REASONING -BASED QUESTIONS (Write-up)

Q1: Choosing the Right Approach

I would use **object detection**, as the task involves identifying whether a label is present or missing on visually similar products. A detection model will allow us to identify the presence as well as locate the position of the label using bounding boxes, which is more precise than a simple classification model. Classification alone may not be sufficient because the products are visually similar, and we need to check a small part of them. If a detection model does not perform well, I would use segmentation, which will be able to outline the exact label area for better verification in cases where labels are small.

Q2: Debugging a Poorly Performing Model

I would first check for any sort of data quality issues, such as differences between the training images and the new factory images. Then, I would visualize the predictions and misclassifications to see if there are any common error patterns. Next, I'd check for any class imbalance in the training data and confirm that the annotations are correct. Finally, I'd test if the model is overfitting by evaluating the training and validation accuracy. Data augmentation and fine-tuning with real factory images can be done to obtain better results.

Q3: Accuracy vs Real Risk

No, the accuracy is not the right metric for this case because it can be misleading when the defects are rare. I would rather use precision, recall, and F1-score for this case. I'd also give special attention to recall, as in this case, missing a defective product is costly. A model with high accuracy but low recall could still allow defective products to go undetected; hence, monitoring the false negatives will be the most important task in this case.

Q4: Annotation Edge Case

Blurry or partially visible images should generally be kept in the dataset as they reflect real-world scenarios, and therefore will help the model to handle imperfect data. However, keeping these images may reduce the model's confidence. Removing them could make the model perform better on clear images but fail in real conditions. The trade-off here would be between the model's robustness and the model's numerical accuracy.