Supplemental Material:
Fully Convolutional Cross-Scale-Flows for Image-based Defect Detection

Marco Rudolph[1] Tom Wehrbein[1] Bodo Rosenhahn[1] Bastian Wandt[2]
[1]Leibniz University Hannover, Germany [2]University of British Columbia, Canada
rudolph@tnt.uni-hannover.de

A. Localization

A.1. Experimental Setup

In the following, we show detailed results of the defect localization provided by our method, as described in Section 3.3 of the paper. Figures 1 to 44 show every fourth defective image from the MVTec AD dataset [1] to prevent a selection bias. The original images from the dataset are shown in the upper rows. Corresponding to Figure 7 in the paper, we show the output $z^{(0)}$ in the middle rows by visualizing the sum of squares over the channels for each local position. The intensities of the colormap are normalized per image. The bottom rows show the overlay of both top rows for better visual alignment. The product category and defect type are described above each column. Note that we here only used 3×3 convolutions in the internal networks to limit the dilatation effect which is described in the following analysis.

A.2. Analysis

Our method detects various defects, *e.g*. changes in color, texture and shape. For images from the category *texture* it works almost flawless as shown in Figures 1 to 13 and an average image-level detection AUROC of $99.8\,\%$ (cf. Table 1 in the paper). In Figure 6, the structure of the crack is accurately reproduced in the output maps. Even subtle anomalies, such as the small metal contamination in the left image of Figure 11, are detected.

For the object categories, in the vast majority of cases, the region with the error is highlighted. Interestingly, we even detect slight anomalies in images that were not labeled by the designer of the dataset: For example, we spot small irregularities on the edges of the zippers in the 3 right images of Figure 28. Note that originally only the defects in the interior surface were annotated in the dataset.

The maps of images from object categories are not as clean as for the textures in some cases. There are very few false positive highlights (see the rightmost image of Figure 22). The receptive field of the convolutions and the size of the output maps (24x24 compared to image sizes up to 1024x1024 pixels) lets the predicted defect area dilate in some cases as in the rightmost image of Figure 15. Nevertheless, it highlights the anomalies and can therefore be used in practice to find and analyze defects.

B. Inference Time

For inference time measurements, we run code with *PyTorch 1.5* on a *NVIDIA GTX 1080 Ti (CUDA 11.2)*. The tests were performed on MVTec AD [1] images using the hyperparameters described in Section 4.1 of the paper. The inference time for feature extraction on 3 scales with EfficientNet-B5 [2] is $85\,ms$. For density estimation with CS-Flow, the inference time is $7.4\,ms$. Thus, the feature extractor is the bottleneck of performance.

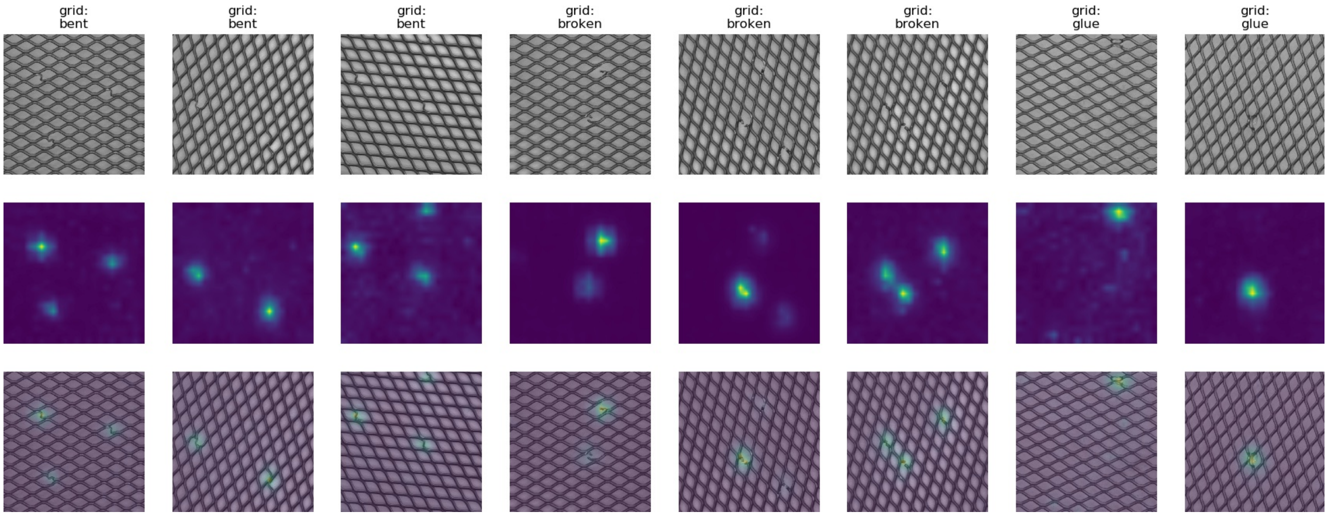

Figure 1.

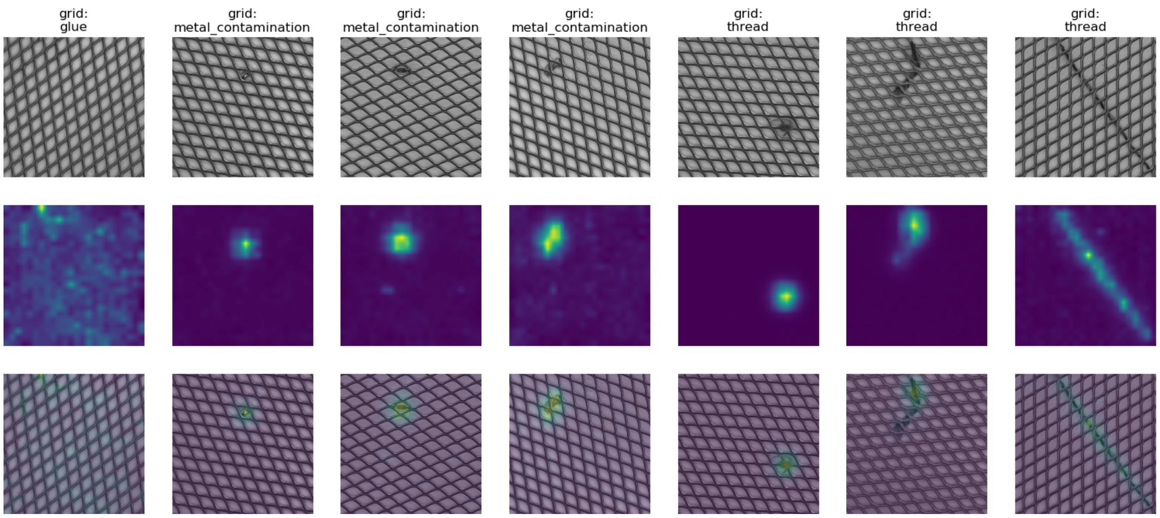

Figure 2.

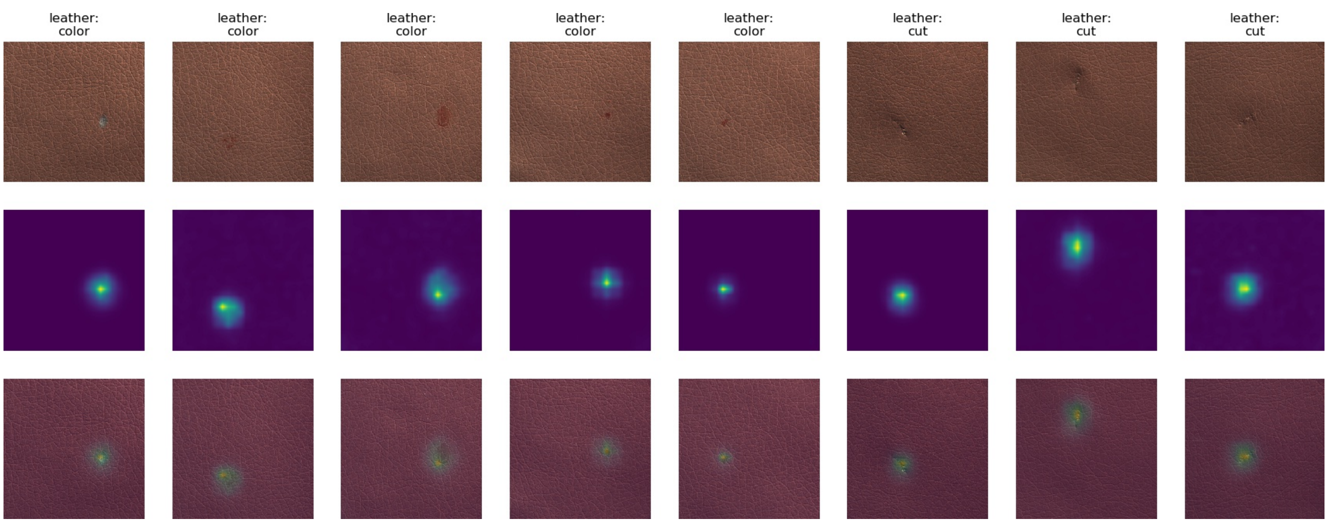

Figure 3.

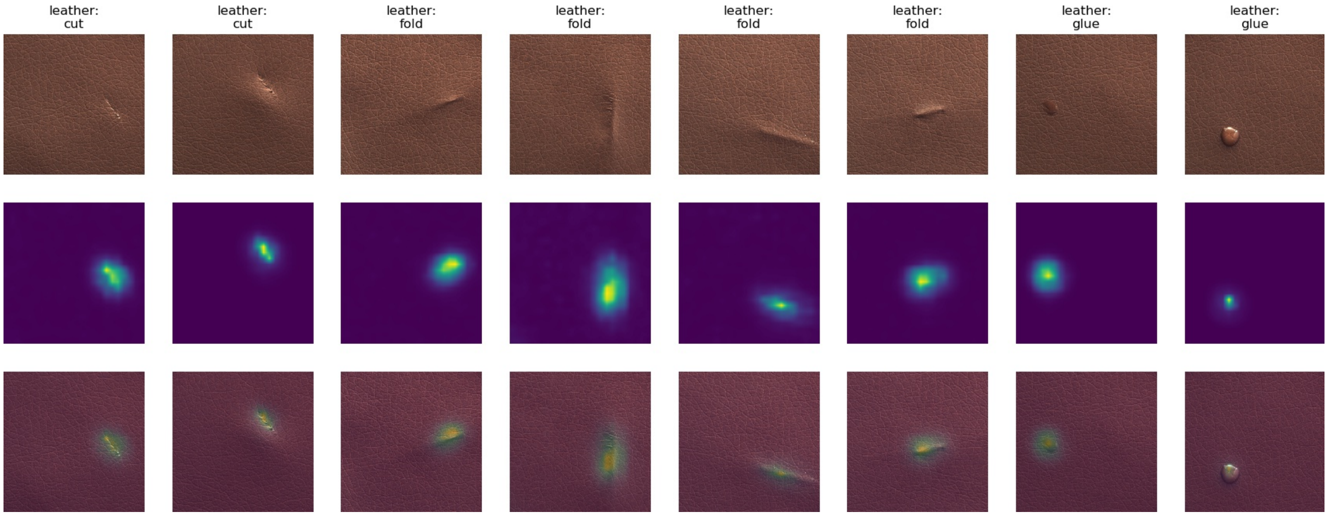

Figure 4.

