

# SolderNet: Towards Trustworthy Visual Inspection of Solder Joints in Electronics Manufacturing Using Explainable Artificial Intelligence

Hayden Gunraj<sup>1,2</sup>, Paul Guerrier<sup>3</sup>, Sheldon Fernandez<sup>2</sup>, and Alexander Wong<sup>1,2</sup>

<sup>1</sup>Vision and Image Processing Research Group, University of Waterloo, Waterloo, Canada

<sup>2</sup>DarwinAI Corp., Waterloo, Canada

<sup>3</sup>Moog Inc., New York, USA

{hayden.gunraj, a28wong}@uwaterloo.ca

## Abstract

In electronics manufacturing, solder joint defects are a common problem affecting a variety of printed circuit board components. To identify and correct solder joint defects, the solder joints on a circuit board are typically inspected manually by trained human inspectors, which is a very time-consuming and error-prone process. To improve both inspection efficiency and accuracy, in this work we describe an explainable deep learning-based visual quality inspection system tailored for visual inspection of solder joints in electronics manufacturing environments. At the core of this system is an explainable solder joint defect identification system called **SolderNet** which we design and implement with trust and transparency in mind. While several challenges remain before the full system can be developed and deployed, this study presents important progress towards trustworthy visual inspection of solder joints in electronics manufacturing.

## Introduction

In electronics manufacturing, solder joint defects are a common problem affecting both surface-mounted and through-hole components of printed circuit boards (PCBs) (see Figure 1 for example solder joint defects). Defects introduced in the soldering process can lead to electrical issues and faulty parts, especially if not caught early in the process. This is particularly concerning in critical applications such as the aerospace and medical industries, where defective PCBs can cause catastrophic failures in critical systems.<sup>1</sup> If defects are caught early, the solder can be reworked to minimize potential issues later in the manufacturing process and avoid unnecessary electronic waste.

To identify and correct solder joint defects, the solder joints on a PCB are often visually inspected by trained inspectors. However, human inspectors are estimated to make visual inspection errors in 20-30% of cases, and the performance of human inspectors varies considerably depending on experience, mental fatigue, defect type, frequency of defect occurrence, and a variety of other factors (See et al. 2017; Klamklay and Bishu 1998). Manual inspection is also

Copyright © 2023, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

<sup>1</sup>This paper consists of general capabilities information that is not defined as controlled technical data under ITAR Part 120.10 or EAR Part 772.

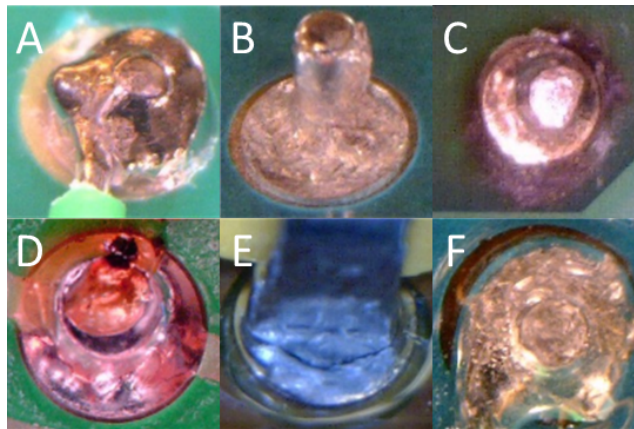


Figure 1: Example images of solder joint defects from the dataset examined in this study: (A) fractured joint, (B) cold joint, (C) burns, (D) flux residue, (E) poor wetting, and (F) disturbed solder.

expensive and time-consuming, often requiring multiple inspectors to achieve reasonable throughput.

In contrast to manual inspection, automated visual inspection of solder joints offers many benefits, such as high throughput, high performance, and zero fatigue. However, automating this task is technically challenging due to the small size of the joints, the wide variety of possible joint and defect types, limited computing resources, limited inspection time, and the need for high performance. Deep neural networks offer an attractive solution to this problem, as they have been successfully applied to detection and inspection tasks in a variety of manufacturing scenarios (Kim et al. 2021; Zhang et al. 2022; Westphal and Seitz 2021; Bhatt et al. 2021; Yang et al. 2020). While deep neural networks are capable of achieving human-level performance in visual inspection tasks, they have several drawbacks in critical manufacturing scenarios:

1. **Computational complexity:** high-performance neural networks are often compute-hungry and require specialized hardware for fast inference. When computing resources are limited (as in many manufacturing scenarios), such networks can be slow and reduce throughput.

