International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 11 | Nov - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Sentinel-X: A Category-Wise Audit of Grad-CAM Defect Localization on the VisA Benchmark

Niranjan P[1]

[1]Department of Artificial Intelligence and Data Science
Sri Krishna College of Technology, Coimbatore, Tamil Nadu, India
Email: niranjan0855@gmail.com

Abstract—An inspection system is far more useful when the operator can see why an image was flagged. We study Sentinel-X, a ResNet18 classifier for normal versus anomalous products, and ask how well its Grad-CAM explanations agree with real defect annotations on the VisA benchmark. As a classifier it performs well: on 1,629 test images it reaches an AUROC of 0.981 and an accuracy of 95.9%. The heatmaps are a different matter. On 180 anomalous test images (15 per category, all with readable, non-empty masks) the pooled pixel AUROC is 0.756 but the average precision is only 0.030, and at a fixed threshold of 0.5 the mean per-image IoU is 0.020 and the mean Dice coefficient 0.035. A threshold sweep peaks at 0.8 (IoU 0.029, Dice 0.050); because that value was picked on test data, we report it as exploratory. Overlap differs about 28-fold between categories and follows defect size: the thresholded map covers on average 23 times more area than the defect itself, and the categories with the largest defects overlap most. We conclude that Grad-CAM from this classifier shows roughly where the network looked, not where the defect is, and we list the protocol changes needed before such maps could support inspection decisions.

Index Terms—industrial inspection, visual anomaly detection, explainable artificial intelligence, Grad-CAM, ResNet18, VisA, defect localization

I. INTRODUCTION

On a production line, a defect can be a hairline crack, a missing component, or a small discolouration, and it often sits in a visually busy scene. A classifier that outputs only "normal" or "defective" gives the reviewer nothing to check. When a false alarm has to be explained, or a missed defect investigated afterwards, a map showing which region drove the decision is much more helpful than the label alone.

Post-hoc explanation methods are attractive here because they need no extra annotation and no change to the trained network. Grad-CAM is the most widely used of them [3]. It is also easy to over-read: a heatmap that looks sensible on a few hand-picked images says little about whether it overlaps the true defect across a whole benchmark. The VisA dataset supplies pixel-level masks for twelve object subsets [1], which lets us test this question directly instead of relying on visual impressions.

We therefore treat Sentinel-X, a binary ResNet18 classifier [2], as the object of an audit rather than as a new segmentation architecture. We do not retrain or modify the checkpoint. The classifier itself is accurate, which sharpens the question: a model can be right about an image while its explanation points elsewhere. We ask a narrower question: if Grad-CAM were used to point at a defect, how often and how well would it hit it? The contributions are:

- an audited evaluation set of 180 anomalous test images, 15 per category, each with a matching non-empty ground-truth mask;
- a category-wise analysis of Grad-CAM overlap (IoU and Dice) together with pooled pixel AUROC and average precision, including how the picture changes between two thresholds;
- a threshold-sensitivity study that separates the fixed operating point ($\tau = 0.5$) from an exploratory one chosen on test data, with the selection bias stated openly;
- an analysis that relates overlap to defect area and to the extent of the thresholded map;
- a list of concrete protocol changes (validation-based thresholding, inclusion of normal images, stronger baselines) that follow from the weak overlap we observe.

The results are mostly negative, and we report them as such. A clear measurement of where a common explanation method fails is useful to anyone considering it for inspection tasks.

II. RELATED WORK

A. Industrial anomaly benchmarks and methods

MVTec AD brought pixel-accurate defect annotations to unsupervised anomaly detection, with 15 object and texture categories [4]. The journal version documents the dataset and its evaluation metrics, including per-region overlap, in more detail [16], and MVTec LOCO AD later added logical anomalies such as wrongly placed or missing components [17]. VisA followed with twelve subsets that include printed circuit boards and scenes containing several instances of the same object [1]. More recently, MVTec AD 2 added eight new scenarios with harder conditions such as lighting variation [5]. Because defects differ so much between subsets, we report every number per category in addition to the aggregate.

Most strong methods on these benchmarks do not use a classifier at all. PaDiM models the distribution of pretrained patch features at each image position [6]. PatchCore stores a reduced memory bank of locally aware patch features and scores test patches by their distance to it [7]. Reverse distillation reconstructs teacher features from a compact one-class embedding [8], and EfficientAD targets very low inference latency [9]. Few-shot approaches use language-vision or self-supervised backbones: WinCLIP compares windowed features of CLIP [20] against text prompts and a few normal examples [10]; AnomalyCLIP learns object-agnostic text prompts [21]; AnomalyGPT couples a vision-language model with a language model that answers questions about defects [22]; and AnomalyDINO matches patch features

of DINOv2 [23] to a small reference set [11]. These methods produce an anomaly map by construction. Earlier work trains student networks to regress the features of a pretrained teacher and reads their disagreement as the anomaly signal [18], and a recent survey organises the field by how the normal distribution is modelled [19]. Our setting differs: the map is a by-product of a classifier trained for a different output.

Table I: How the present study relates to the literature.