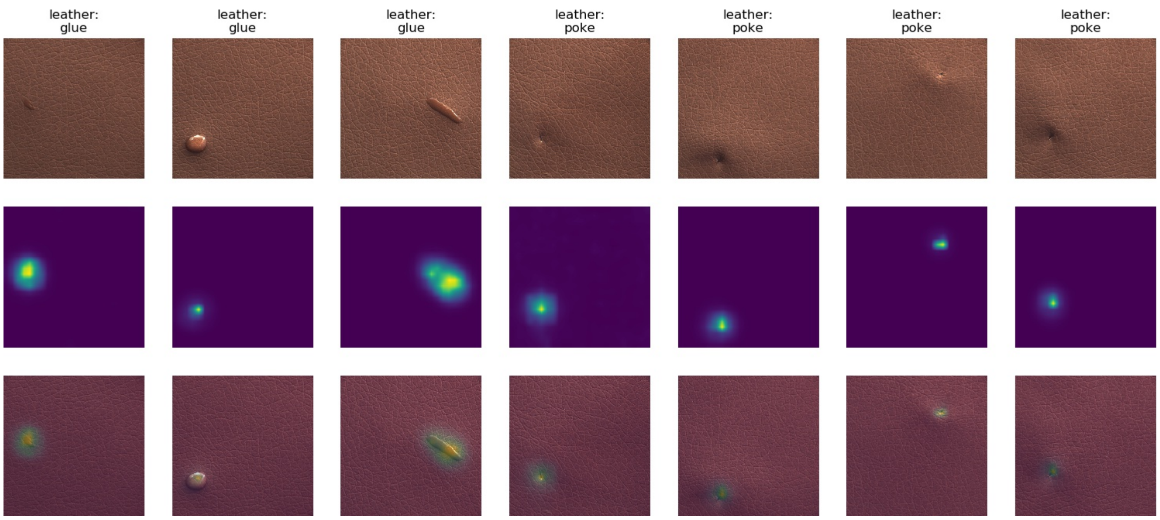

Figure 5.

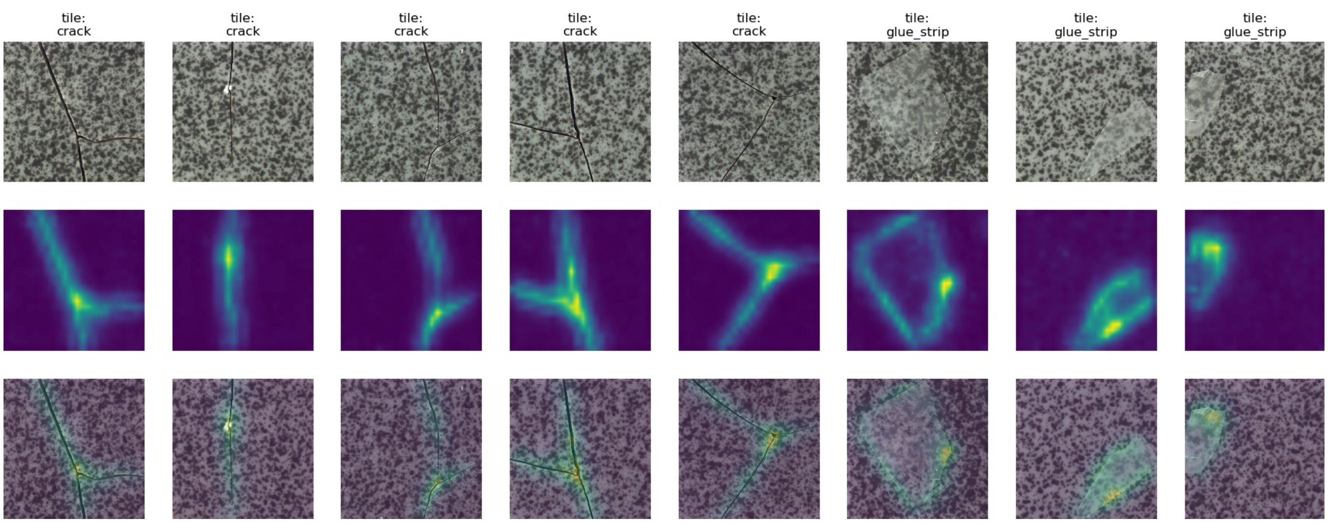

Figure 6.

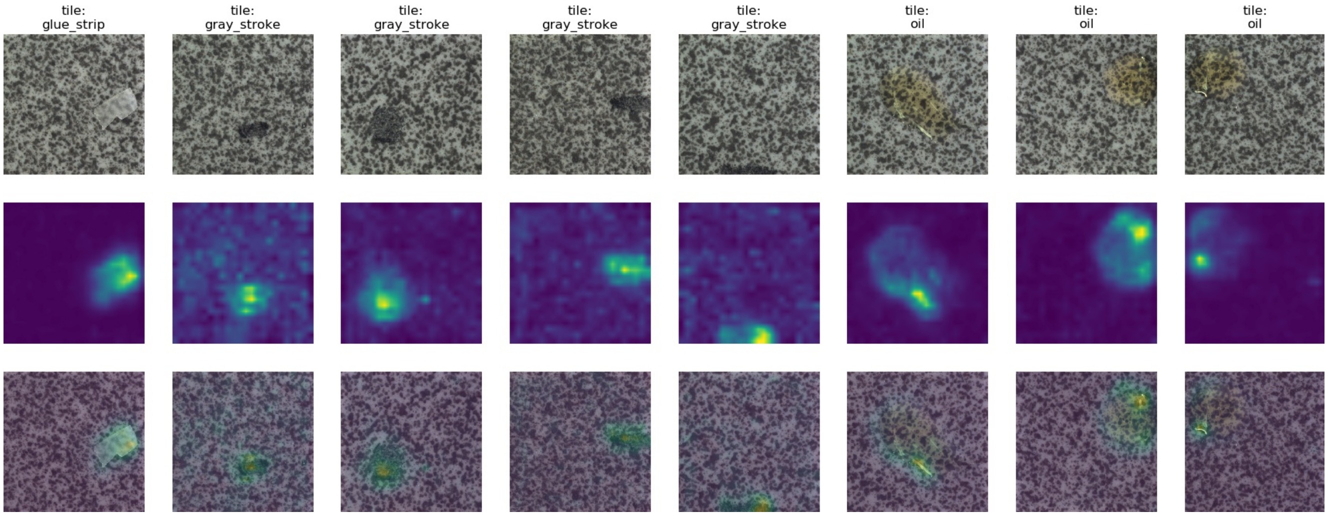

Figure 7.

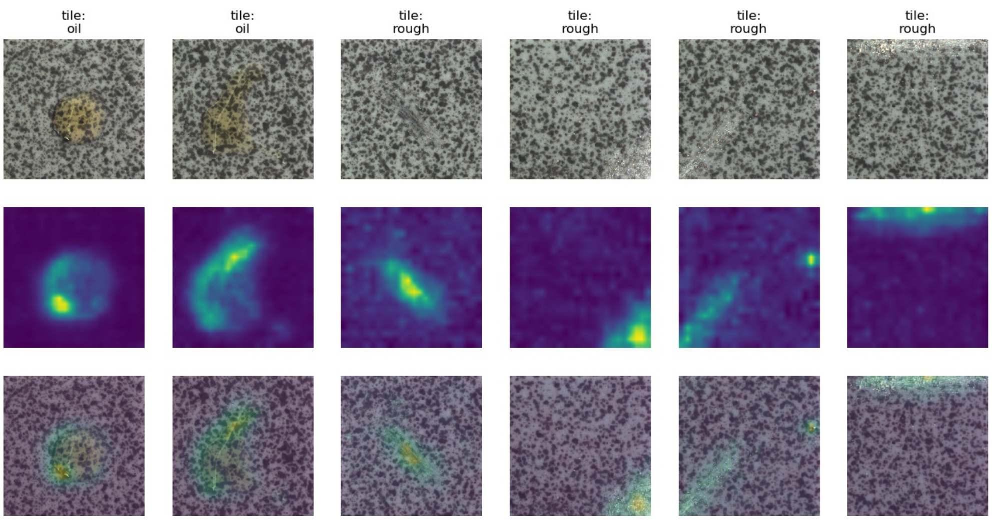

Figure 8.

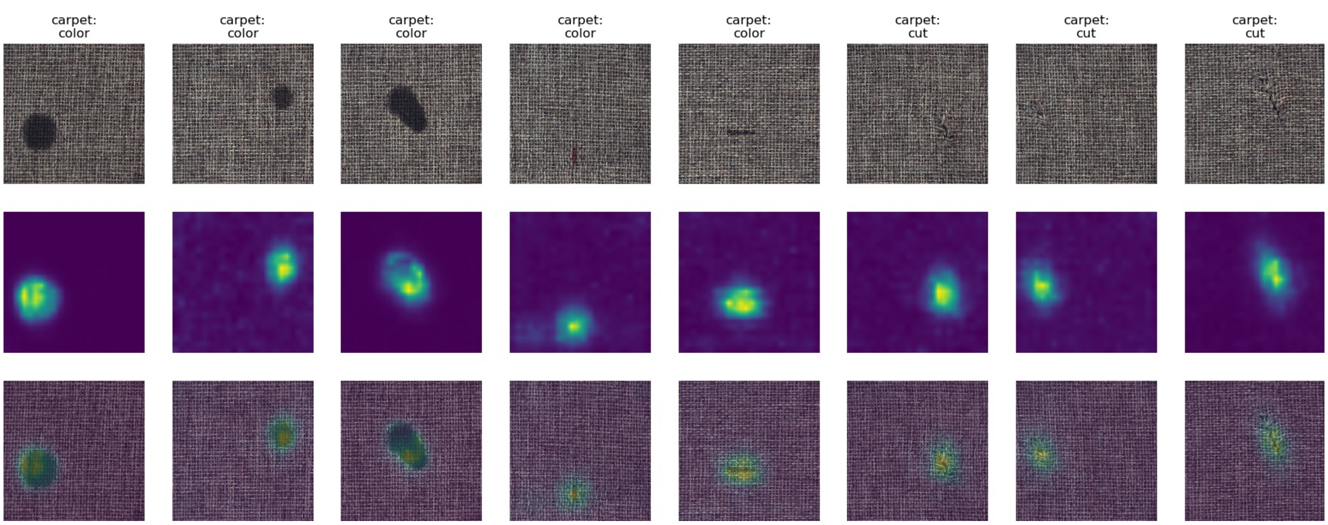

Figure 9.

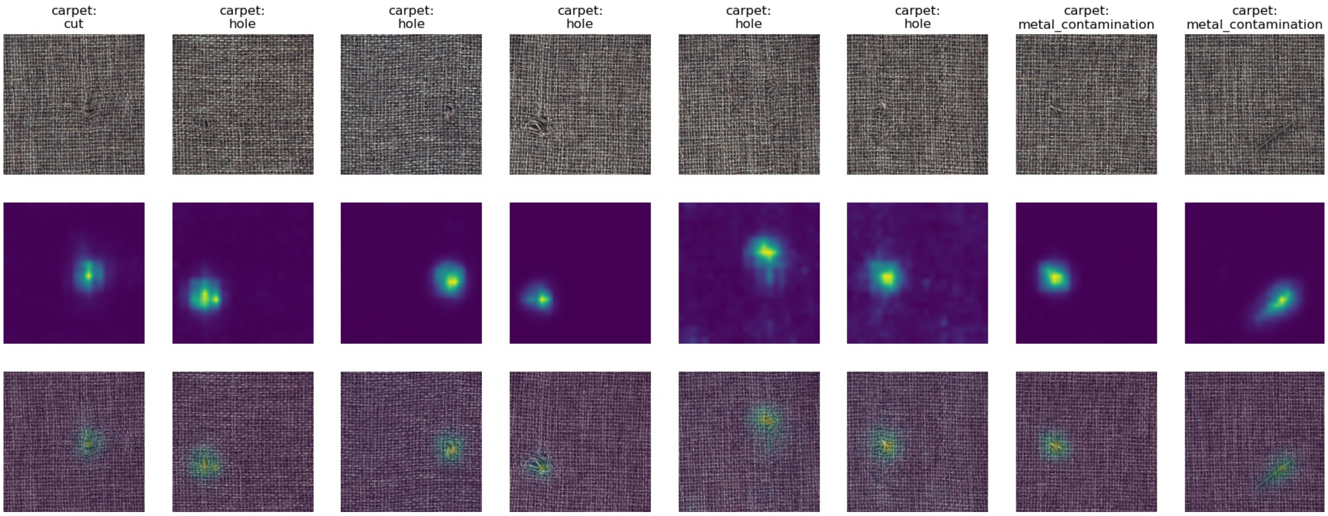

Figure 10.