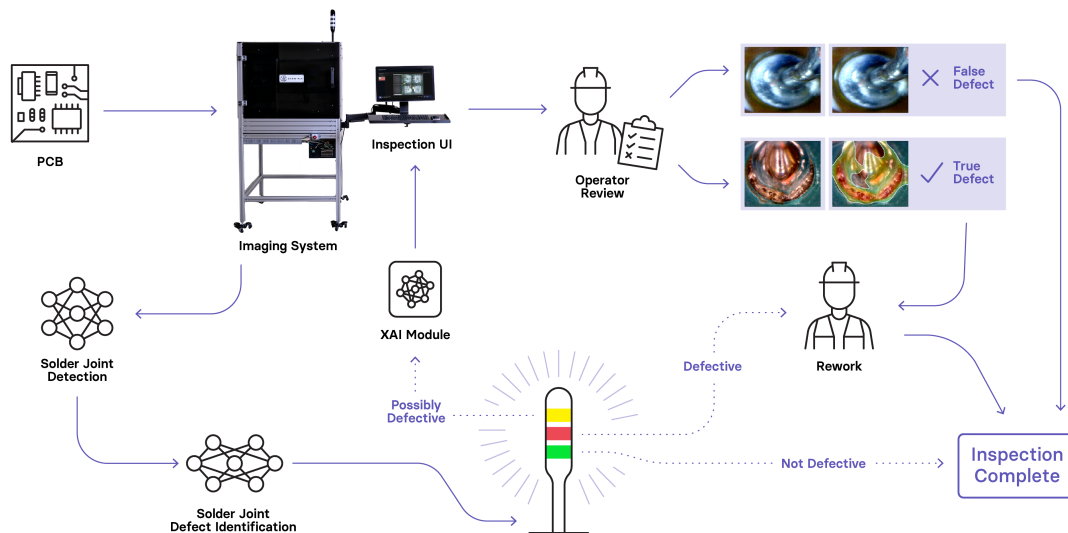


Figure 2: Overview of the proposed solder joint inspection system. This study focuses on developing the core defect identification system (SolderNet), which comprises the defect identification and XAI modules.

2. **Lack of explanations:** neural networks typically operate as black boxes which provide a prediction but cannot explain how the prediction was made. In critical inspection scenarios, determining why a network made a particular prediction falls to the human inspector, which reduces the potential throughput benefits offered by these networks.
3. **Unclear reliability:** the development of neural networks usually involves testing using data not seen by the network. Despite this, a common challenge when deploying deep neural networks is understanding when a model's predictions can be trusted and which scenarios a model is likely to fail in.

In this work, we outline a trustworthy, explainable deep learning-driven solder joint inspection system for electronics manufacturing. The proposed system is capable of handling both through-hole and surface-mounted components and can provide explanations of its decision in order to facilitate manual review when necessary. We perform a number of experiments to implement and test the core functionality of this system with a focus on practical considerations for manufacturing settings. Moreover, we leverage a mix of quantitative explainability and trust quantification techniques to further analyze the behaviour and trustworthiness of the system, identify gaps in the model development process, and provide insights during the inspection process.

## Methods

The proposed system consists of several stages which are designed to provide high performance, high throughput and high reliability. Figure 2 illustrates the inspection system, which proceeds via the following steps:

1. Images of a PCB are captured by an imaging system.
2. PCB images are passed through a solder joint detector which identifies and extracts solder joint images.

3. Solder joint images are passed through a defect identification network which categorizes solder joints as defective, possibly defective, or non-defective and activates the appropriate indicator.
4. For boards with possibly defective joints, an XAI algorithm is used to generate explanations of the network's predictions.
5. A human operator reviews the possibly defective joints and their XAI visualizations to determine which are truly defective and which are not defective.
6. All defective solder joints are reworked.

In this study, we focus on defect identification, explainability, and trustworthiness. These three aspects of the system are described in detail in the following subsections.