Work	Role	Spatial output	Link to this study
VisA [1]	Dataset	Pixel masks	Source of images and masks
PatchCore [7]	Anomaly detector	Patch-level anomaly map	Type of baseline we do not yet include
FCDD [12]	Explanation by design	Heatmap from architecture	Alternative to post-hoc maps
Grad-CAM [3]	Post-hoc explanation	Coarse heatmap	Method under audit
This study	Audit of classifier plus Grad-CAM	IoU, Dice, AUROC, AP	Category-wise agreement with masks

B. Explanation in anomaly detection

One line of work builds the explanation into the model. Fully convolutional data description, for example, yields a heatmap directly from the architecture of a one-class detector [12]. A second line, which we follow, leaves the model untouched and explains it afterwards. Model-agnostic tools such as LIME and SHAP attribute a prediction to input regions through local surrogate models or Shapley values [24], [25], and Integrated Gradients accumulates gradients along a path from a baseline input [26]. We use the cheaper class of gradient-based activation maps.

C. Gradient-based explanation and its evaluation

Using the gradient of a class score with respect to the input as a saliency map goes back to [27], and class activation mapping showed that feature maps can localise class evidence in networks that end with global average pooling [28]. Grad-CAM removes that architectural restriction: it weights the feature maps of a convolutional layer by the pooled gradient of a target score and keeps the positive part [3]. Grad-CAM++ refines the weighting to handle several instances of an object in one image [13], and Score-CAM replaces gradients with score changes obtained from masked inputs [14]. A well-known check on saliency methods is whether they depend on the trained weights and on the data [15]; passing that check shows that a map is sensitive to the model, not that it overlaps a human-annotated region. Overlap with ground truth is the question we address.

III. METHODOLOGY

A. Pipeline overview

Figure 1 shows the evaluation workflow. An image passes through the ResNet18 classifier, Grad-CAM produces a map from the last convolutional block, the map is resized and normalised, and the result is compared with the ground-truth mask.



Fig. 1. Sentinel-X evaluation workflow: ResNet18 classifier, Grad-CAM map, normalisation and thresholding, and comparison with the VisA ground-truth mask.

B. Dataset and evaluation split

We use the twelve VisA categories: candle, capsules, cashew, chewinggum, fryum, macaroni1, macaroni2, pcb1, pcb2, pcb3, pcb4 and pipe_fryum. Our test loader contains 1,629 images, of which 1,449 are normal and 180 are anomalous. An audit of the anomalous images found exactly 15 per category, each with a readable, non-empty mask. Only these 180 images enter the localisation analysis (Table II). Normal images are left out because they have no defect mask, so overlap with a defect is undefined for them. The exact construction of the 1,629-image loader must be documented from the experiment script or split manifest before submission; it is not the official VisA split, so numbers should not be compared with published pixel-level leaderboards.

Table II: Evaluation data audit.

Item	Count
Test images in loader	1,629
Normal test images	1,449
Anomalous test images	180
Anomalous images with valid masks	180
Categories evaluated	12
Anomalous images per category	15

C. Classifier and Grad-CAM

Sentinel-X is an existing ResNet18 checkpoint with a single logit for the anomalous class. The checkpoint was trained with a binary cross-entropy loss for 24 epochs (Fig. 5). The training/validation split, optimiser, learning rate and augmentation settings must be recovered from the experiment configuration and reported before submission. For an input image we take the last convolutional block as the target layer. With A^k the k-th feature map and y the anomaly logit, the channel weight is

$$\alpha_k = \frac{1}{Z}\sum_i \sum_j \frac{\partial y}{\partial A^k_{ij}} \quad (1)$$

where Z is the number of spatial positions, and the raw map is

$$L_{\text{GradCAM}} = \text{ReLU}(\textstyle\sum_k \alpha_k A^k) \quad (2)$$

The standard ResNet18 reduces resolution by a factor of 32 in its last stage; with our 224 × 224 inputs the raw map is therefore 7 × 7, and each cell spans a 32 × 32 pixel block. This map is upsampled to the image resolution and min–max normalised separately for each image. The normalisation serves visualisation and thresholding only; the values are relative activations, not calibrated probabilities.

D. Localisation metrics

Let P be the heatmap thresholded at τ and G the ground-truth mask. We compute

$$\text{IoU} = \frac{|P \cap G|}{|P \cup G|} \quad (3)$$

$$\text{Dice} = \frac{2|P \cap G|}{|P| + |G|} \quad (4)$$

for every image and average over the 180 images. These are the Jaccard index [30] and the Dice coefficient [29]. We do not compute the per-region-overlap AU-PRO used by the MVTec benchmarks [16]. Pixel AUROC and average precision (AP) are computed on pixels pooled across the evaluated images. Defect pixels are a small fraction of each image, and the AP of a random ranking equals that fraction, so AP must be read against the positive-pixel rate, which is 0.76% on the 224 × 224 evaluation grid of our 180 images.

E. Threshold analysis

The primary result uses the fixed threshold $\tau = 0.5$. We also evaluated τ from 0.1 to 0.9 in steps of 0.1 to see how sensitive overlap is. The best value on this sweep is exploratory, since choosing it on the test set biases the estimate upward. A deployable threshold should be selected on a separate validation split and then applied once to untouched test data.