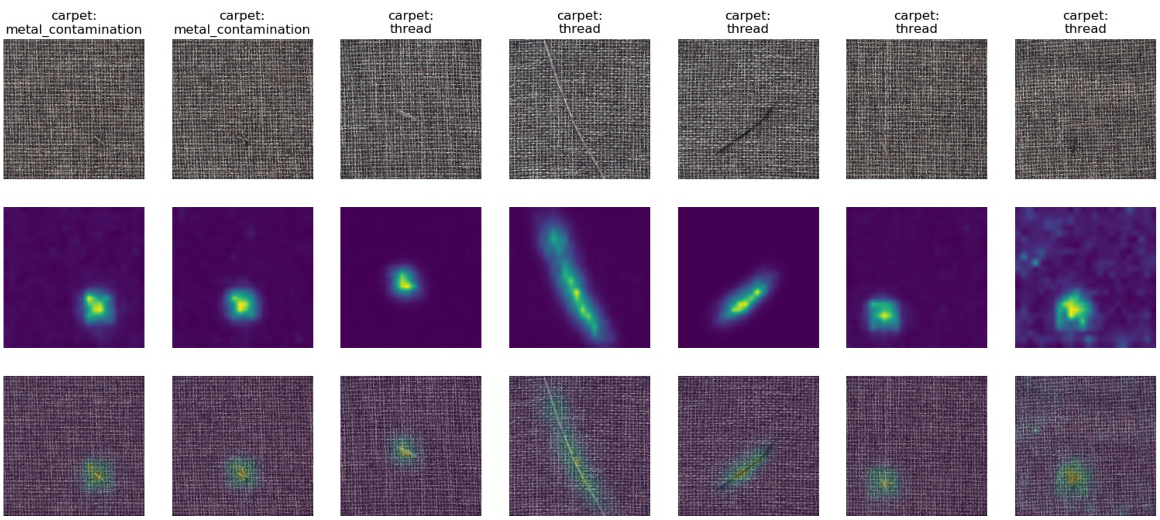

Figure 11.

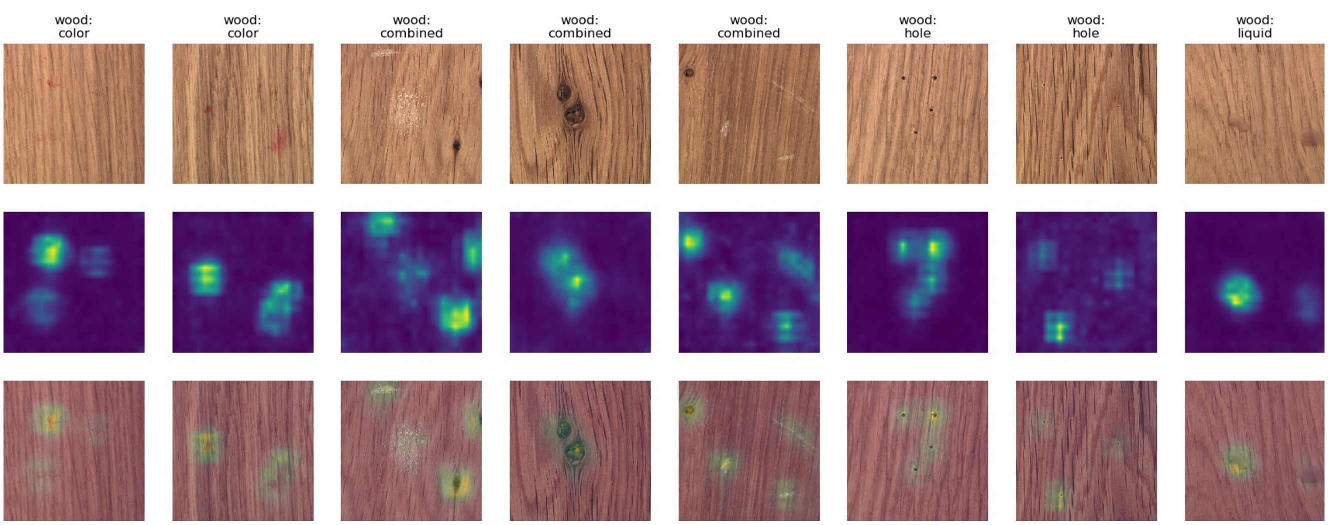

Figure 12.

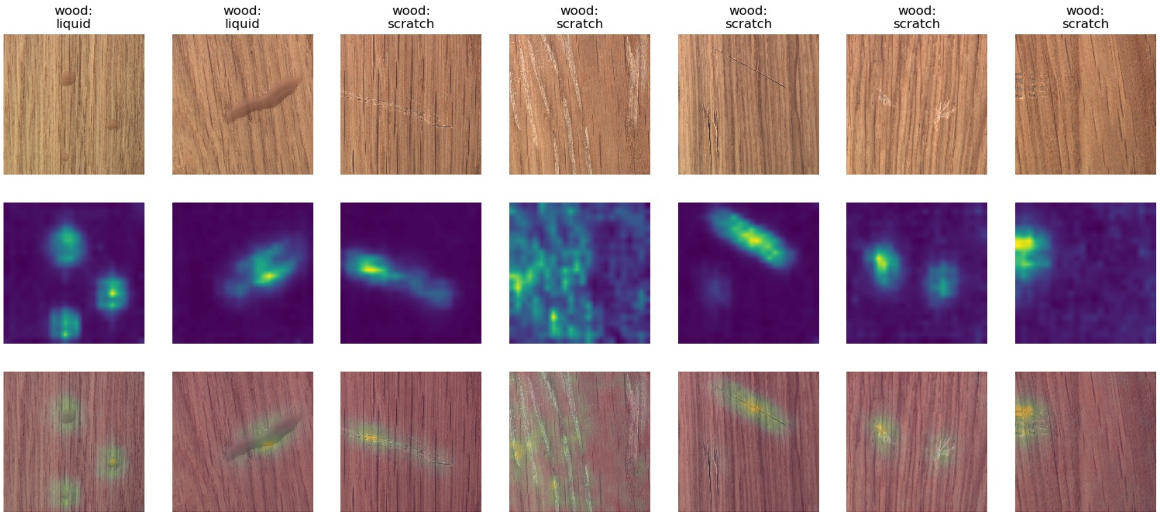

Figure 13.

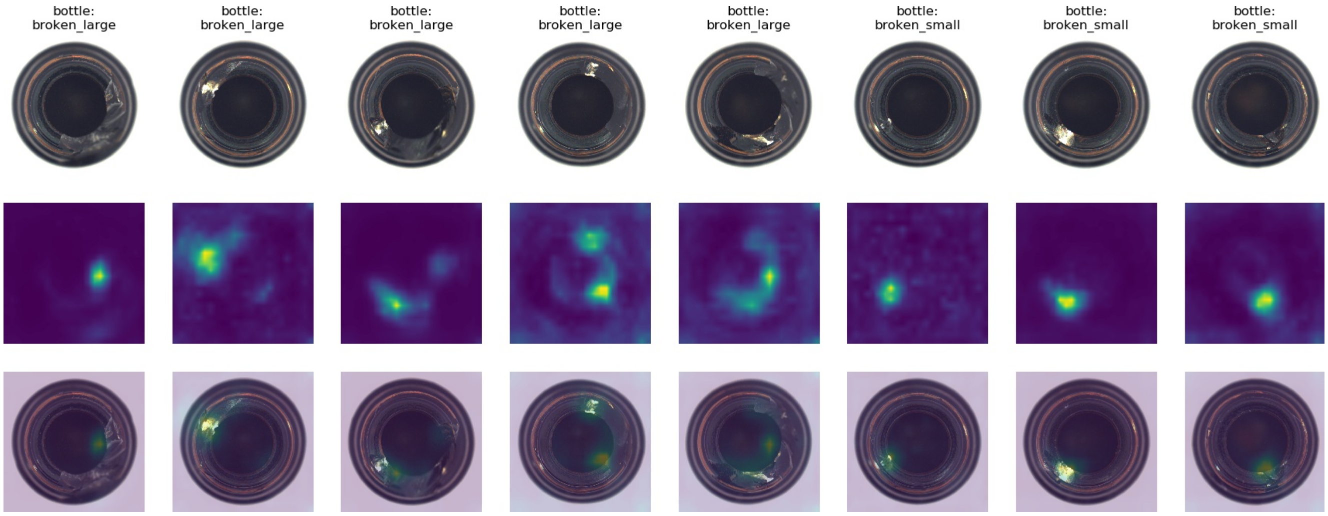

Figure 14.

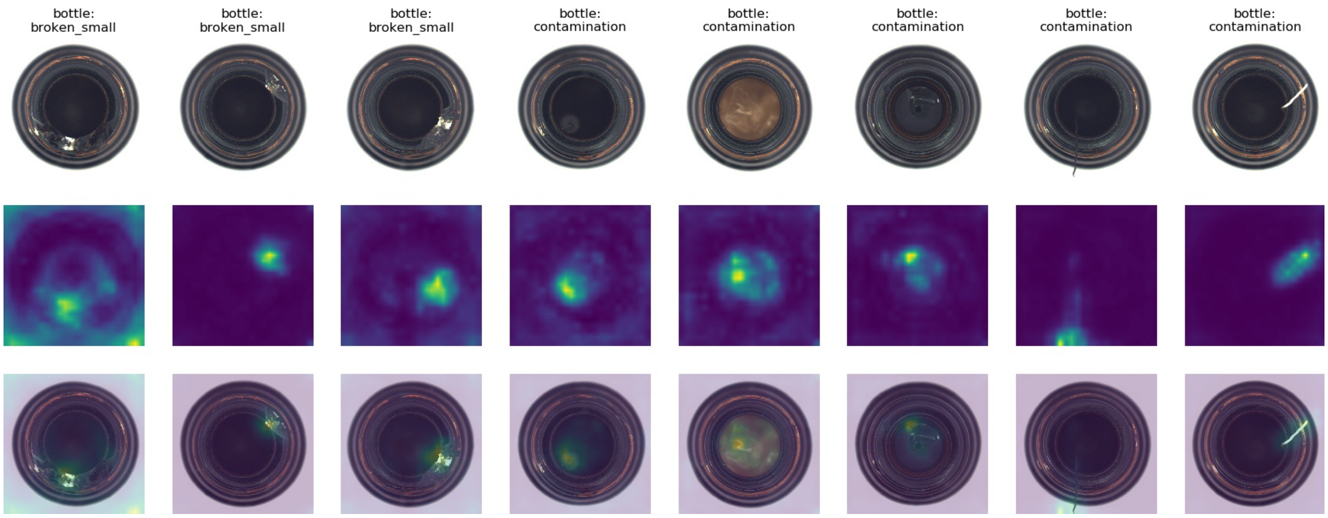

Figure 15.