## Solder Joint Defect Identification

We focus on convolutional neural networks (CNNs) to perform the task of solder joint defect identification. CNNs are a ubiquitous technology in image identification tasks, and recent advances in network design have enabled the creation of high-performance networks which remain computationally efficient. We examined both efficiency-focused architectures and performance-focused architectures to examine their respective trade-offs between network complexity and network performance. Specifically, we tested Attend-NeXt (Small and Large) (Wong et al. 2022), MobileNet (V2 and V3 Small)(Sandler et al. 2018; Howard et al. 2019), ShuffleNetV2 (Ma et al. 2018), and ConvNeXt (Tiny, Small, and Base) (Liu et al. 2022). Table 1 shows the number of parameters and floating-point operations (FLOPs) for each of the network architectures examined in this work. We train each of the aforementioned architectures as binary classifiers which provide a confidence score on the interval  $[0, 1]$  indicating the network's confidence that an input image

represents a defective solder joint. In deployment, this allows us to define confidence regions which correspond to definite defects (requiring repair), possible defects (requiring review), and definite non-defects (no action required).

### Quantitative Explainability

Following defect identification, as can be seen in Figure 2 some solder joints may require review by a human operator given a level of uncertainty during the solder joint defect identification process with respect to whether that joint is indeed defective. In such scenarios, while the identification network’s prediction and confidence provide useful information to the operator, the operator must still identify the particular defects (or lack thereof) in the solder joint images to determine if it is a false defect detection (in which case inspection is complete) or if it is indeed a true defect detection (in which case the solder needs to be reworked). To make this task easier and faster for operators, we propose a quantitative explainability module which identifies the critical factors in an image which led to the neural network’s decision in a quantitative manner, allowing the operator to rapidly identify the locations of potential defects and validate the model’s predictions.

To this end, we introduce an extended form of GSInquire (Lin et al. 2019) to provide visual explanations of a neural network’s decision-making process. We choose GSInquire as the core approach to extend upon because it identifies specific critical factors that quantitatively impact the decisions made by the deep neural network, in contrast to other explainability methods such as Grad-CAM (Selvaraju et al. 2017), Expected Gradients (Erion et al. 2021), LIME (Ribeiro, Singh, and Guestrin 2016), and SHAP (Lundberg and Lee 2017) that generate only qualitative heatmaps depicting relative importance.

In its original form, GSInquire examines a network’s activation signals in response to an input image and uses them to identify critical factors within the image which impact the network’s decision in a quantitatively significant way. These critical factors may then be projected into the same space as the image to produce a visual interpretation. Building upon GSInquire, we introduce an extension which determines the relative importance of different aspects within the critical factors of a given image. This allows for more fine-grained interpretation of the critical factors, as we can now see not only which critical factors contribute the most to the neural network’s decision-making process but also which aspects of a particular critical factor are most important. We present the relative importance of different aspects within the critical factors as regional heatmap overlays within the boundaries of their corresponding critical factors, as shown in Figure 3.

### Second-order Explainability

In addition to providing visual explanations in deployment settings, visual explainability is also a valuable model validation tool during development. While quantitative metrics such as accuracy provide important measures of a deep neural network’s performance, they do not provide information regarding how decisions are made. To address this gap in performance analysis, visual XAI enables auditing of a

Architecture	Parameters (M)	FLOPs (G)
Attend-NeXt Small	1.632	0.423
Attend-NeXt Large	3.897	0.861
MobileNetV2	2.225	0.409
MobileNetV3 Small	1.519	0.076
ShuffleNetV2	1.255	0.193
ConvNeXt Tiny	27.821	5.832
ConvNeXt Small	49.455	11.364
ConvNeXt Base	87.567	20.084

Table 1: Comparison of architectural and computational complexity for each of the tested network architectures. FLOPs were measured with a  $256 \times 256$  3-channel input and a batch size of 1.

model’s decisions during development to ensure that they are based on relevant visual indicators and elucidate potential biases in the training data, which may then be used to guide improvements to the training framework. However, reviewing visual explanations manually is a time-consuming task, particularly for large-scale image datasets with many classes or high intra-class variability. Manual review may also be influenced by human biases, and it can be challenging if not intractable to mentally conceptualize key trends and patterns based on the individual explanations at hand.