IV. EXPERIMENTAL RESULTS

A. Classifier performance

Before judging the heatmaps we characterise the classifier on the full test loader of 1,629 images. At the 0.5 decision threshold it classifies 1,563 images correctly (accuracy 95.95%), with an AUROC of 0.9806 and an average precision of 0.8980 (Table III). Of the 180 anomalous images, 144 are detected (recall 0.800) and 36 are missed; of the 1,449 normal images, 30 are flagged wrongly (specificity 0.979). Precision on the anomalous class is 0.828 and F1 is 0.814. Only about 11% of the test images are anomalous, so AP and F1 are more informative than accuracy. Category-wise accuracy (Fig. 2) ranges from 89.6% for capsules to 98.9% for fryum. Validation AUROC and AP rise from about 0.89 and 0.57 in the first epoch to roughly 0.98 and 0.89 and are stable after about ten epochs, apart from a dip near epoch 15 (Fig. 5b). The training loss falls to about 0.05 while the validation loss levels off near 0.15 after roughly ten epochs (Fig. 5a), a modest generalisation gap that does not show up in the validation AUROC or AP. Figure 4 gives the ROC curve, the precision–recall curve and the confusion matrix. Precision stays high up to a recall of roughly 0.6 and falls off steeply at higher recall, which is where the 36 missed defects sit. Per-category AUROC (Fig. 3) is high for all twelve categories, roughly 0.94 or higher, with capsules the lowest, in line with its lowest accuracy.

Table III: Image-level classification performance on the 1,629-image test loader (threshold 0.5). Precision, recall, F1 and specificity are computed from the confusion matrix.

Metric	Value
Accuracy	0.9595
AUROC	0.9806
Average precision	0.8980
Precision (anomalous)	0.8276
Recall (anomalous)	0.8000
F1 (anomalous)	0.8136
Specificity (normal)	0.9793
TP / FP / FN / TN	144 / 30 / 36 / 1419

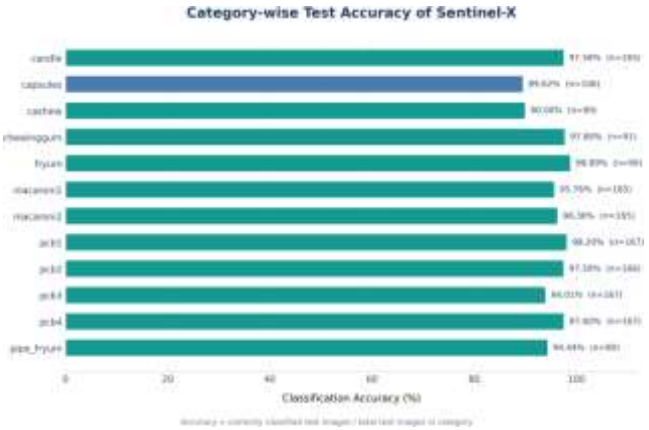


Fig. 2. Category-wise test accuracy of the classifier (number of test images per category in parentheses).

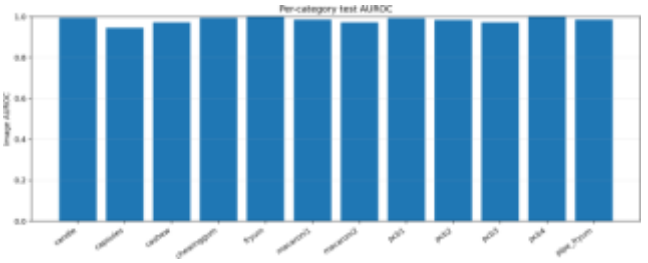


Fig. 3. Per-category test AUROC of the classifier.

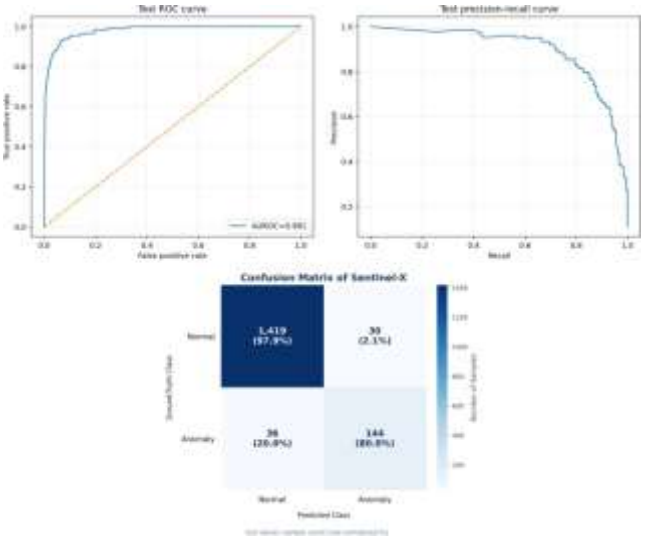


Fig. 4. Test-set performance of the classifier on 1,629 images: ROC curve (AUROC 0.981), precision–recall curve (AP 0.898) and confusion matrix at threshold 0.5.

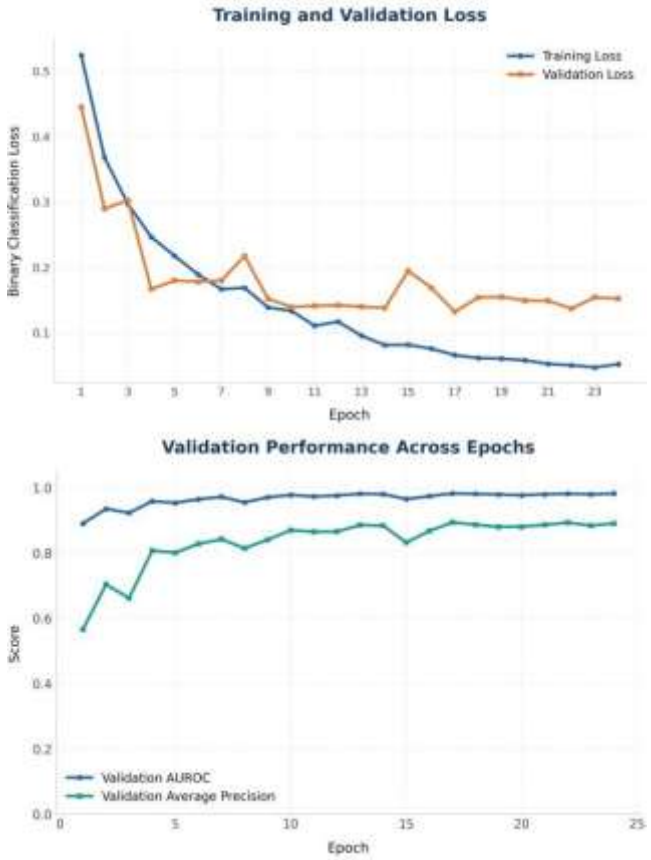


Fig. 5. Training history over 24 epochs. (a) Training and validation binary cross-entropy loss. (b) Validation AUROC and average precision.

