know. *arXiv preprint arXiv:1909.12180*, 2019.

Sina Mohseni, Mandar Pitale, Vasu Singh, and Zhangyang Wang. Practical solutions for machine learning safety in autonomous vehicles. *arXiv preprint arXiv:1912.09630*, 2019.

Sina Mohseni, Haotao Wang, Zhiding Yu, Chaowei Xiao, Zhangyang Wang, and Jay Yadawa. Practical machine learning safety: A survey and primer. *arXiv preprint arXiv:2106.04823*, 4, 2021.

Sina Mohseni, Haotao Wang, Chaowei Xiao, Zhiding Yu, Zhangyang Wang, and Jay Yadawa. Taxonomy of machine learning safety: A survey and primer. *ACM Computing Surveys*, 55(8):1–38, 2022.

Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2):1–38, 2021.

Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020.

Waseem Rawat and Zenghui Wang. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449, 2017.

Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. *Advances in neural information processing systems*, 32, 2019.

Zhonghe Ren, Fengzhou Fang, Ning Yan, and You Wu. State of the art in defect detection based on machine vision. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 9(2):661–691, 2022.

Nusrat Sultana, Musfika Jahan, and Mohammad Shorif Uddin. An extensive dataset for successful recognition of fresh and rotten fruits. *Data in Brief*, 44:108552, 2022.

Yanan Sun, Bing Xue, Mengjie Zhang, and Gary G. Yen. Evolving deep convolutional neural networks for image classification. *IEEE Transactions on Evolutionary Computation*, 24(2):394–407, 2020. doi: 10.1109/TEVC.2019.2916183.

Andrei-Alexandru Tulbure, Adrian-Alexandru Tulbure, and Eva-Henrietta Dulf. A review on modern defect detection models using dcnns–deep convolutional neural networks. *Journal of Advanced Research*, 35:33–48, 2022.

Dennis Ulmer. A survey on evidential deep learning for single-pass uncertainty estimation. *arXiv preprint arXiv:2110.03051*, 2021.

Joost van Amersfoort, Lewis Smith, Andrew Jesson, Oscar Key, and Yarin Gal. On feature collapse and deep kernel learning for single forward pass uncertainty. *arXiv preprint arXiv:2102.11409*, 2021.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008.

Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. Deep kernel learning. In *Artificial intelligence and statistics*, pages 370–378. PMLR, 2016.

Yanan Wu, Keqing He, Yuanmeng Yan, QiXiang Gao, Zhiyuan Zeng, Fujia Zheng, Lulu Zhao, Huixing Jiang, Wei Wu, and Weiran Xu. Revisit overconfidence for ood detection: Reassigned contrastive learning with adaptive class-dependent threshold. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4165–4179, 2022.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*, 2021.

Guided Gaussian Process-Based Anomaly Detectors for Uncertainty-Aware Defect Detection in New Objects (Supplementary Material)

A PREDICTION CONVERTER

In the evolving landscape of deep learning research, the transformation of raw predictions into meaningful actionable insights remains pivotal. In this supplementary material, we provide an in-depth understanding of how our novel approach effectively harnesses the outputs of the Anomaly Detector to yield defect predictions. We introduce the concept of the **Prediction Converter** that employs a dual-threshold mechanism tailored for each component, facilitating a nuanced defect detection. Further, we elucidate the procedure to determine these thresholds and the intricate algorithmic details that underpin the entire conversion process. By diving into these nuances, this supplementary content aims to offer clarity, ensuring that researchers and practitioners alike can appreciate the depth, robustness, and innovations of our proposed method.

A.1 INTRODUCTION

To convert the output of the Anomaly Detector into a defect prediction, we have designed a **Prediction Converter**. We use two thresholds, T_j^1 and T_j^2, for each component. These thresholds are utilized to determine the type of defect as depicted in Figure 8. The prediction method is as follows:

$$\hat{y}_{def}^{a} = \begin{cases} Unknown, & \text{if } U > T_j^2 \\ defective, & \text{if } T_j^1 < U \leq T_j^2 \\ normal, & \text{if } U \leq T_j^1 \end{cases}$$

A.2 THRESHOLD DETERMINATION PROCESS

To obtain these two thresholds, we list the details of the process below.

- Calculate the predicted probability (p_i) of samples from training set and validation set with trained model.
- Estimate the normalized uncertainty (U) of the samples from training set and validation set using the predicted probability(p_i). $U = \sum_{i=1}^{O}(-P_i \cdot log(P_i) \div log(o))$, O : Number of class output
- Find T_j^1, T_j^2 optimized by Bayesian optimizer that produces the best harmonic score according to the overkill rate, leakage rate, and unknown rate in the training and validation sets.

A.3 DEFECT PREDICTION CONVERSION AND ALGORITHMIC PROCEDURE

In the above way, we successfully designed a method to convert the output $p(y_{obj}|z)$ of the original component classification into the defect prediction $\hat{y}_{def}^{a}$, and the procedure is described in Algorithm 1.