To facilitate auditing of the model and dataset during development, we introduce the concept of second-order explainable artificial intelligence (SOXAI) which extends the concept of XAI from the sample level to the dataset level. Rather than manually reviewing visual explanations to explore patterns in a model’s decision-making behaviour, SOXAI aims to reveal these patterns automatically through analysis of the relationships between quantitative explanations. This allows for rapid identification of the most common visual concepts leveraged by a model when making its decisions, and can reveal obvious model and dataset biases. This can also increase transparency by helping users understand what a model has learned and what it has not. In essence, SOXAI enables us to “explain explainability” by providing higher-level interpretations of the behaviours of deep neural networks.

In this study, we formulate SOXAI as an embedding problem: given an image  $I$  and corresponding quantitative explanation  $\alpha$ , we define an embedding  $f : (I, \alpha) \rightarrow \mathbf{R}^N$  which embeds the regions of  $I$  indicated by  $\alpha$ . Performing this embedding for all images in a dataset allows for similar embeddings to be grouped together with secondary algorithms. To generate the visualizations presented in this work, we leverage t-distributed stochastic neighbour embedding (t-SNE) (van der Maaten and Hinton 2008) to group the embeddings and map them to a 2-dimensional space.

### Trust Quantification

In industrial applications, it is important to understand how trustworthy a model is in order for it to be deployed as an automation tool. Quantifying trust allows for models to be compared in terms of their trustworthiness during the model development process, and may help guide decisions as to

Architecture	Latency (s)	Accuracy (%)	Overkill (%)	Escape (%)	NetTrustScore
Attend-NeXt Small	<b>0.275</b>	86.6	8.4	5.0	0.864
Attend-NeXt Large	0.430	<b>91.1</b>	<b>5.0</b>	3.9	<b>0.904</b>
MobileNetV2	4.082	88.5	7.8	<b>3.7</b>	0.878
MobileNetV3 Small	1.647	88.3	7.1	4.6	0.883
ShuffleNetV2	1.775	87.0	7.1	5.9	0.867
ConvNeXt Tiny	5.975	88.8	7.2	3.9	0.887
ConvNeXt Small	12.027	89.6	5.8	4.6	0.890
ConvNeXt Base	17.575	88.8	6.3	4.8	0.889

Table 2: Comparison of quantitative performance metrics for each of the network architectures on the solder joint test dataset. Latency was measured on an ARM Cortex-A72 with a  $256 \times 256$  3-channel input and a batch size of 1, and is reported as the average of 100 measurements taken after 20 warm-up runs. Accuracy, overkill, and escape were calculated at a confidence threshold of 0.5. Best results highlighted in **bold**.

the level of human supervision and review required once a model is deployed. Recent studies by Wong et al. (Wong, Hryniowski, and Wang 2020; Wong, Wang, and Hryniowski 2020) and Hryniowski et al. (Hryniowski, Wong, and Wang 2021) introduced several human-interpretable approaches to trust quantification which allow for a model’s overall trustworthiness to be analyzed. These techniques are based on the notion of question-answer trust, in which a question  $x$  (i.e., ”is this solder joint defective?”) is answered by both a model  $M$  and an oracle  $O$ . The model’s answer and confidence in its answer are compared to the oracle’s answer to compute a question-answer trust score which reflects the trustworthiness of the model’s answer.

To evaluate SolderNet’s trustworthiness, we make use of two key trust metrics which summarize a model’s overall question-answer trust:

1. **Trust matrix**: a matrix of expected question-answer trusts for each possible model-oracle answer pair, and
2. **NetTrustScore**: a scalar metric summarizing the overall trustworthiness of a deep neural network across all possible predictions.

## Experimental Setup

**Dataset** To build and evaluate the proposed solder joint defect identification and explanation framework, we leverage a dataset comprising 2690 images of both through-hole and surface-mount joints acquired by Moog Inc. Images were acquired using a microscope camera and were labelled as either defective joints (1644 images) or non-defective joints (1046 images) by an experienced inspector. Notably, labelling was performed off-line without time constraints, thereby minimizing label noise in the resulting data. The dataset includes a variety of solder defects such as fractured solder joints, cold joints, burns, flux residue, poor wetting, and disturbed solder as illustrated in Figure 1. We used a 60%/20%/20% split stratified by class to divide this dataset into training, validation, and test sets, respectively.