B. Overall localisation performance

Table IV gives the main result. A pooled pixel AUROC of 0.756 says that, across the pooled pixels, defect pixels tend to receive higher heatmap values than background pixels. The AP of 0.030 and the mean IoU of 0.020 tell a different story: the highest-ranked pixels are mostly not defect pixels, and the thresholded region shares very little area with the annotation. In short, the maps contain a weak ranking signal but a poor spatial match. For scale, the random-ranking baseline of AP equals the positive-pixel rate of 0.0076, so 0.030 is about four times chance, but still low in absolute terms.

Table IV: Overall Grad-CAM localisation results on 180 anomalous images.

Metric	Value
Anomalous images evaluated	180
Pixel AUROC (pooled)	0.756285
Pixel average precision (pooled)	0.030006
Mean per-image IoU ($\tau = 0.5$)	0.019949
Mean per-image Dice ($\tau = 0.5$)	0.034577

C. Category-wise performance

Table V and Fig. 6 break the result down by category. At $\tau = 0.5$ the best mean IoU is on pcb4 (0.0598), followed by fryum (0.0506), pipe_fryum (0.0479) and chewinggum (0.0449). The weakest are macaroni2 (0.0021) and macaroni1 (0.0023), roughly 28 times lower than pcb4. Even the best category covers only about six percent of the union of prediction and annotation. The three other PCB subsets sit near 0.006, an order of magnitude below pcb4, so object family alone does not explain the spread. These are descriptive findings for this checkpoint and split, not a general ranking of category difficulty.

Table V: Category-wise Grad-CAM localisation metrics (15 anomalous images per category).

Category	Images	Mean IoU	Mean Dice	IoU at $\tau = 0.8$
candle	15	0.004603	0.009076	0.007510
capsules	15	0.003839	0.007445	0.000020
cashew	15	0.004364	0.008622	0.004520
chewinggum	15	0.044888	0.082875	0.124590
fryum	15	0.050588	0.076412	0.043790
macaroni1	15	0.002287	0.004551	0.003190
macaroni2	15	0.002139	0.004259	0.005430
pcb1	15	0.006301	0.012371	0.012890
pcb2	15	0.006101	0.011946	0.008390
pcb3	15	0.006532	0.012890	0.012020
pcb4	15	0.059802	0.110101	0.090610
pipe_fryum	15	0.047946	0.074375	0.034660

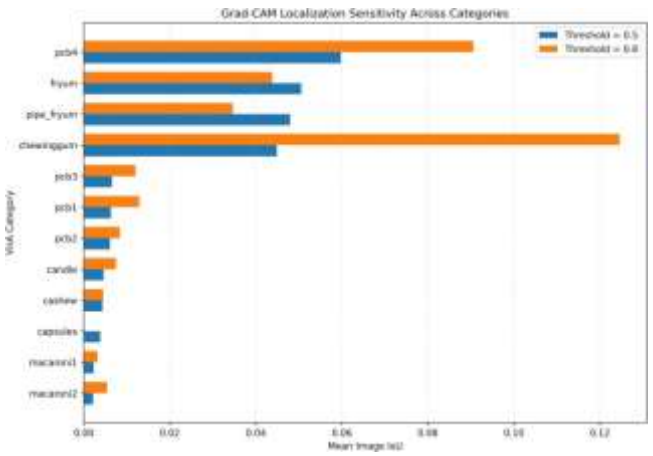


Fig. 6. Category-wise mean IoU at the fixed threshold 0.5 and at the exploratory threshold 0.8. The latter was chosen after inspecting test results and is not an unbiased operating point.

D. Threshold sensitivity

Table VI and Fig. 7 show that mean IoU grows steadily from 0.0124 at $\tau = 0.1$ to 0.0290 at $\tau = 0.8$, a rise of about 2.3 times, and then falls to 0.0261 at 0.9. Dice follows the same shape and peaks at 0.0499. Tightening the threshold helps because a loose threshold lets the coarse map spill far beyond a small defect. Even the best value remains small in absolute terms, and since it was chosen on the test set, we keep $\tau = 0.5$ as the headline result.

Table VI: Threshold sensitivity on the test set.