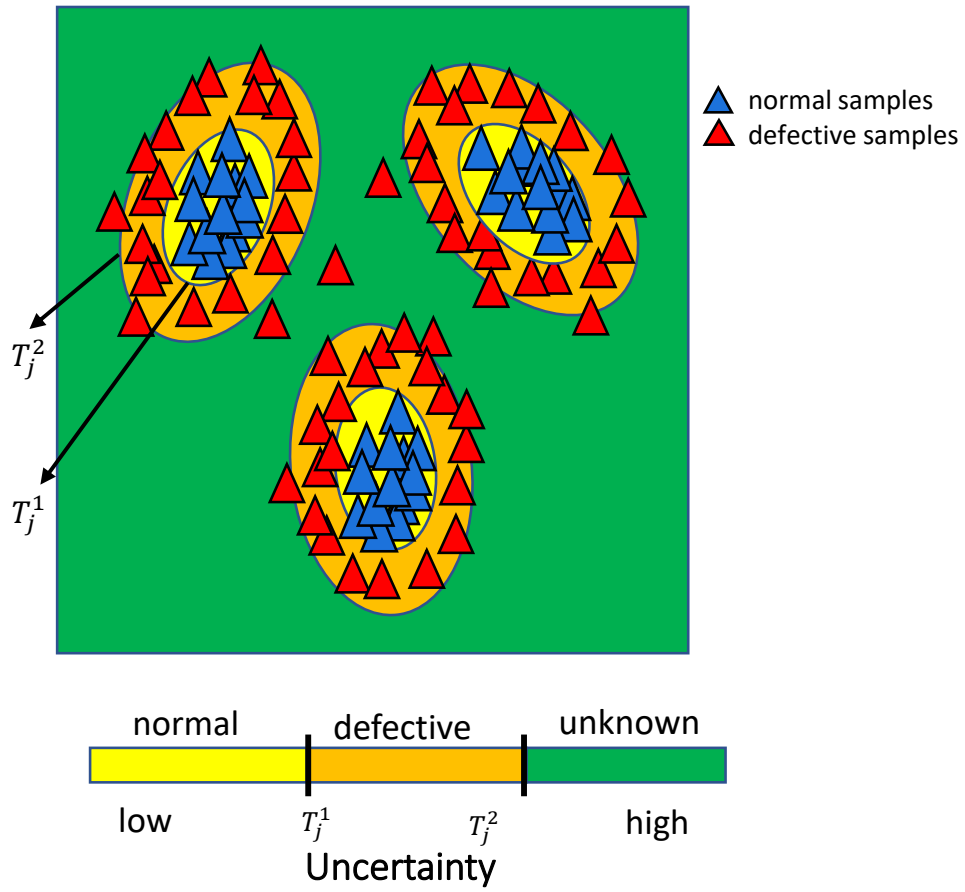

Figure 8: Visualization of Defect Classification Based on Uncertainty Levels

B WITHOUT SPECTRAL NORMALIZATION

If spectral normalization is not used, a greater separation between the positive and negative samples will occur. We analyze this through two images: Figure 9 and Figure 10. From these figures, it can be determined that our algorithm is unable to effectively detect defects under such conditions.

Algorithm 1 Threshold Calculation

Input: Trained model, training set X_{train}, validation set X_{val}
Output: Thresholds T_j^1, T_j^2 for each component j
Initialize: T_j^1, T_j^2
Step 1: Calculate predicted probability
    $P_{\text{train}} \leftarrow \text{predict\_probability}(\text{model}, X_{\text{train}})$
    $P_{\text{val}} \leftarrow \text{predict\_probability}(\text{model}, X_{\text{val}})$
Step 2: Estimate uncertainty
    $U_{\text{train}} \leftarrow \text{calculate\_uncertainty}(P_{\text{train}})$
    $U_{\text{val}} \leftarrow \text{calculate\_uncertainty}(P_{\text{val}})$
Step 3: Optimize T_j^1 and T_j^2
    $T_j^1, T_j^2 \leftarrow \text{optimize\_thresholds}(\text{model}, U_{\text{train}}, U_{\text{val}})$
    return T_j^1, T_j^2