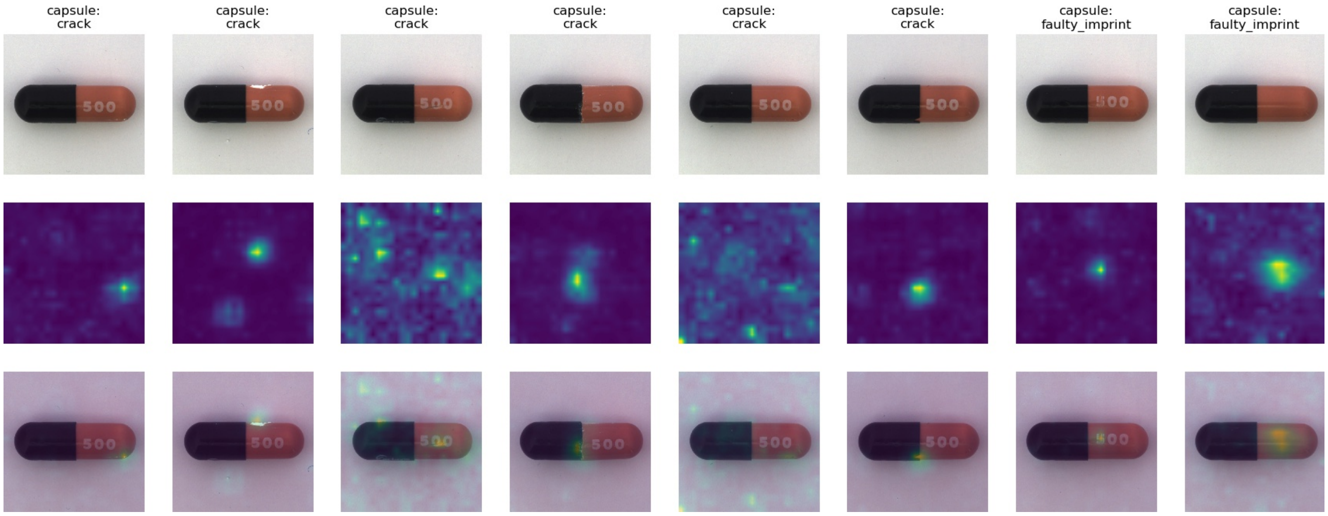

Figure 16.

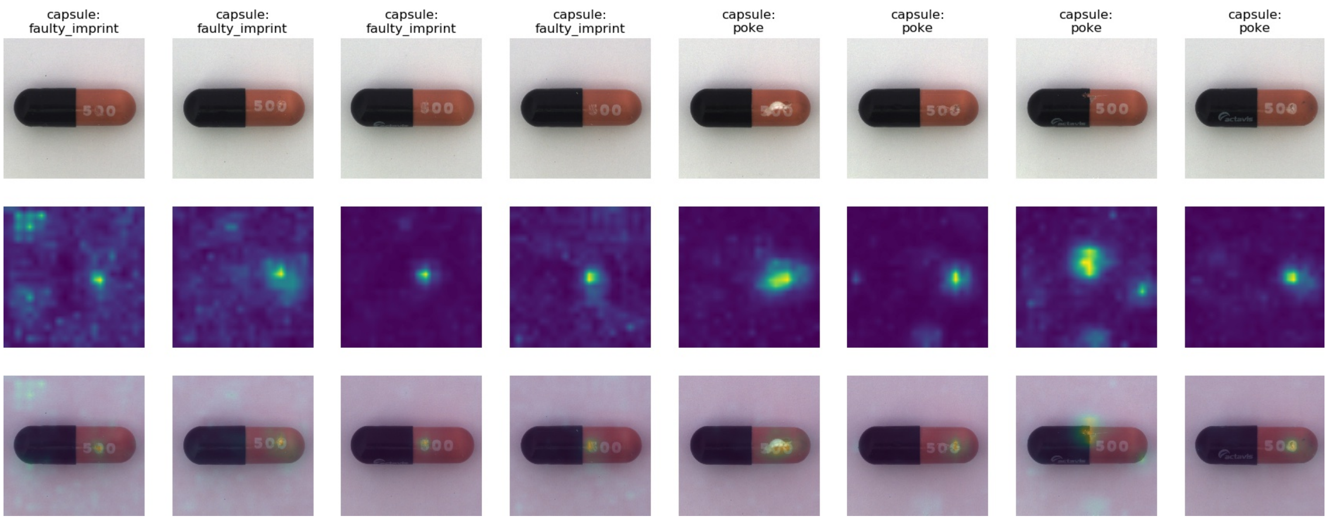

Figure 17.

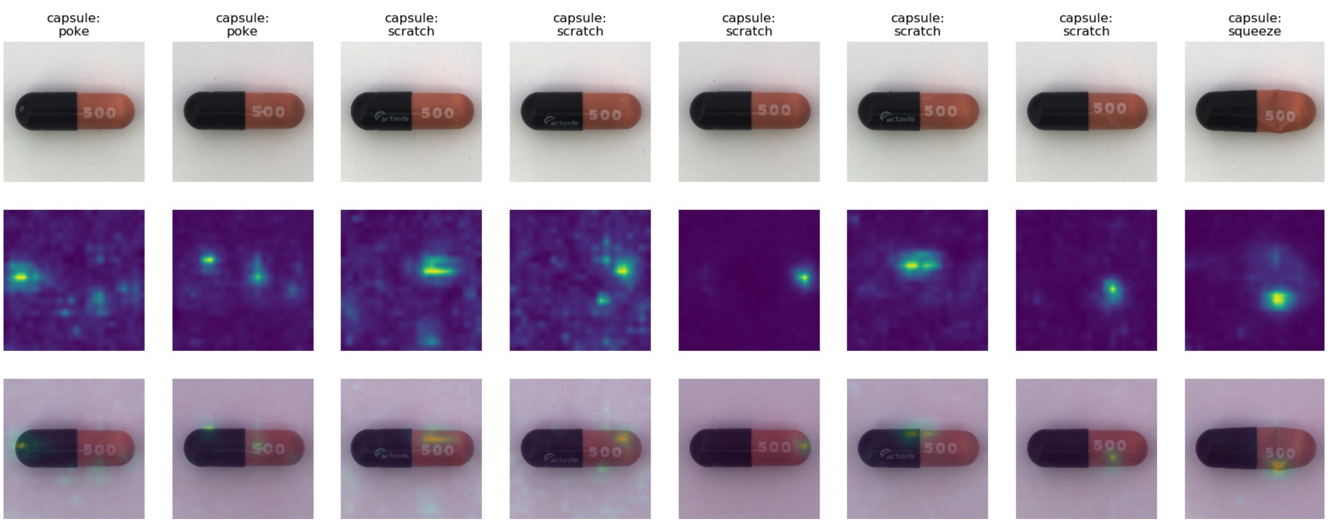

Figure 18.

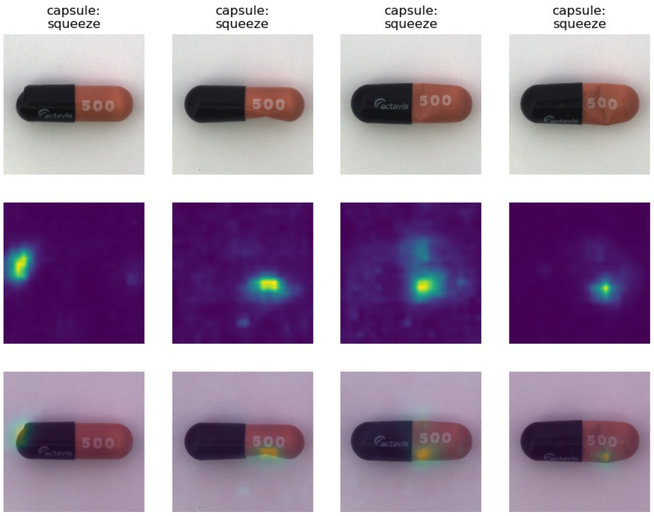

Figure 19.

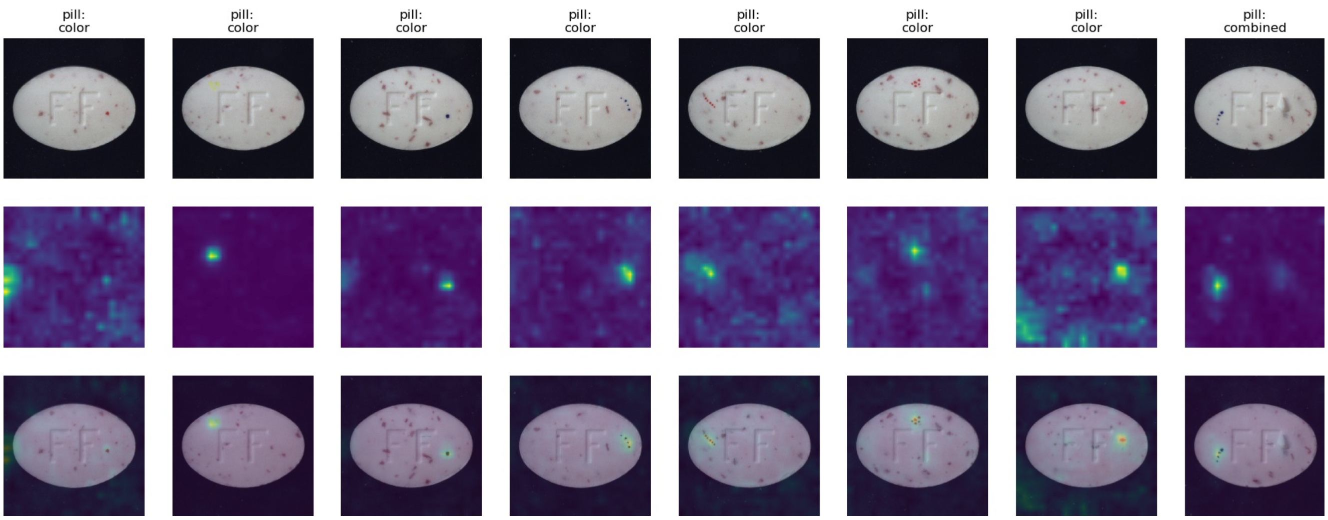

Figure 20.

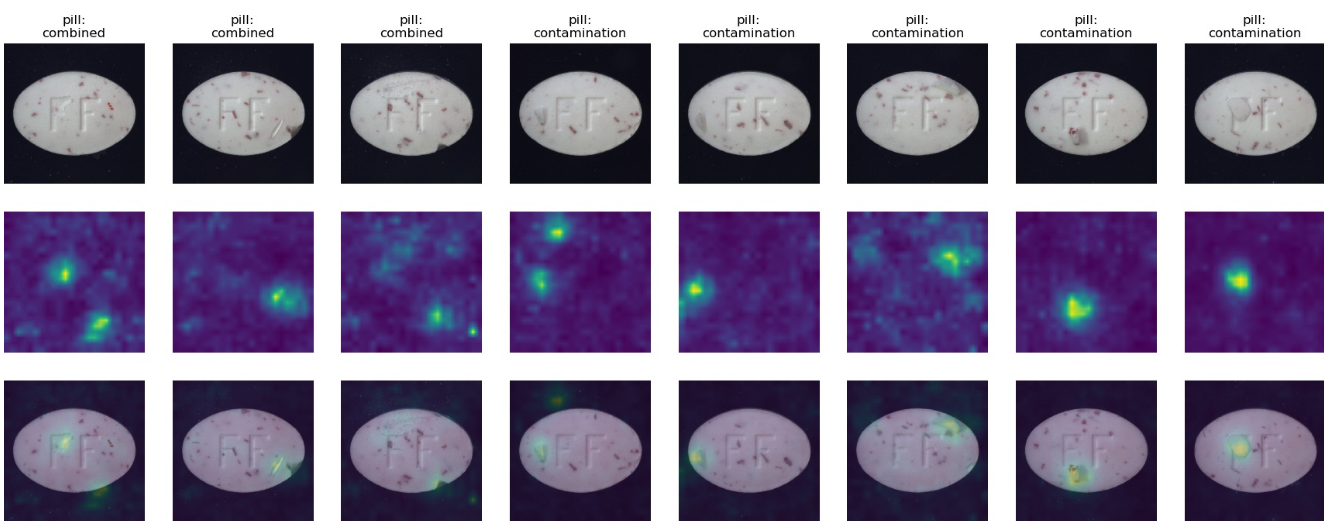

Figure 21.

