You are tasked with identifying whether a product is missing its label on an assembly line. The products are visually similar except for the label.

Q: Would you use classification, detection, or segmentation? Why? What would be your fallback if the first approach doesn't work?

Ans. It is a task that has to be performed in a controlled environment where it is assumed that the products on an assembly line follow a pattern so position and the camera angle can be fixed.

As the scope for randomness in placing of product is minimized , The usage of classification,detection and segmentation works.
I would go for object detection as It looks at the expected location to check whether the label is present or not and it is also robust if there are minor dispacements in the product placement which is inevitable.

The fallback approach would be dependent upon the need for size of model and budget for training the model. If there are constraints in these aspects I would use classification technique to find the labels as the randomness in position of product is minimized and if budget allows and size of model is not a hindrance segmentation model would be my fallback approach.

Q2: Debugging a Poorly Performing Model

You trained a model on 1000 images, but it performs poorly on new images from the factory.

Ans : The root cause analysis for such a situation could lie in various reasons.
The first step would be to gather information starting from the factory floor such as the color gradient in reality and in the data that has been trained on ,If there lies much difference in it , One has to get capture dataset that aligns with the reality on floor.
Check whether the dataset that has been trained on comprises of classes to be detected in equal share approximately and there is no class imbalance
Check whether the annotation of data is well done or not by examining few samples of annotated data or not and get the images that has been failed and examine them.
Check whether the training data,valid,data,test data were clearly divided and data transformations applied for the randomness.

If model is overfitting we reduce the complexity the model by reducing layers or removing dropout and if the model is underfitting we try to increase the learning rate and increase epochs and add the complexity of model by adding more layers or filters,

Q3: Accuracy vs Real Risk

Your model has 98% accuracy but still misses 1 out of 10 defective products.

Q: Is accuracy the right metric in this case? What would you look at instead and why?

Ans - In this case we are facing a problem with False Negatives which could be measured by recall metrics.
In this case there is a presence of clear class imbalance.
When failing to identify a defective product has more cost associated with it. In this case we rely on recall score which captures the False Negatives

Q4: Annotation Edge Cases

You’re labeling data, but many images contain blurry or partially visible objects.

Q: Should these be kept in the dataset? Why or why not? What trade-offs are you considering?

Ans: It depends on the use case of our model, if the model when deployed is supposed to come across these kind of scenarios where blurry and partially visible objects is normal, it is better to leave them in the dataset and annotate them accordingly.
However if that is not the case it is better to remove them in the annotation phase.