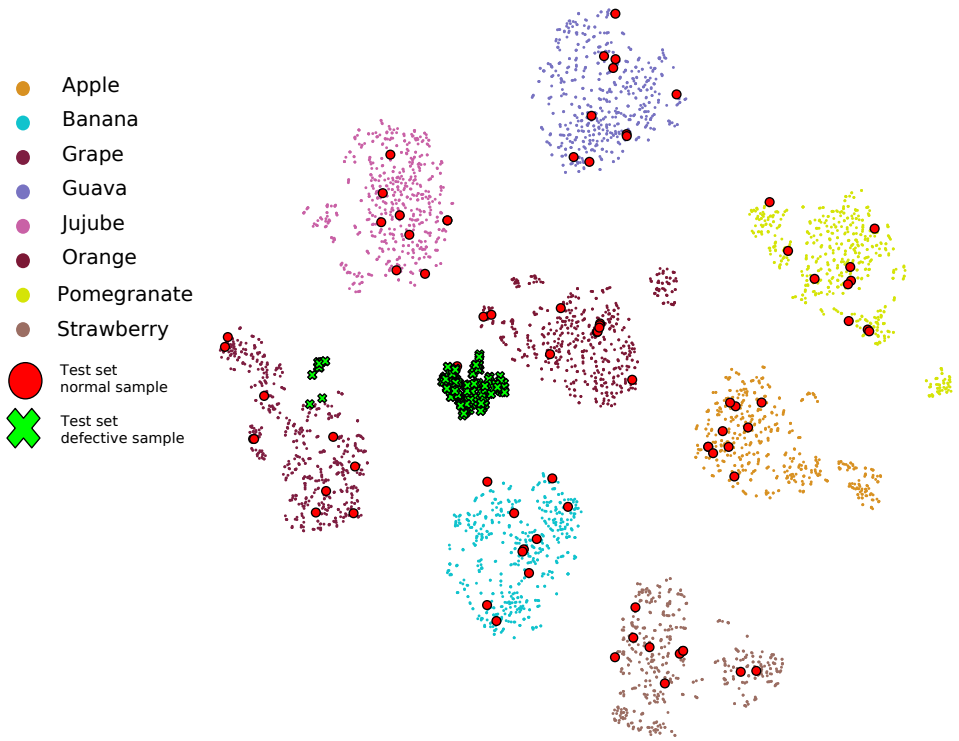

Figure 9: The t-SNE [Van der Maaten and Hinton, 2008] plot of the feature embedding z when the defect detector guidance is employed. This plot is the t-SNE diagram without using spectral normalization. It can be observed that without spectral normalization, normal samples and defective samples are distinctly separated.

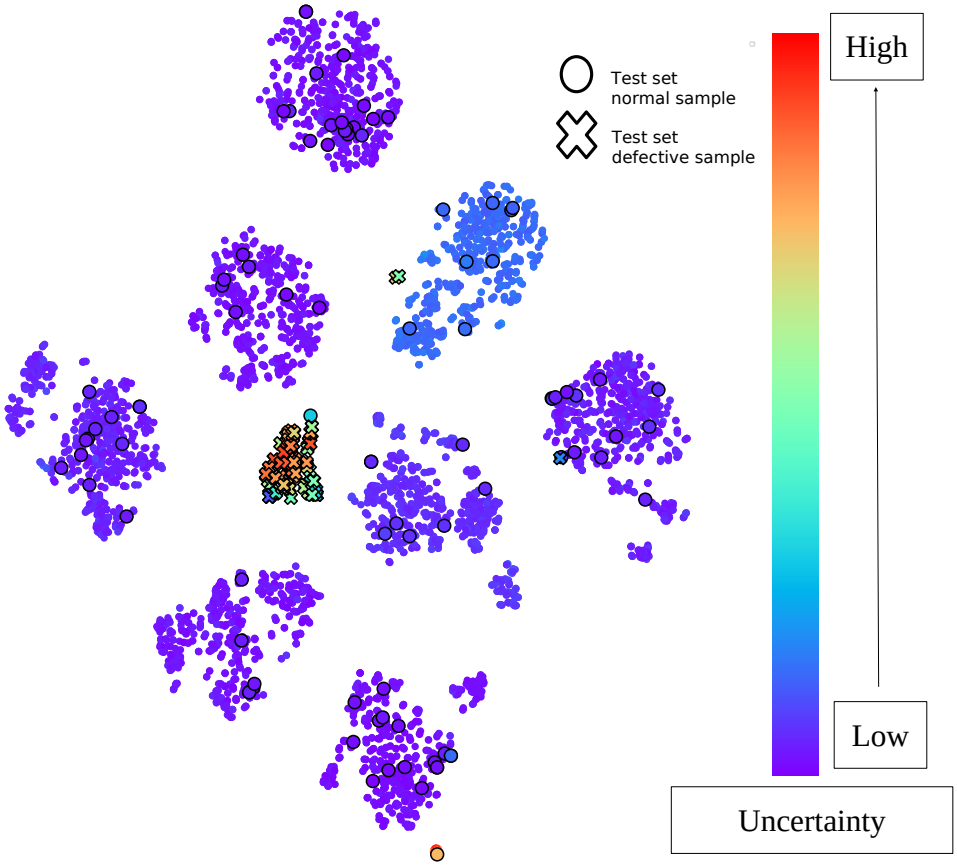

Figure 10: This plot is the uncertainty map without using spectral normalization. It can be observed that without spectral normalization, the uncertainty of the defective samples is very large. Furthermore, all defective samples are clustered together. Under such conditions, it's impossible to determine through the Prediction Converter which are defective samples and which are unknown samples.