Threshold	Mean IoU	Mean Dice
0.1	0.01242	0.02137
0.2	0.01464	0.02496
0.3	0.01644	0.02810
0.4	0.01815	0.03124
0.5	0.01995	0.03458
0.6	0.02259	0.03932
0.7	0.02635	0.04573
0.8	0.02897	0.04985
0.9	0.02605	0.04418

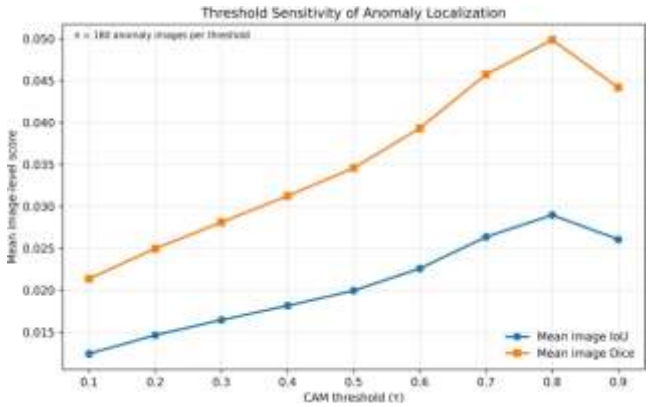


Fig. 7. Threshold sensitivity of mean per-image IoU and Dice.

The effect is not uniform across categories (Table V). Raising the threshold to 0.8 roughly triples IoU for chewinggum (0.045 to 0.125) and raises it for pcb4 (0.060 to 0.091). It lowers IoU for capsules (0.0038 to nearly zero), fryum (0.0506 to 0.0438) and pipe_fryum (0.0479 to 0.0347). A single global threshold therefore trades one category against another, which is a further reason to select any deployed threshold on validation data and possibly per category.

Figure 8 shows the mean IoU for every category and threshold. The best threshold differs widely: 0.9 for chewinggum and macaroni2, 0.8 for candle and pcb1–pcb3, 0.7 for pcb4, cashew and macaroni1, 0.6 for fryum, 0.3 for pipe_fryum and 0.1 for capsules. Choosing the best threshold separately for each category would raise the mean IoU from 0.0290 (global $\tau = 0.8$) to 0.0357. Since that choice is made on the test set, 0.0357 is an optimistic upper bound for this method and not an attainable estimate.

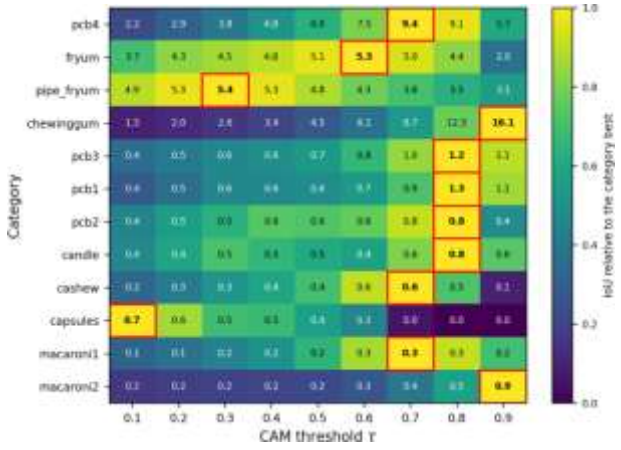


Fig. 8. Mean IoU (×100) per category and threshold. Colour shows IoU relative to the best threshold of the same category, and the red box marks that best threshold. Choosing it uses the test set, so it is exploratory.

E. Defect size and heatmap extent

The annotated defects are very small. Measured on the 224 × 224 grid, the median defect covers 98 pixels (0.2% of the image), three quarters of the 180 defects cover fewer than 200 pixels, and the mean of 379 pixels is pulled up by a few large defects. The thresholded Grad-CAM region at $\tau = 0.5$ covers 17.3% of the image on average, about 23 times the mean defect area (0.76%). Against such a small target the overlap is capped by the size of the prediction, and in 64 of the 180 images (36%) the IoU is exactly zero. Per image, IoU rises with defect area (Spearman $\rho = 0.55$, Fig. 9), and across the twelve categories mean defect area and mean IoU are strongly rank-correlated ($\rho = 0.85$). The two macaroni subsets have the smallest defects (about 30 pixels on average) and the lowest IoU; pipe_fryum, fryum, pcb4 and chewinggum have the four largest mean defect areas and the highest IoU. The relation is not exact: fryum and pipe_fryum have a median defect of about 90 pixels and owe their mean to a few large defects. Size therefore explains a large part of the category spread, although this correlational evidence does not settle causality.

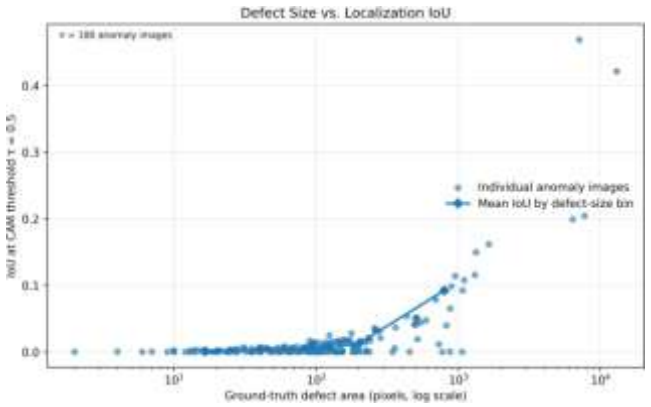
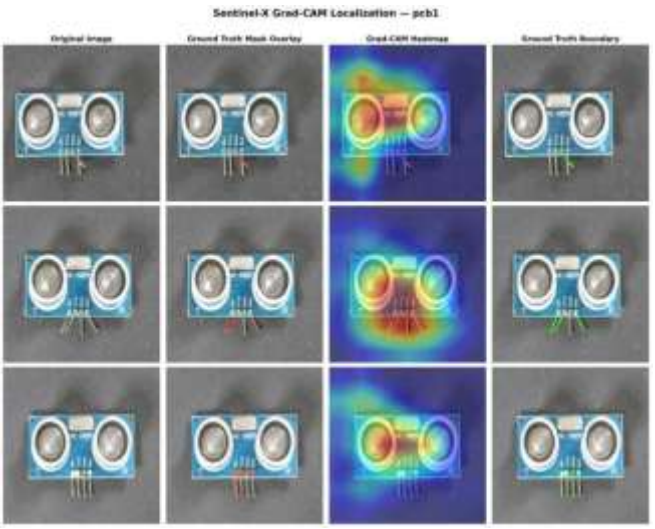


Fig. 9. Per-image IoU at $\tau = 0.5$ against ground-truth defect area for the 180 evaluated images. Many points lie at zero.