**Preprocessing** Images are preprocessed by resizing them to  $256 \times 256$  pixels and mapping their original unsigned 8-bit integer pixel values to the range  $[0, 1]$  through division by 255. Additionally, the following data augmentations were used during training, each applied with a 50% probability:

random rotation in the range  $[0^\circ, 45^\circ]$ , random horizontal and vertical flipping, random translation of  $\pm 10\%$  in each direction, random brightness jitter of  $\pm 20\%$ , and random contrast jitter of  $\pm 20\%$ .

**Training** All models examined in this work were pre-trained on ImageNet-1k (Deng et al. 2009). Following (Kumar et al. 2022), each model’s fully-connected layers were trained for 100 epochs followed by full-model fine-tuning for 1000 epochs. Binary cross-entropy loss and an AdamW optimizer (Loshchilov and Hutter 2019) with  $(\beta_1, \beta_2) = (0.9, 0.999)$  and weight decay of  $1 \times 10^{-4}$  were used in all experiments. We used a batch size of 128 and a learning rate of  $1 \times 10^{-3}$  to train the fully-connected layers, followed by full-network fine-tuning with a batch size of 128, an initial learning rate of  $5 \times 10^{-4}$ , and cosine learning rate decay.

**Evaluation** To evaluate performance, we report accuracy, overkill rate (number of false-positives divided by number of samples), and escape rate (number of false-negatives divided by number of samples) on the the holdout test set. We chose these metrics due to the manufacturing context of this work; manufacturers care more about the absolute rates of false-positives and false-negatives rather than the proportional rates. Additionally, we report inference latency on an ARM Cortex-A72 processor, as well as NetTrustScore (Wong, Wang, and Hryniowski 2020) to evaluate the trustworthiness of each model. Lastly, visual explanations and trust quantification plots were qualitatively evaluated.

## Results

### Quantitative Results

Quantitative performance metrics for each of the tested network architectures are shown in Table 2. All tested architectures achieve high performance, with Attend-NeXt Large achieving the highest accuracy (91.1%) and lowest overkill rate (5.0%), MobileNetV2 achieving the lowest escape rate (3.7%), and Attend-NeXt Small achieving the lowest latency (0.275 s).

An intuitive interpretation of these metrics can be obtained by considering a scenario where 100 solder joints are to be inspected in a deployment setting. Considering the metrics of Attend-NeXt Large, we see that it would take

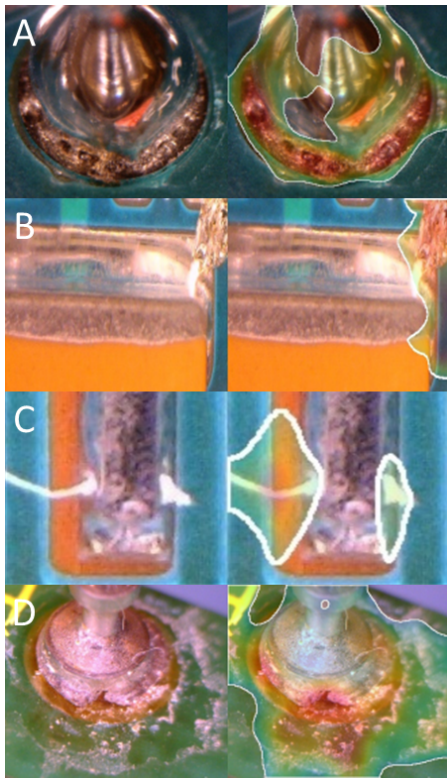


Figure 3: Images of solder joint defects (left) and corresponding visual explanations (right) from the dataset examined in this study: (A) poor wetting, (B) solder splash, (C) foreign object in solder, and (D) pad and board damage.

43 s to inspect the set of joints and about 91/100 joints would be correctly classified. Of the remaining 9 misclassified joints, about 5 would be misclassified as defective and about 4 would be misclassified as non-defective. However, it is important to note that this interpretation is only meaningful if the proportions of defective and non-defective joints in the test data are representative of the true proportions in deployment. In this study, defective solder joints are over-represented in the test data, and as such we might expect higher overkill rates and reduced escape rates in practice.