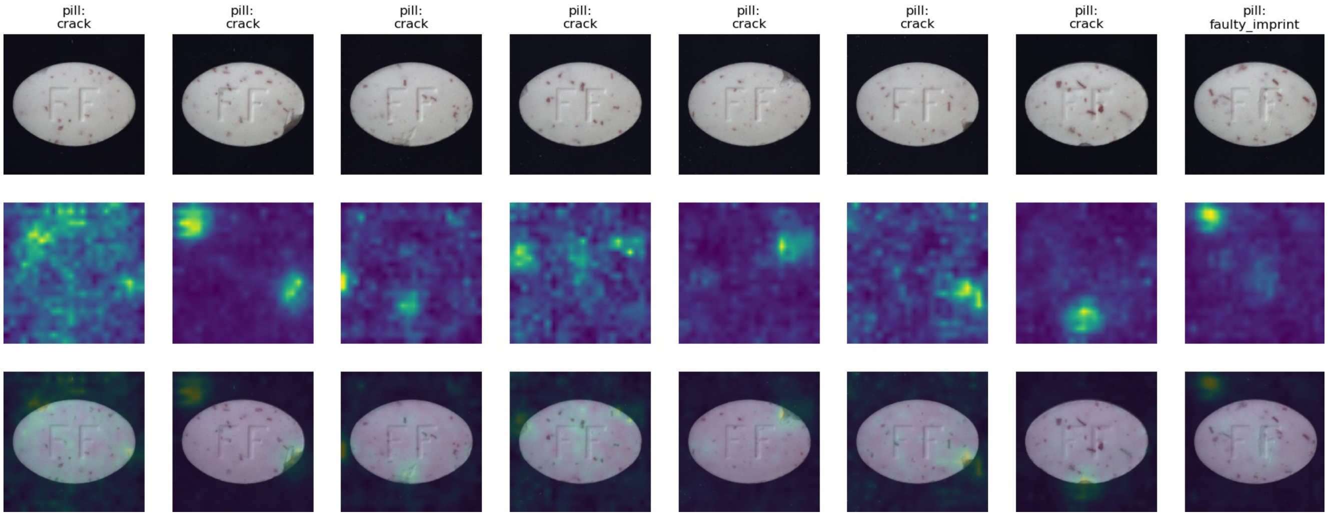

Figure 22.

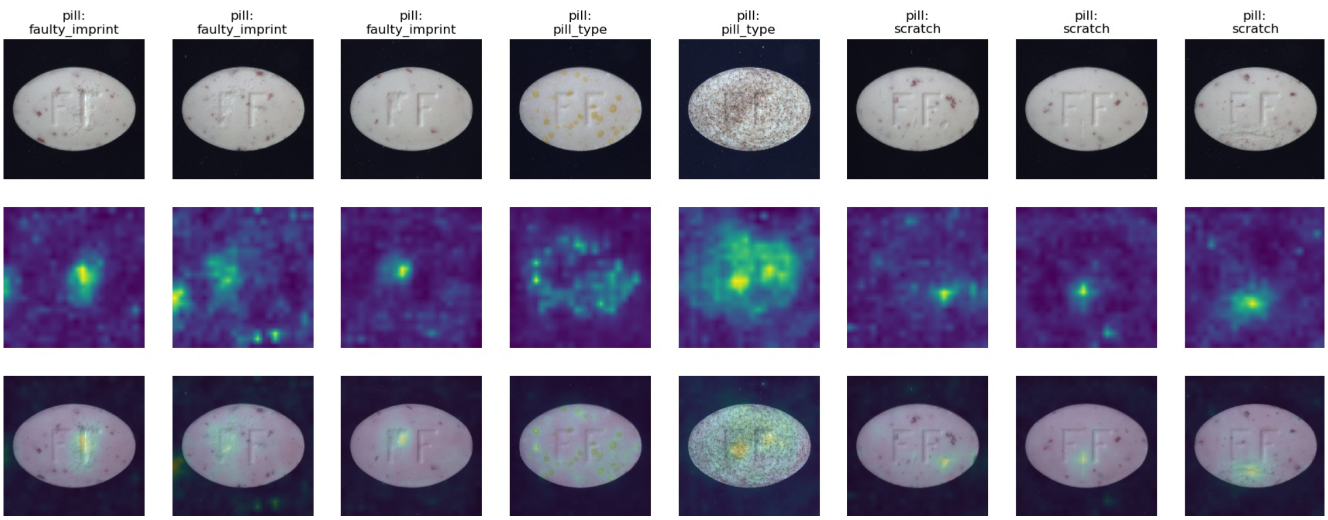

Figure 23.

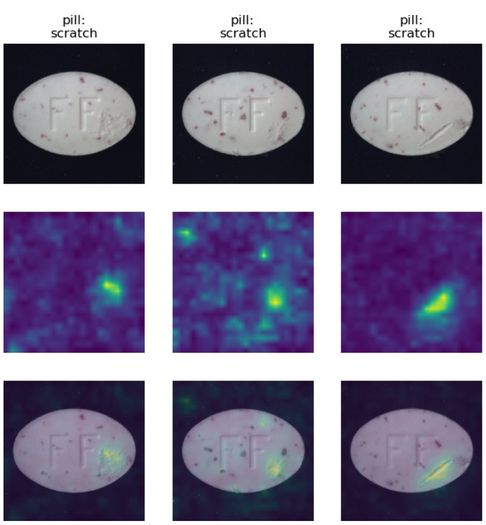

Figure 24.

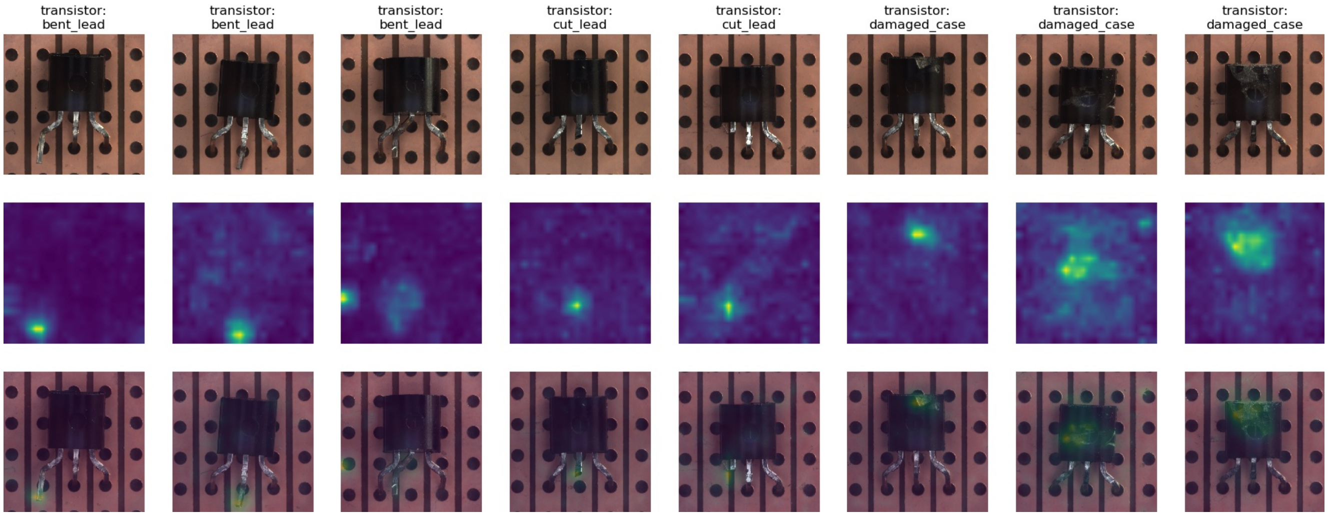

Figure 25.

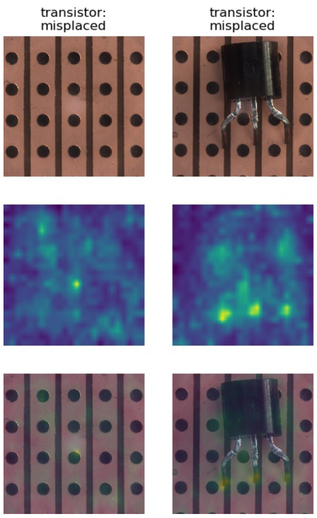

Figure 26.

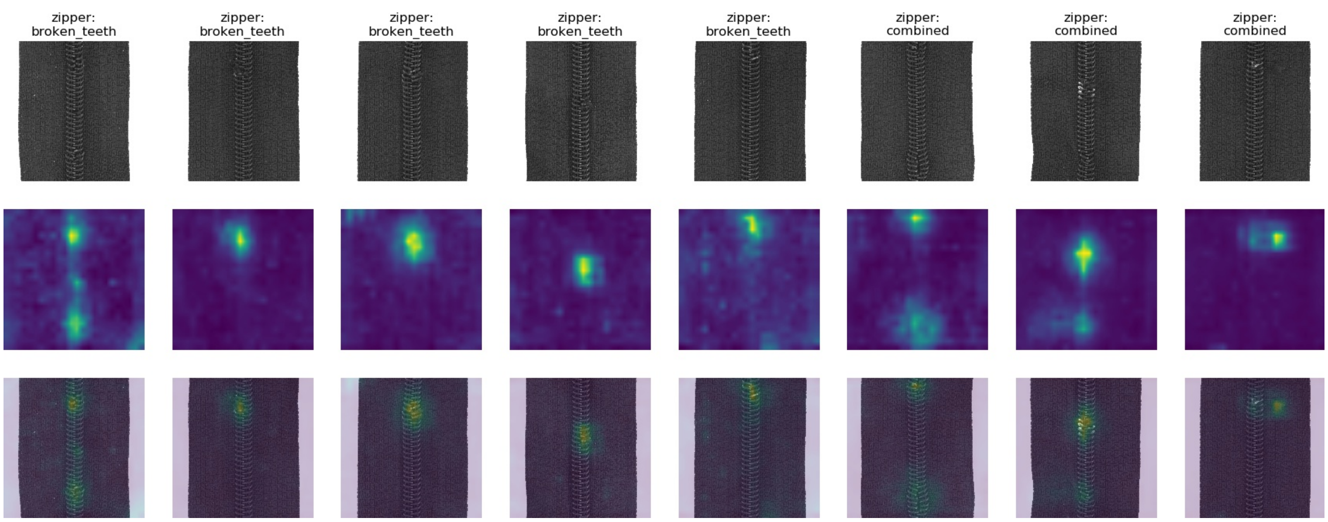

Figure 27.

