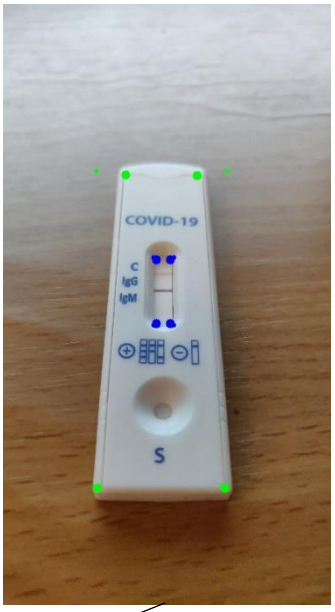
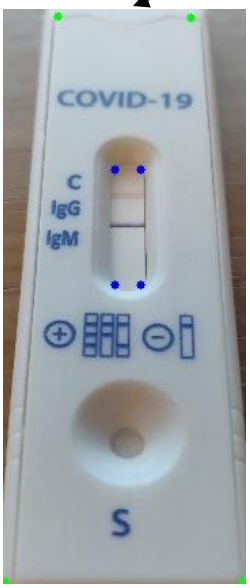
Computer Vision Pipeline for Membrane Detection (BTNx)

Input Image: This is one of the most difficult examples because of the vertical skew.

Masks from Mask RCNN: We obtain two masks i) kit and ii) membrane from Mask RCNN. As opposed to the bounding box coordinates (xmin, ymin, xmax, ymax), the masks give us true segmentation and hence curvatures and better adaptability to differing kit and membrane shapes across different images and test kits. Using the mask, note that we can also compute the rotation angle using the blue and green triangles. In case blue == green, we have a vertical skew (e.g. as we have here), otherwise we have a horizontal skew.

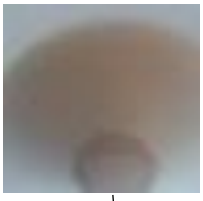


Coordinates from Mask: We obtain the classical bounding box coordinates (SMALL) and also the more natural lefttop, righttop, leftbottom, rightbottom coordinates (BIG) for both kits (GREEN) and membranes (BLUE).



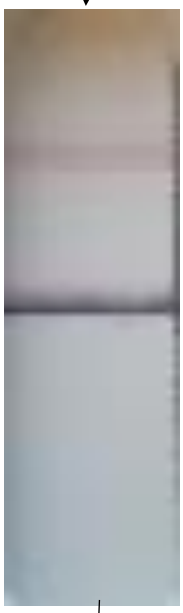
Close-Up Kit: We use the classical bounding box coordinates to get to the following close-up shot of the kit. Now, we can calculate a homography matrix between these classical coordinates (NOT SHOWN) and natural coordinates (GREEN) of the kit.

Homograph-Corrected Close-Up Kit: We warp the kit via the calculated homography matrix to get the correct, straight-up perspective. Now we enforce a membrane localization constraint. We only move on if the mask-calculated natural coordinates (GREEN) cover approximately the same space as the expected coordinates (RED) of the membrane which is calculated thanks to pre-measured real-life values.



Inset: Using the same homography-corrected close-up kit, we can again use a set of pre-measured real-life values to localize the inset.

Red-thresholded Inset: Using lower & upper values for red in HSV range, we construct a mask and apply the threshold. Here, the inset doesn't contain any blood and hence no red pixels. We would stop here and return an error message. For the purpose of the demo, let's assume we proceed.



Rotated and Close-up Membrane: Note that we had the membrane coordinates before all of the constraint checking operations below, in fact with the output of the Mask RCNN. The final membrane we pass on to classification, however, is actually the **rotated membrane using the calculated angle in the case in the first step.**

CLASSIFICATION