F. Further per-category outputs

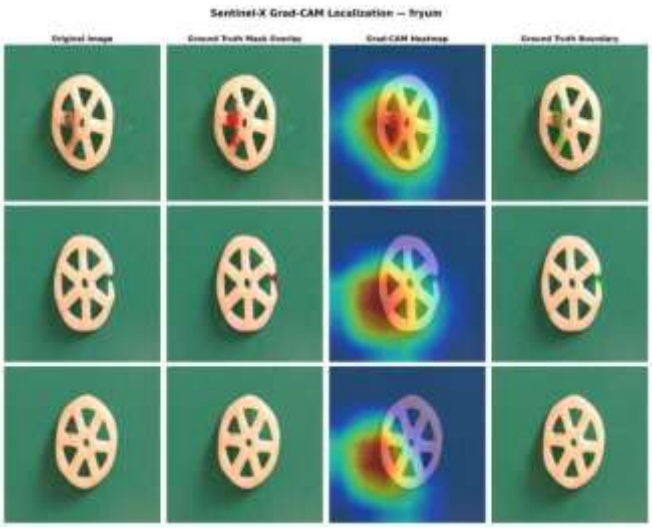
Figures 10–12 show three further anomalous test images for each of five more categories (Outputs 1 to 5); the text under each box states what the heatmaps do. Only for chewinggum, whose defects are large, does the strongest activation consistently fall on or next to the defect, and even there the map is much larger than the defect. For the small defects of capsules, candle and pcb1, the maxima often lie at the centre of the image, between the objects, or at the image border, which looks more like a centre-and-border tendency than like evidence taken from the defect. A centre-prior baseline, which is not part of the present study, would be the direct test of this impression.

Output 1: pcb1



Mean IoU at $\tau = 0.5$: 0.0063. The defects are bent pins and solder blobs below the board. Only in the middle example does strong activation lie over the bent pins; in the other two the map peaks on the left transducer and on the background.

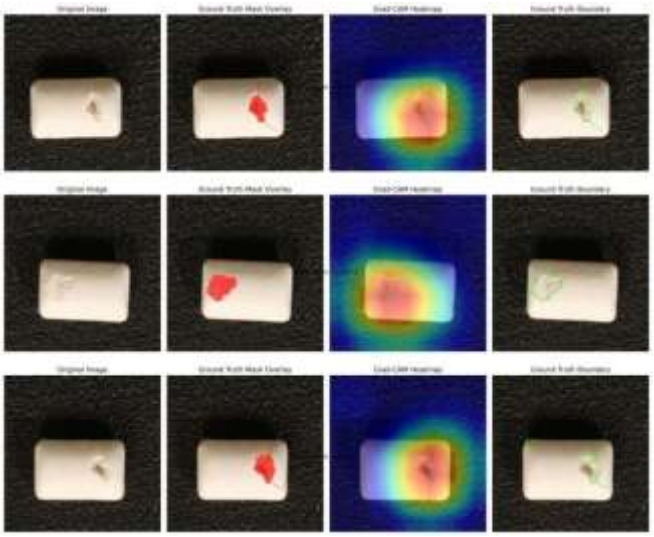
Output 2: fryum



Mean IoU at τ = 0.5: 0.0506. In the first example the map covers the dark contamination near the hub. In the other two the maximum lies on the background to the left of the object, away from the defect.

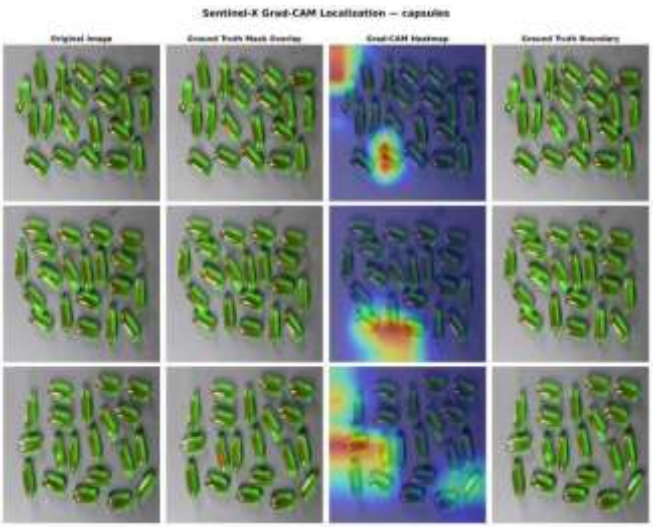
Fig. 10. Outputs 1 and 2. Columns in each box: input, ground-truth overlay, Grad-CAM heatmap, ground-truth boundary.

Output 3: chewinggum



Mean IoU at τ = 0.5: 0.0449. The defects are large chips. The maximum activation lies on or next to the defect in all three examples, but the map extends over much of the gum and into the background beyond its outline.