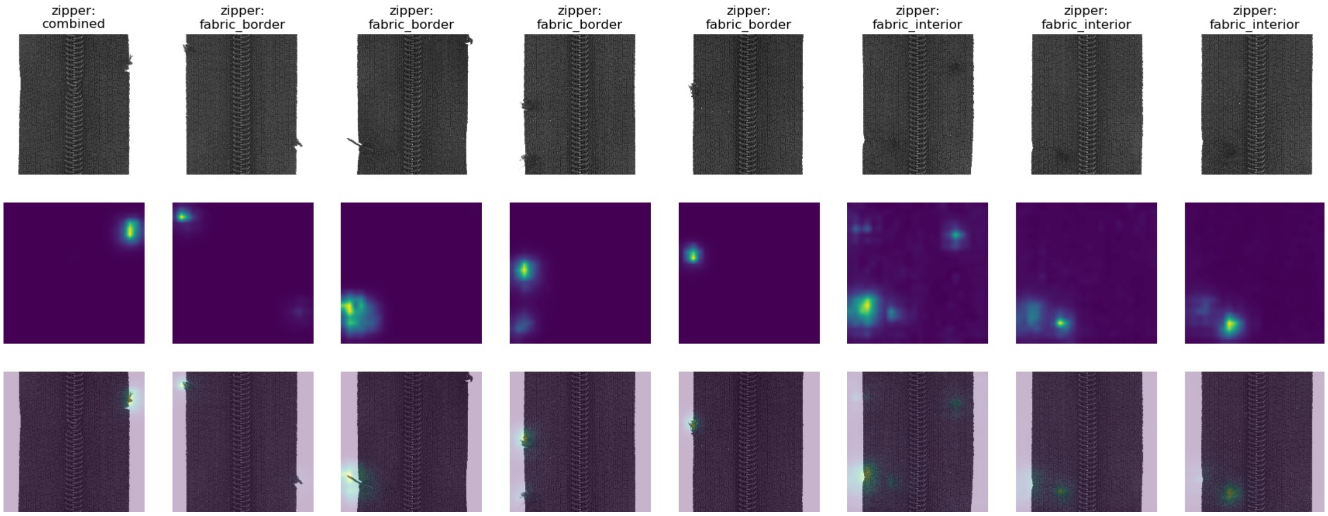

Figure 28.

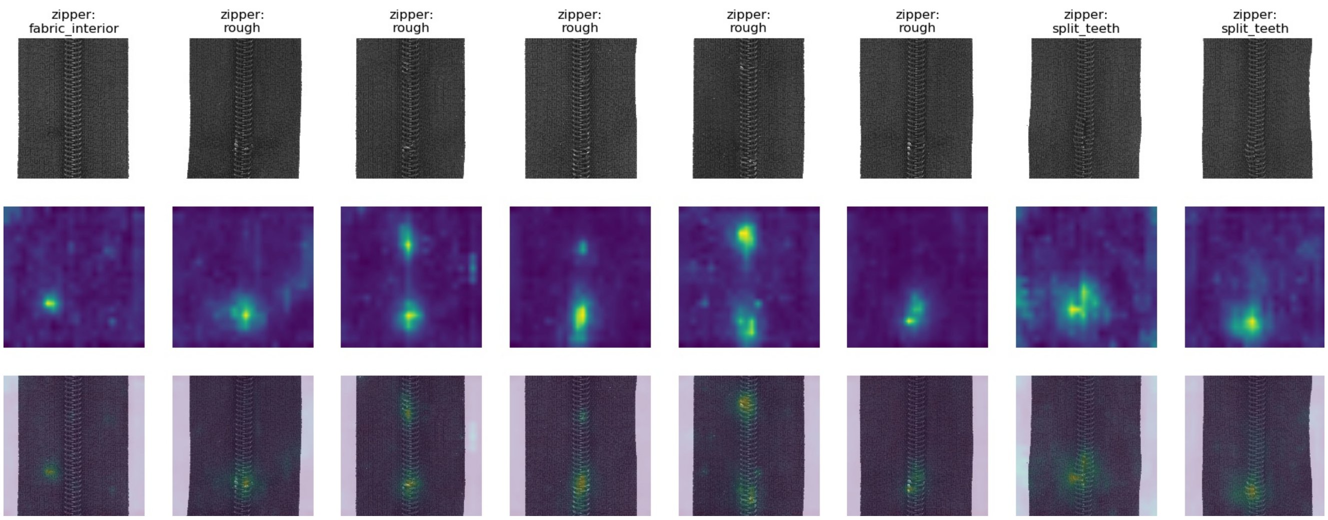

Figure 29.

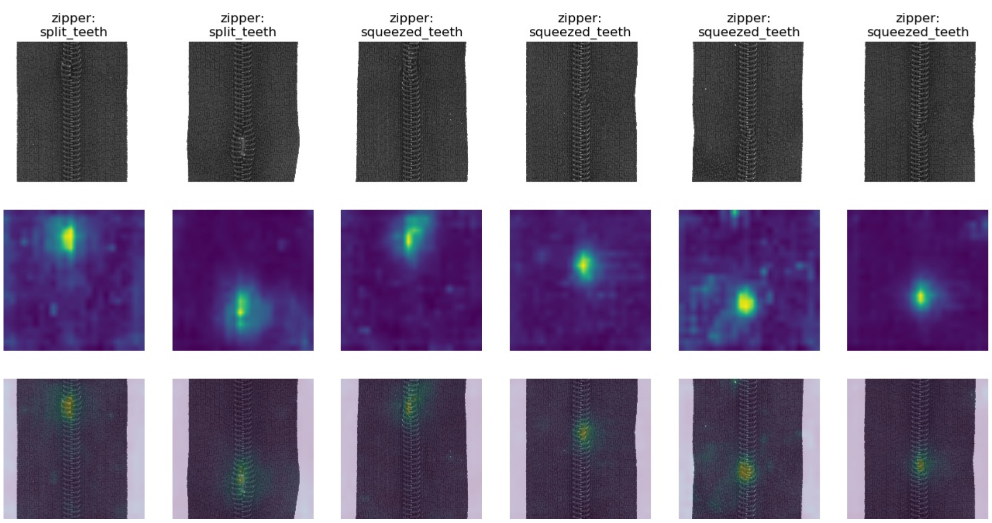

Figure 30.

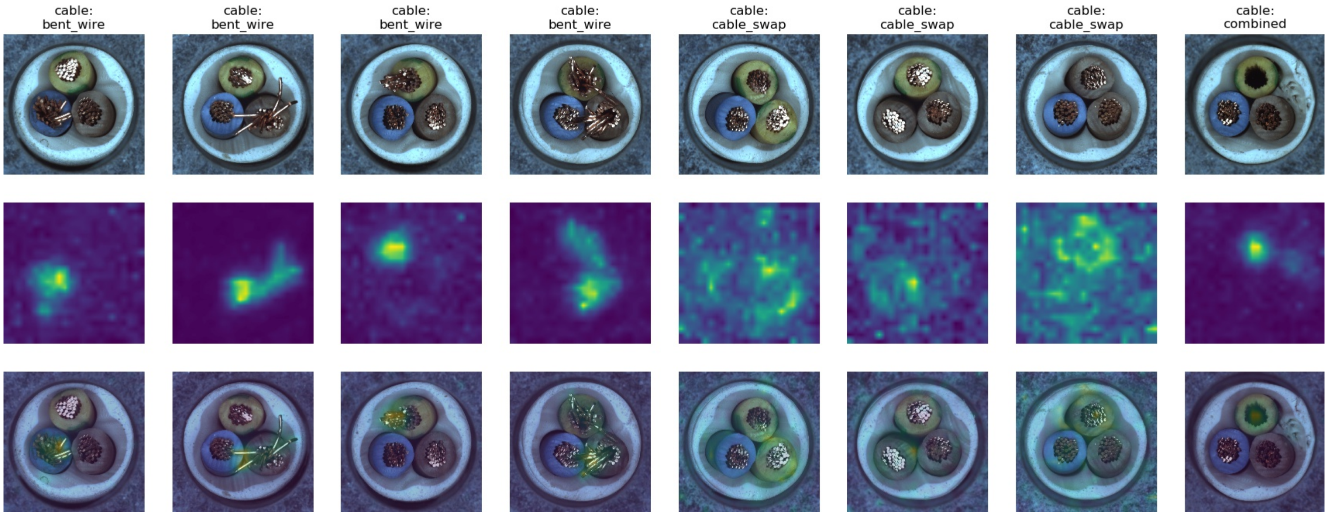

Figure 31.

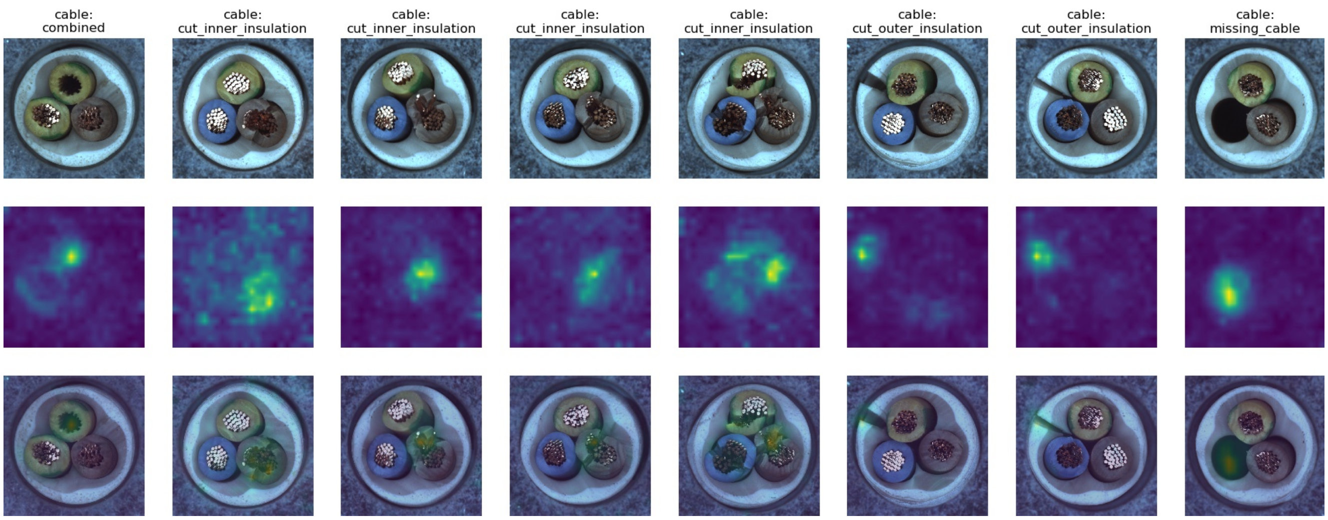

Figure 32.

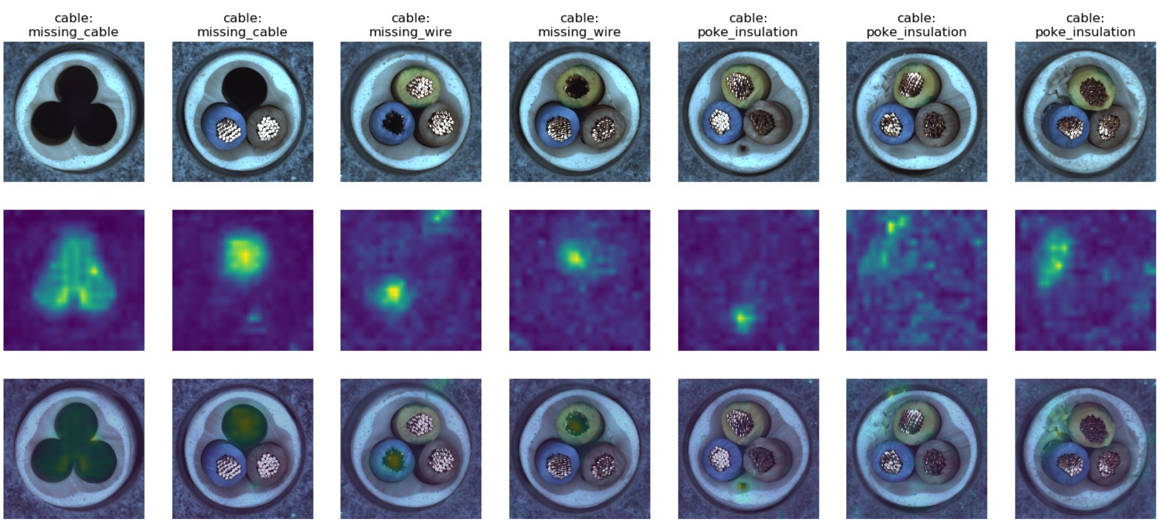

Figure 33.