### Explainability Results

In this section we present and describe defect explanation and localization results obtained through the proposed explainability module. Figure 3 illustrates defective solder joints and the corresponding explanations of an AttendNeXt Large model’s predictions for these images. In each case, the XAI algorithm identified the critical factors that quantitatively drove the resulting decision (white outline) and the relative importance of aspects within each critical factor (semi-transparent regional heatmaps) indicating the critical factors used by the model to make its prediction. In all five cases, we observe that GSInquire identifies the key areas of interest driving the network’s decision-making process and further localizes specific features via the regional heatmaps. These cases are described in more detail below.

**Case A** This image illustrates poor wetting, where a globule of solder which has not adequately joined with the solder pad can be seen. Additionally, the solder pad itself appears lumpy and discoloured which indicates possible residue or contamination. Examining the visual explanation, we see that the model correctly focuses on the non-wetted regions of the joint when making its prediction.

**Case B** In this example, the soldering process has introduced a solder splash which can be seen in the top-right corner of the image. Examining the visual explanation, we see that the model correctly identified this solder splash as the defect and effectively ignored the solder joint (which is well-formed) when making its decision.

**Case C** In this image, a foreign object (a piece of fibre) was inadvertently embedded in the solder when the joint was created. This foreign object is captured in its entirety by visual explanation, with the fibre itself being highlighted as the most important aspect in the explanation.

**Case D** Extensive damage to the pad and board are shown in this example, where a large piece has chipped off of the solder pad and extensive damage to the board’s surface around the joint can be seen. While both of these aspects would constitute a defective joint, we see in the visual explanation that the model focuses on the damage to the solder pad while still including the surface damage.

### Second-order Explainability Results

SOXAI visualizations are produced by placing the quantitative explanations produced by GSInquire across the wealth of data at hand at their corresponding 2-dimensional embedding locations following t-SNE. The resulting image illustrates groups of similar quantitative explanations which can more easily be examined for semantic groupings and common trends and patterns.

Figure 4 illustrates a SOXAI visualization for an AttendNeXt Large model. As shown in the figure, SOXAI automatically groups explanations with similar characteristics, making it easier to find trends in the visual explanations. For example, in Figure 4 (A), we see that a homogeneous set of overhang defects has been tightly grouped. The relatively large size of this group indicates that this particular defect is well-recognized by the model but may be over-represented. In contrast, Figure 4 (B) shows a group of lifted leads which exhibit greater diversity but may be under-represented. In the neighbourhood of Figure 4 (C) we observe a large variety of through-hole defects, with (C) highlighting a group of wetting defects. The large size of the through-hole group paired with the intra-group variability indicates that through-hole joints and their defects are well-represented in the dataset and well-recognized by the model.

### Trust Analysis

The NetTrustScores for the models examined in this work are shown in the rightmost column of Table 2. These scores give an overall measure of how trustworthy each model’s predictions are. As shown, there is little disparity in trust

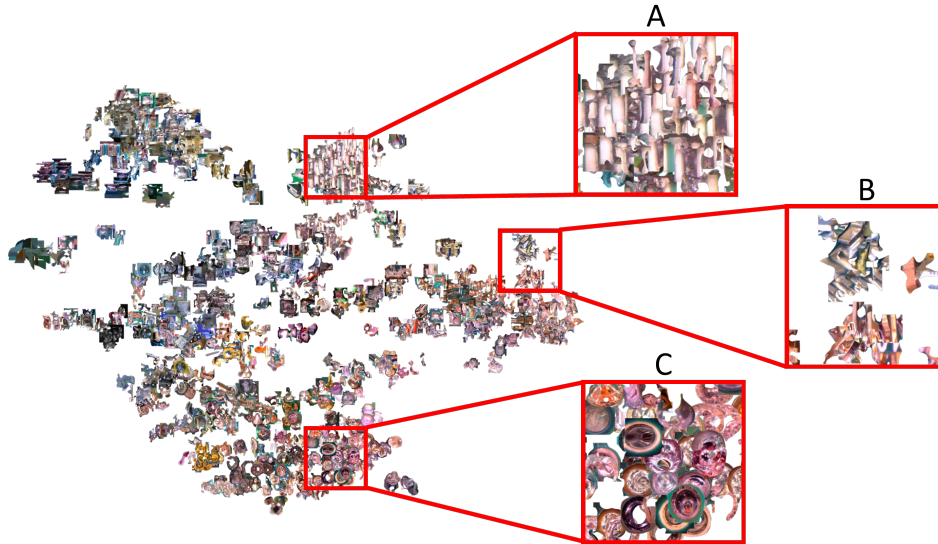


Figure 4: Second-order visual explainability illustrating various types of solder joint defects as viewed by the network: (A) side overhang, (B) lifted/unsoldered leads, and (C) wetting defects.