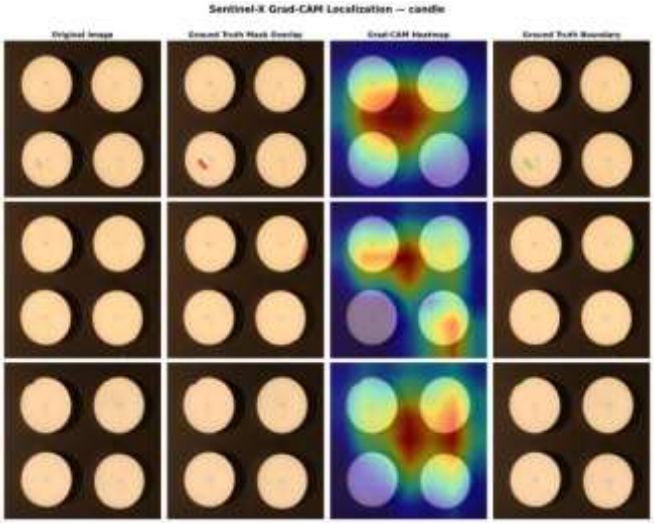
Output 4: capsules



Mean IoU at τ = 0.5: 0.0038. The defect is a small spot on one of many capsules. In all three examples the map peaks on other capsules or at the image border. This category also has the lowest classification accuracy (89.6%).

Fig. 11. Outputs 3 and 4. Panels as in Fig. 10.

Output 5: candle



Mean IoU at τ = 0.5: 0.0046. Each image shows four candles; the defects are a dent or thin chips on the edge of one candle. In all three examples the strongest activation sits in the gap between the candles, not on the defective candle.

Fig. 12. Output 5. Panels as in Fig. 10.

V. DISCUSSION

The experiment separates two questions that are easy to conflate: whether a classifier can tell defective images from good ones, and whether a post-hoc heatmap traces the defect itself. The first is answered well (AUROC 0.981, accuracy 95.9%). The second is not. Pixel AUROC is moderately above chance and AP is about four times its random baseline, yet the overlap is small, so the ranking signal does not become a usable mask.

The two questions come apart sharply within some categories. Macaroni1 has an image-level accuracy of 95.8% (Fig. 2) and still the second-lowest overlap (0.0023), while capsules has the lowest accuracy (89.6%) and loses nearly all overlap when the threshold is raised to 0.8. A classifier trained only for the image-level decision has no incentive to concentrate on the defect rather than on any other cue that

predicts the label, such as object arrangement or background. The low overlap scores are consistent with this explanation, although they do not establish the cause.

Three further factors are visible in our data. The map is coarse: each cell of the 7 × 7 map covers a 32 × 32 pixel block, while the median defect covers 98 pixels. The thresholded region is on average 23 times larger than the defect (Section IV-E). And per-image min–max normalisation forces every map to use the full range from 0 to 1, so even an image with no clear focus receives high values somewhere. Category differences follow defect size closely ($\rho = 0.85$ across categories) but probably also reflect object arrangement and defect appearance; separating these requires controlled experiments, for example grouping images by defect area and repeating the study over several seeds and checkpoints.

A. Limitations

First, the localisation analysis uses only the 180 anomalous images, so it is not a full benchmark comparison with dedicated segmentation methods, and our split is not the official VisA split. Second, with 15 images per category, per-category means are noisy and we do not report confidence intervals. Third, normalisation is applied per image, so thresholds describe relative activation. Fourth, the 0.8 threshold was chosen from test results; the 0.5 figure remains the primary result until a validation-selected threshold is evaluated on untouched data. Fifth, we examine one checkpoint and one explanation method.

VI. CONCLUSION AND FUTURE WORK

Sentinel-X, a ResNet18 classifier, separates normal from anomalous VisA images well (AUROC 0.981, accuracy 95.9% on 1,629 test images), but its Grad-CAM maps localise defects poorly. On 180 anomalous test images with valid masks, pooled pixel AUROC is 0.756 while mean IoU and Dice at $\tau = 0.5$ are only 0.020 and 0.035; overlap differs strongly between categories and follows defect size. Raising the threshold to 0.8 helps on average but hurts several categories, and it was selected on test data. The maps are a coarse guide to the model's focus and should not be presented as defect segmentations.

Future work should (i) choose thresholds on a validation split, per category if possible, and evaluate once on held-out data; (ii) include normal images in a fully specified protocol; (iii) compare against dedicated anomaly-localisation methods such as PatchCore and few-shot DINOv2-based detectors under identical splits and metrics; (iv) report confidence intervals over repeated runs; and (v) analyse overlap by defect size. These steps will show whether the patterns reported here are stable and whether an anomaly-map-based approach is needed for practical inspection.

ACKNOWLEDGMENT

The authors thank the creators of the VisA dataset for releasing the images and annotations. Add funding or institutional acknowledgment only if applicable. If AI tools were used for language editing, state this as required by the target venue's policy.

REFERENCES

[1] Y. Zou, J. Jeong, L. Pemula, D. Zhang, and O. Dabeer, "SPot-the-Difference self-supervised pre-training for anomaly detection and segmentation," in Proc. European Conf. Computer Vision (ECCV), 2022, pp. 392–408, doi: 10.1007/978-3-031-20056-4_23.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[3] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2017, pp. 618–626, doi: 10.1109/ICCV.2017.74.

[4] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, "MVTec AD – A comprehensive real-world dataset for unsupervised anomaly detection," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2019, pp. 9584–9592, doi: 10.1109/CVPR.2019.00982.