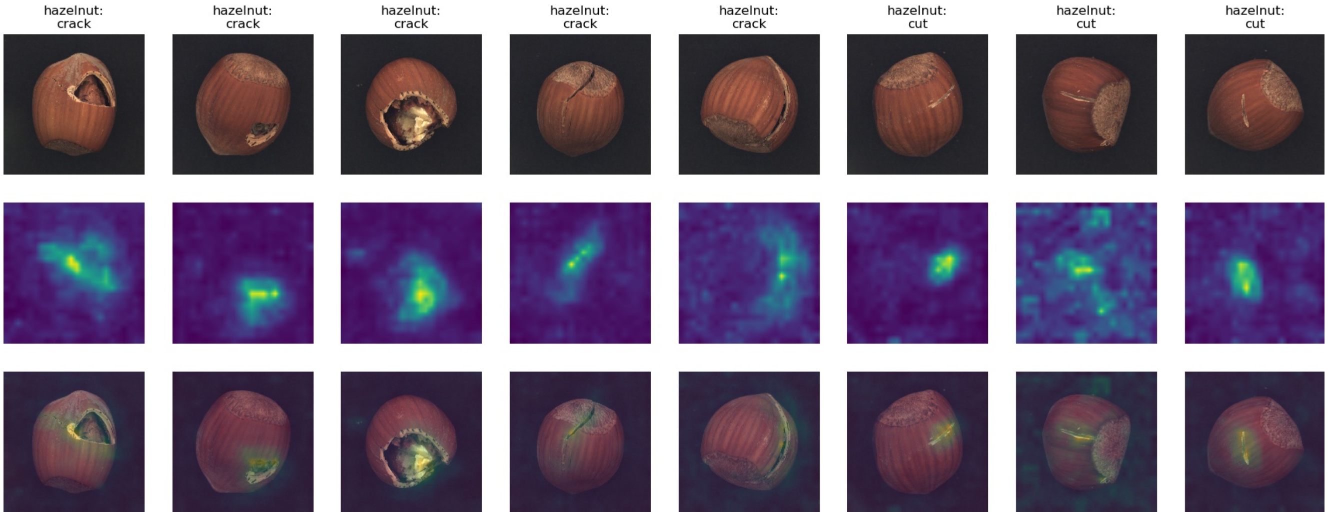

Figure 34.

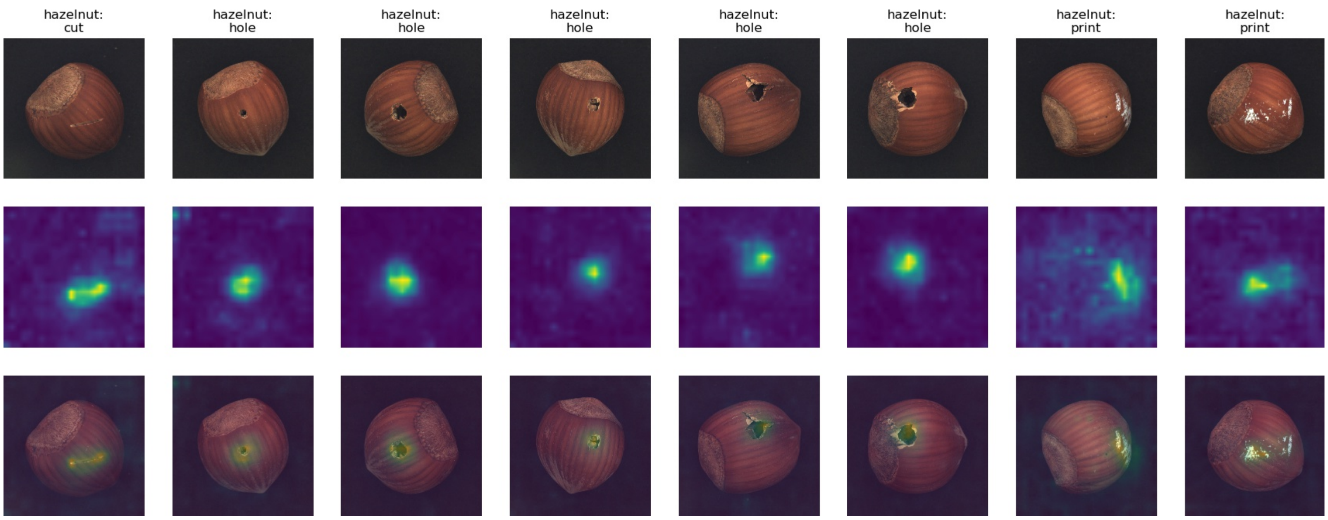

Figure 35.

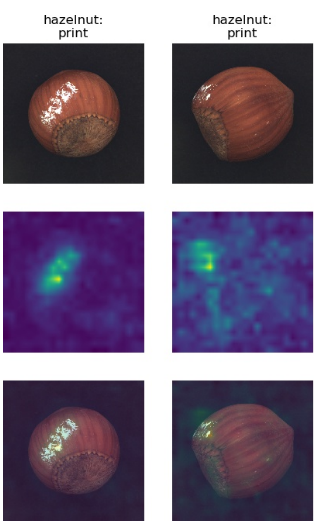

Figure 36.

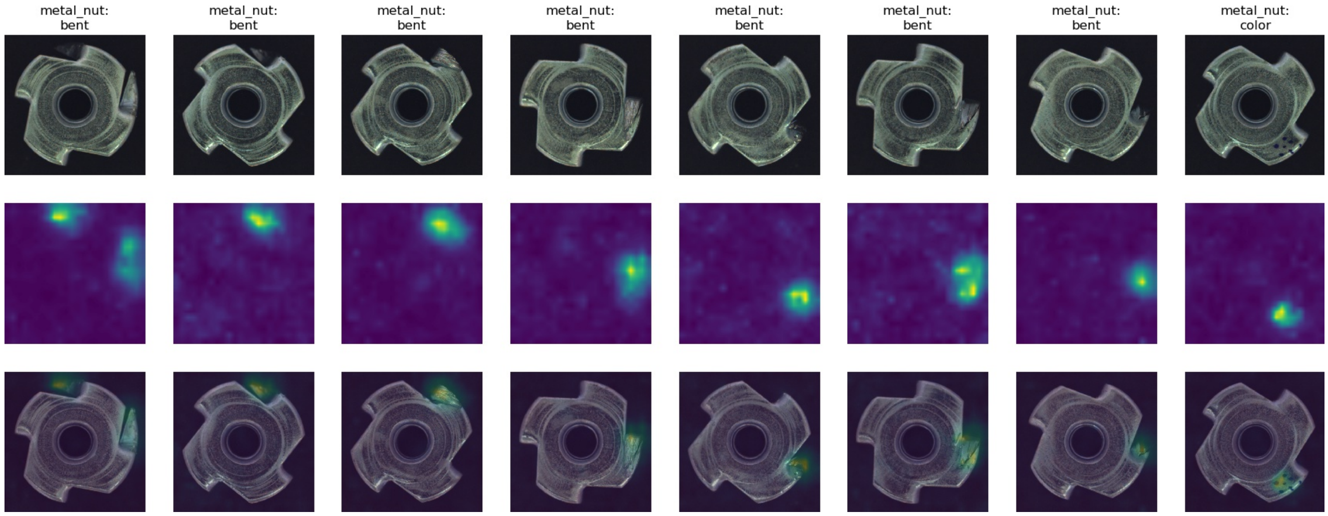

Figure 37.

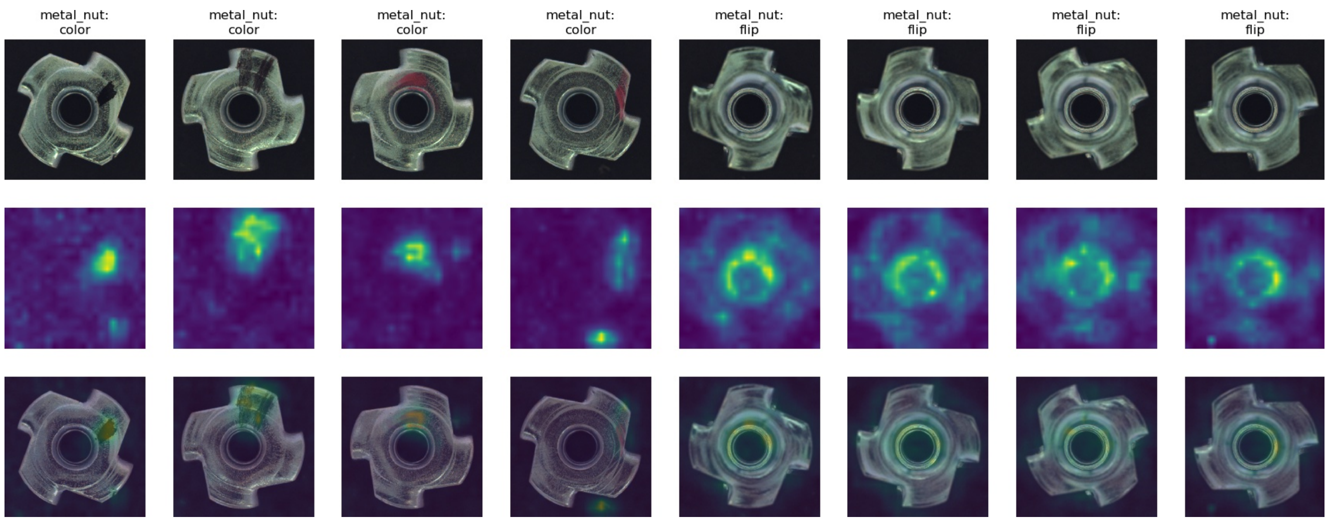

Figure 38.

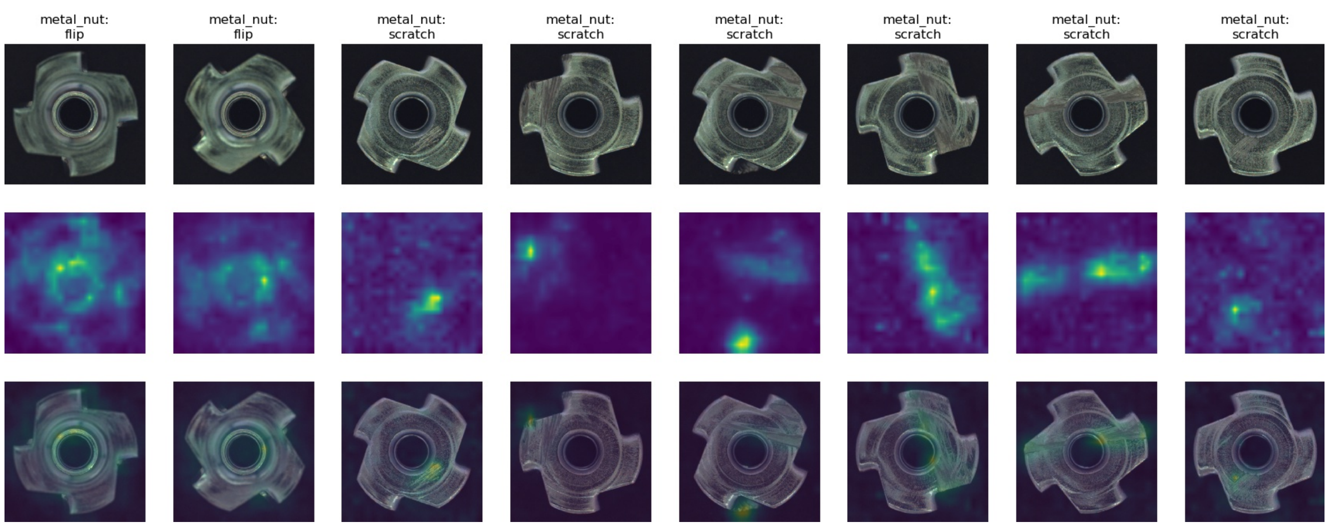

Figure 39.