True Label	Non-defective	0.97663	0.022753
	Defective	0.078545	0.99382
		Non-defective	Defective
		<b>Predicted Label</b>	

Figure 5: Trust matrix of Attend-NeXt Large.

amongst the models examined in this work, with Attend-NeXt Large achieving the highest score of 0.904 and Attend-NeXt Small having the lowest score of 0.864.

To explore trust in more detail, Figure 5 shows the trust matrix for Attend-NeXt Large. Notably, we show this matrix as an illustrative example since the trust matrices for the other architectures have a similar pattern. To interpret the trust matrix, consider that each entry indicates the expected question-answer trust for the given ground-truth/prediction pair. As such, higher values are better in all cells. Examining Figure 5, we see that the diagonal entries (i.e., correct prediction scenarios) exhibit high trust. However, the off-diagonal entries (i.e., incorrect prediction scenarios) exhibit extremely low trust, indicating that the model is overconfident when it makes incorrect predictions. This is problematic in deployment scenarios, as it makes it more difficult to identify uncertain model predictions in order to flag them for manual review. To alleviate this problem, techniques such as label smoothing and mixup regularization (Zhang et al. 2018) could be used to soften the image labels during training and encourage intermediate confidence scores.

## Discussion

In this work, we described a design for a trustworthy, explainable solder joint inspection system for use in electronics manufacturing. While this system has yet to be fully implemented, we present important progress towards trustworthy, explainable solder joint inspection which forms the core of the proposed system. Moreover, we discuss practical considerations for building and evaluating such a system and show how trust quantification, quantitative explainability, and second-order explainability can be leveraged to analyze the trustworthiness of the system and identify biases or gaps in the data and model development process as well as provide insights during inspection.

The image data analyzed in this study varies considerably in terms of camera viewpoint, magnification, and resolution. In practice, a standardized imaging system would be required, however implementing an adequate imaging system is technically challenging due to the fact that different types of solder joints may need to be imaged at different angles, exposures, or resolutions in order to capture the majority of possible solder defects. In this study, we have assumed that such a system *can* be designed, but the specific details of how to do so are left to future work.

When deploying a system such as the one described in this study, it is important to monitor and validate the system's performance in the field in order to identify and correct any issues that arise. No amount of offline testing can fully simulate a system's real-world performance, and so collecting prediction and performance data in the field is critical to evaluation. Additionally, false predictions observed in the field can be collected, curated, and used to fine-tune the system in order to reduce overkill and escape rates. Such continuous monitoring also helps to identify and mitigate drift in the system (for example, due to drift in camera calibration or other imaging parameters).

## Acknowledgments

We would like to thank Saejith Nair for measuring the inference latencies of the architectures examined in this work.