[5] L. Heckler-Kram et al., "The MVTec AD 2 dataset: Advanced scenarios for unsupervised anomaly detection," Int. J. Computer Vision, 2026, doi: 10.1007/s11263-026-02743-0.

[6] T. Defard, A. Setkov, A. Loesch, and R. Audigier, "PaDiM: A patch distribution modeling framework for anomaly detection and localization," in Pattern Recognition. ICPR International Workshops and Challenges, 2021, doi: 10.1007/978-3-030-68799-1_35.

[7] K. Roth, L. Pemula, J. Zepeda, B. Schölkopf, T. Brox, and P. Gehler, "Towards total recall in industrial anomaly detection," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2022, doi: 10.1109/CVPR52688.2022.01392.

[8] H. Deng and X. Li, "Anomaly detection via reverse distillation from one-class embedding," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2022, doi: 10.1109/CVPR52688.2022.00951.

[9] K. Batzner, L. Heckler, and R. König, "EfficientAD: Accurate visual anomaly detection at millisecond-level latencies," in Proc. IEEE/CVF Winter Conf. Applications of Computer Vision (WACV), 2024, doi: 10.1109/WACV57701.2024.00555.

[10] J. Jeong, Y. Zou, T. Kim, D. Zhang, A. Ravichandran, and O. Dabeer, "WinCLIP: Zero-/few-shot anomaly classification and segmentation," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2023, doi: 10.1109/CVPR52729.2023.01878.

[11] S. Damm, M. Laszkiewicz, J. Lederer, and A. Fischer, "AnomalyDINO: Boosting patch-based few-shot anomaly detection with DINOv2," arXiv:2405.14529, 2024, doi: 10.48550/arXiv.2405.14529.

[12] P. Liznerski, L. Ruff, R. A. Vandermeulen, B. J. Franks, M. Kloft, and K.-R. Müller, "Explainable deep one-class classification," in Proc. Int. Conf. Learning Representations (ICLR), 2021, arXiv:2007.01760.

[13] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in Proc. IEEE Winter Conf. Applications of Computer Vision (WACV), 2018, doi: 10.1109/WACV.2018.00097.

[14] H. Wang et al., "Score-CAM: Score-weighted visual explanations for convolutional neural networks," in Proc. IEEE/CVF CVPR Workshops, 2020, arXiv:1910.01279.

[15] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps," in Advances in Neural Information Processing Systems (NeurIPS), 2018, arXiv:1810.03292.

[16] P. Bergmann, K. Batzner, M. Fauser, D. Sattlegger, and C. Steger, "The MVTec anomaly detection dataset: A comprehensive real-world dataset for unsupervised anomaly detection," Int. J. Computer Vision, vol. 129, pp. 1038–1059, 2021, doi: 10.1007/s11263-020-01400-4.

[17] P. Bergmann, K. Batzner, M. Fauser, D. Sattlegger, and C. Steger, "Beyond dents and scratches: Logical constraints in unsupervised anomaly detection and localization," Int. J. Computer Vision, vol. 130, pp. 947–969, 2022, doi: 10.1007/s11263-022-01578-9.

[18] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, "Uninformed students: Student–teacher anomaly detection with discriminative latent embeddings," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2020, doi: 10.1109/CVPR42600.2020.00424.
[19] J. Liu et al., "Deep industrial image anomaly detection: A survey," Machine Intelligence Research, vol. 21, pp. 104–135, 2024, doi: 10.1007/s11633-023-1459-z.
[20] A. Radford et al., "Learning transferable visual models from natural language supervision," in Proc. Int. Conf. Machine Learning (ICML), 2021, doi: 10.48550/arXiv.2103.00020.
[21] Q. Zhou, G. Pang, Y. Tian, S. He, and J. Chen, "AnomalyCLIP: Object-agnostic prompt learning for zero-shot anomaly detection," in Proc. Int. Conf. Learning Representations (ICLR), 2024, doi: 10.48550/arXiv.2310.18961.
[22] Z. Gu, B. Zhu, G. Zhu, Y. Chen, M. Tang, and J. Wang, "AnomalyGPT: Detecting industrial anomalies using large vision-language models," in Proc. AAAI Conf. Artificial Intelligence, vol. 38, 2024, doi: 10.1609/aaai.v38i3.27963.
[23] M. Oquab et al., "DINOv2: Learning robust visual features without supervision," Trans. Machine Learning Research, 2024, doi: 10.48550/arXiv.2304.07193.
[24] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?': Explaining the predictions of any classifier," in Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2016, pp. 1135–1144, doi: 10.1145/2939672.2939778.
[25] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Advances in Neural Information Processing Systems (NeurIPS), 2017, doi: 10.48550/arXiv.1705.07874.
[26] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in Proc. Int. Conf. Machine Learning (ICML), 2017, doi: 10.48550/arXiv.1703.01365.
[27] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," in Proc. Int. Conf. Learning Representations (ICLR) Workshop, 2014, doi: 10.48550/arXiv.1312.6034.
[28] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2921–2929, doi: 10.1109/CVPR.2016.319.
[29] L. R. Dice, "Measures of the amount of ecologic association between species," Ecology, vol. 26, no. 3, pp. 297–302, 1945, doi: 10.2307/1932409.
[30] P. Jaccard, "The distribution of the flora in the alpine zone," New Phytologist, vol. 11, no. 2, pp. 37–50, 1912, doi: 10.1111/j.1469-8137.1912.tb05611.x.