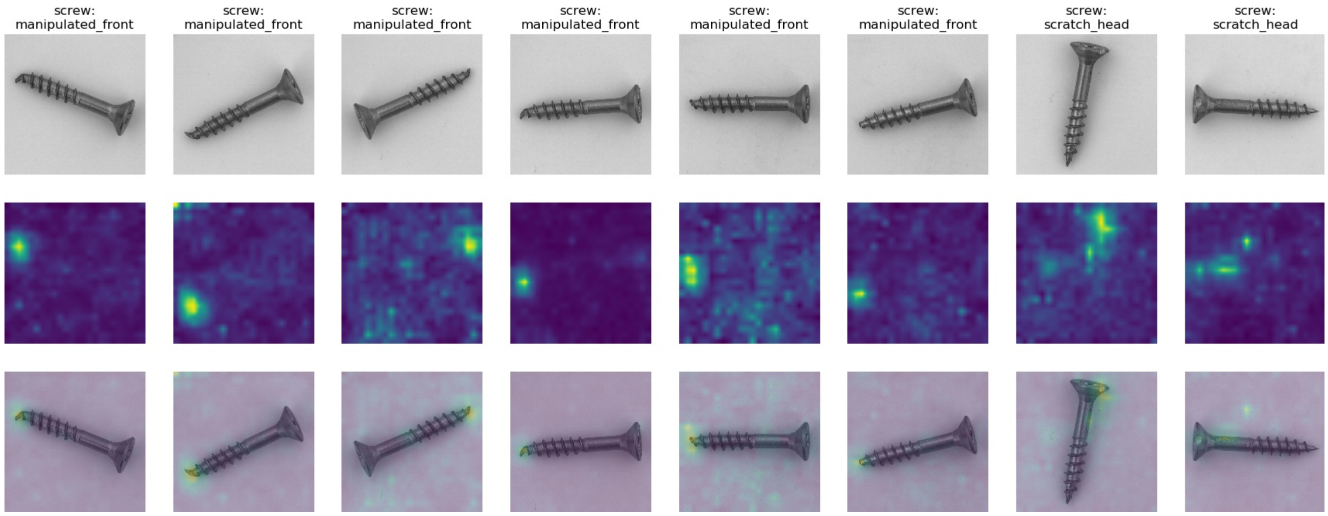

Figure 40.

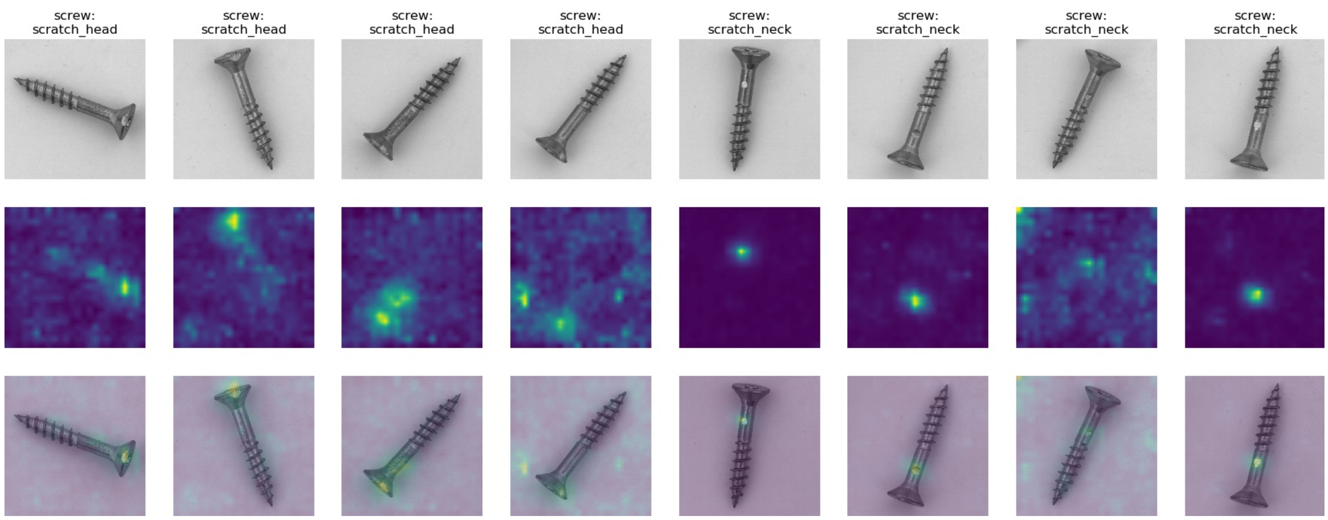

Figure 41.

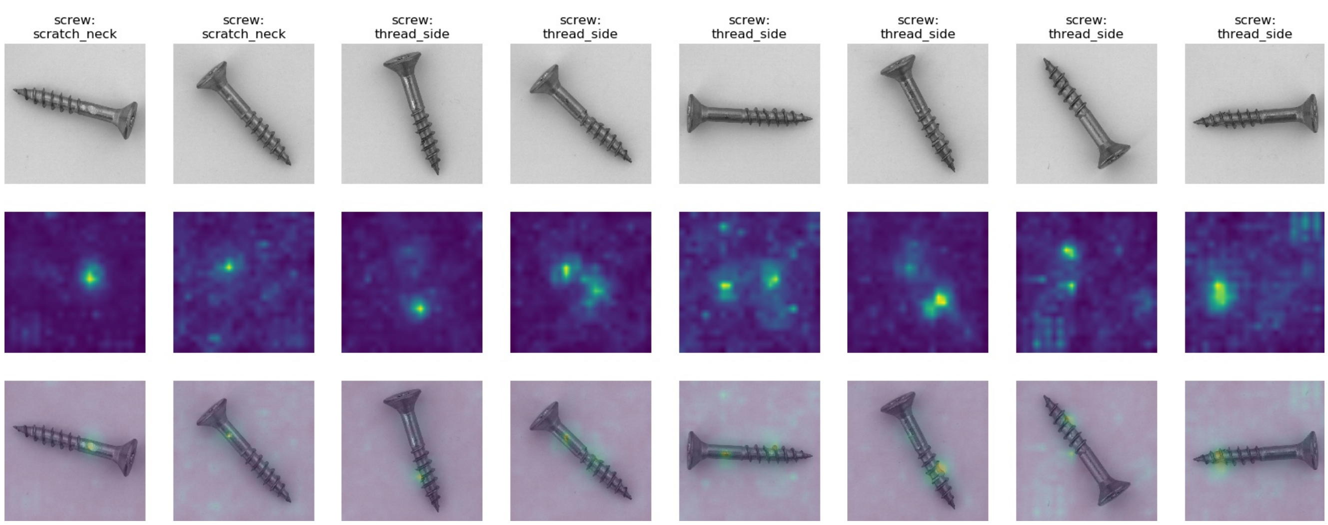

Figure 42.

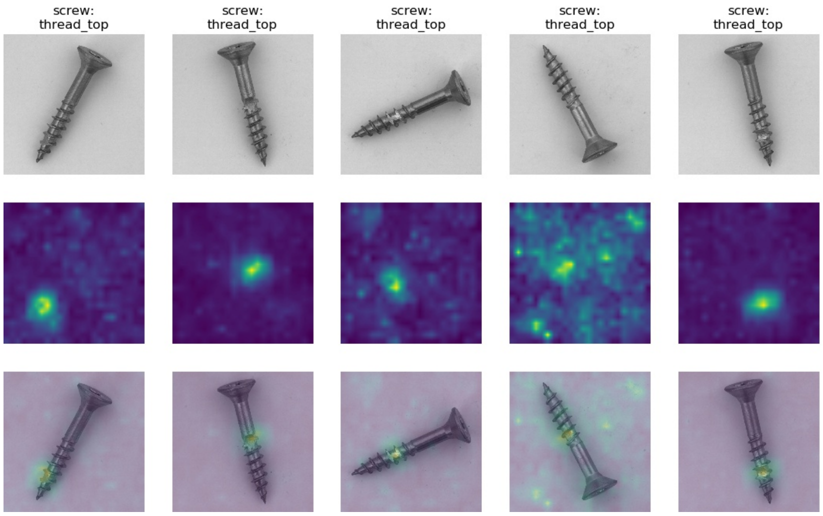

Figure 43.

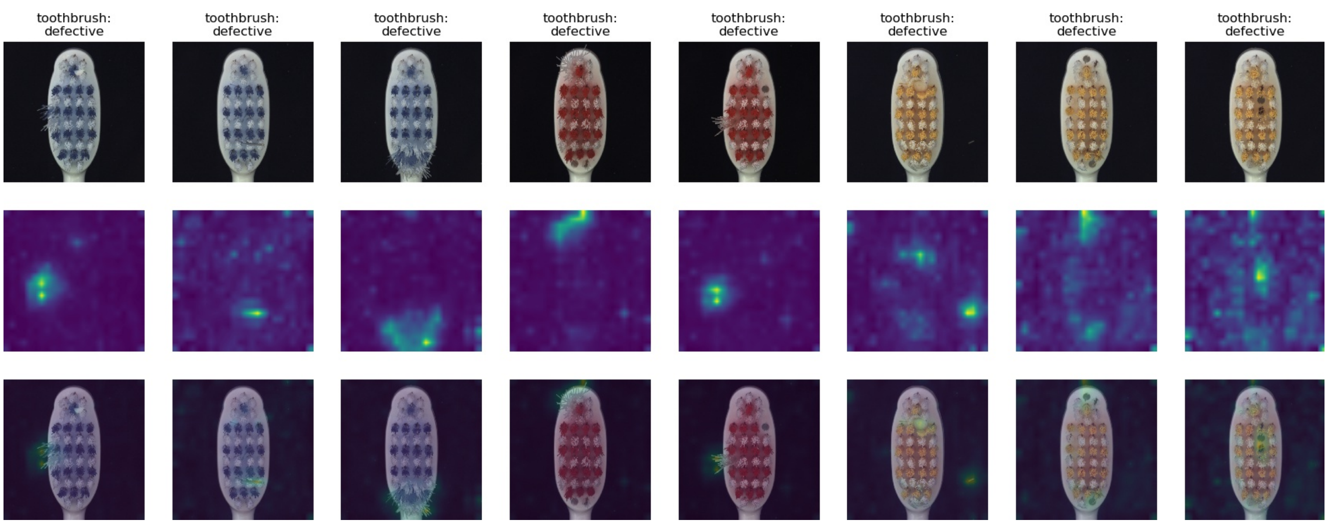

Figure 44.

References

[1] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9592–9600, 2019.

[2] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.