## References

- Bhatt, P. M.; Malhan, R. K.; Rajendran, P.; Shah, B. C.; Thakar, S.; Yoon, Y. J.; and Gupta, S. K. 2021. Image-Based Surface Defect Detection Using Deep Learning: A Review. *Journal of Computing and Information Science in Engineering*, 21(4).
- Deng, J.; Dong, W.; Socher, R.; Li, L.; Kai Li; and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 248–255.
- Erion, G.; Janizek, J. D.; Sturmfels, P.; Lundberg, S. M.; and Lee, S.-I. 2021. Improving performance of deep learning models with axiomatic attribution priors and expected gradients. *Nature Machine Intelligence*, 3: 620–631.
- Howard, A.; Sandler, M.; Chen, B.; Wang, W.; Chen, L.-C.; Tan, M.; Chu, G.; Vasudevan, V.; Zhu, Y.; Pang, R.; Adam, H.; and Le, Q. 2019. Searching for MobileNetV3. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 1314–1324.
- Hryniowski, A.; Wong, A.; and Wang, X. 2021. Where Does Trust Break Down? A Quantitative Trust Analysis of Deep Neural Networks via Trust Matrix and Conditional Trust Densities. *Journal of Computational Vision and Imaging Systems*, 6: 1–5.
- Kim, J.; Ko, J.; Choi, H.; and Kim, H. 2021. Printed Circuit Board Defect Detection Using Deep Learning via A Skip-Connected Convolutional Autoencoder. *Sensors (Basel)*, 21(15).
- Klamklay, J.; and Bishu, R. R. 1998. Visual Inspection of Circuit Boards: Effect of Gender, Age, Defect type, and Defect Proportion. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 42(16): 1161–1164.
- Kumar, A.; Raghunathan, A.; Jones, R. M.; Ma, T.; and Liang, P. 2022. Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution. In *International Conference on Learning Representations*.
- Lin, Z. Q.; Shafiee, M. J.; Bochkarev, S.; St. Jules, M.; Wang, X. Y.; and Wong, A. 2019. Do Explanations Reflect Decisions? A Machine-centric Strategy to Quantify the Performance of Explainability Algorithms. *arXiv preprint. arXiv:1910.07387*.
- Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; and Xie, S. 2022. A ConvNet for the 2020s. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Loshchilov, I.; and Hutter, F. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.
- Lundberg, S.; and Lee, S.-I. 2017. A Unified Approach to Interpreting Model Predictions. In *30th International Conference on Neural Information Processing Systems (NIPS 2017)*, 768–4777.
- Ma, N.; Zhang, X.; Zheng, H.-T.; and Sun, J. 2018. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. In Ferrari, V.; Hebert, M.; Sminchisescu, C.; and Weiss, Y., eds., *15th European Conference on Computer Vision (ECCV)*, 122–138.
- Ribeiro, M.; Singh, S.; and Guestrin, C. 2016. “Why Should I Trust You?”: Explaining the Predictions of Any Classifier. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, 97–101. San Diego, California: Association for Computational Linguistics.
- Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; and Chen, L.-C. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4510–4520.
- See, J. E.; Drury, C. G.; Speed, A.; Williams, A.; and Kalandi, N. 2017. The Role of Visual Inspection in the 21st Century. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 61(1): 262–266.
- Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, 618–626.
- van der Maaten, L.; and Hinton, G. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86): 2579–2605.
- Westphal, E.; and Seitz, H. 2021. A machine learning method for defect detection and visualization in selective laser sintering based on convolutional neural networks. *Additive Manufacturing*, 41: 101965.
- Wong, A.; Hryniowski, A.; and Wang, X. Y. 2020. Insights into Fairness through Trust: Multi-scale Trust Quantification for Financial Deep Learning. *arXiv preprint. arXiv:2011.01961*.
- Wong, A.; Shafiee, M. J.; Abbasi, S.; Nair, S.; and Famouri, M. 2022. Faster Attention Is What You Need: A Fast Self-Attention Neural Network Backbone Architecture for the Edge via Double-Condensing Attention Condensers. In *All things Attention: Bridging Different Perspectives on Attention (NeurIPS 2022 Workshop)*.
- Wong, A.; Wang, X. Y.; and Hryniowski, A. 2020. How Much Can We Really Trust You? Towards Simple, Interpretable Trust Quantification Metrics for Deep Neural Networks. *arXiv preprint. arXiv:2009.05835*.
- Yang, J.; Li, S.; Wang, Z.; Dong, H.; Wang, J.; and Tang, S. 2020. Using Deep Learning to Detect Defects in Manufacturing: A Comprehensive Survey and Current Challenges. *Materials*, 13(24).
- Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2018. mixup: Beyond Empirical Risk Minimization. *International Conference on Learning Representations*.
- Zhang, Q.; Zhang, M.; Gamanayake, C.; Yuen, C.; Geng, Z.; Jayasekara, H.; wei Woo, C.; Low, J.; Liu, X.; and Guan, Y. L. 2022. Deep learning based solder joint defect detection on industrial printed circuit board X-ray images. *Complex & Intelligent Systems*, 8: 1525–1537.