Defect Detection Pipeline - SEM

Assumptions

We assume that the inspected image and reference image are similar enough that they can be aligned, although ***perfect*** alignment is not required.

In this suggested pipeline, recall performance is prioritized over time complexity. Consequently, each alignment is optimized for the number of matches used, even though this significantly increases runtime.

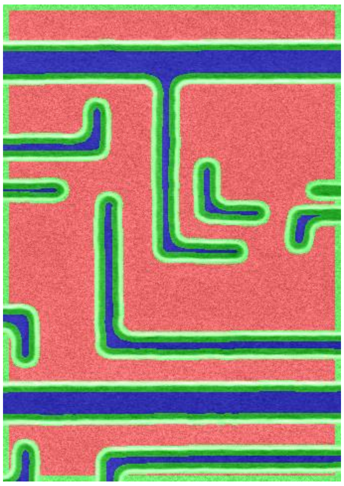
To improve defect capture, I suggest three different methods for segmentation (proposing defects), each one has its drawbacks and advantages in capturing different types of defects. The outputs of these methods are followed by a final classification step to determine which proposed defect really is a defect.

Method 1 – Contour Masks Difference

How it works:
Very simple – The two images (inspected + reference) are segmented to dark and bright regions by contouring the preprocessed images. The contours are filled to create masks of each region and the masks are aligned and subtracted (absdiff). If there's a defect on the edges, the misalignment will be seen in this mask difference. The mask difference goes through erosion and dilation, and the result is a mask of the proposed defects.
Advantages – can capture the challenging defects on the edges.
Disadvantages – can't capture defects that have gradual change.

Method 2 – Range of Values (ROV) Analysis

In a short study (found in the attached Jupyter notebook) I analyzed the ROV of each region in the reference image. The main conclusion was that for each given inspected image, we can only use the statistics of its reference image. We would also have to equalize the inspected image histogram to avoid mismatch between the images' intensities due to different lighting, for example.



Separating Regions for Statistics Estimation: Red – Bright, Blue – Dark, Green - Edges

How it works:
We are given an equalized inspected image and its reference. We calculate the statistics (μ, σ) of the reference, and segment the inspected image to dark and bright regions. The inspected image is blurred and dilated, and in each of its regions we search for all pixels that are $1.3 \times \sigma$ higher or lower from μ. A pixel is then proposed as defect if it has more than 4 neighbors that satisfy that in its 5x5 neighborhood (to reduce noise). That way we get defect masks from both bright and dark regions, and combine them with morphological operations.

Advantage: This method is able to catch the small, detailed defects that don't necessarily have high volume (for example, the middle narrow defect in case 2).

Disadvantage: Since we work with the pixel statistics of each region (bright/ dark) separately, it does a bad job on the when applied on the edges where pixels have high variance. To avoid it we have to remove a big range of pixels around the edges contour.

Method 3 – Local Background Analysis

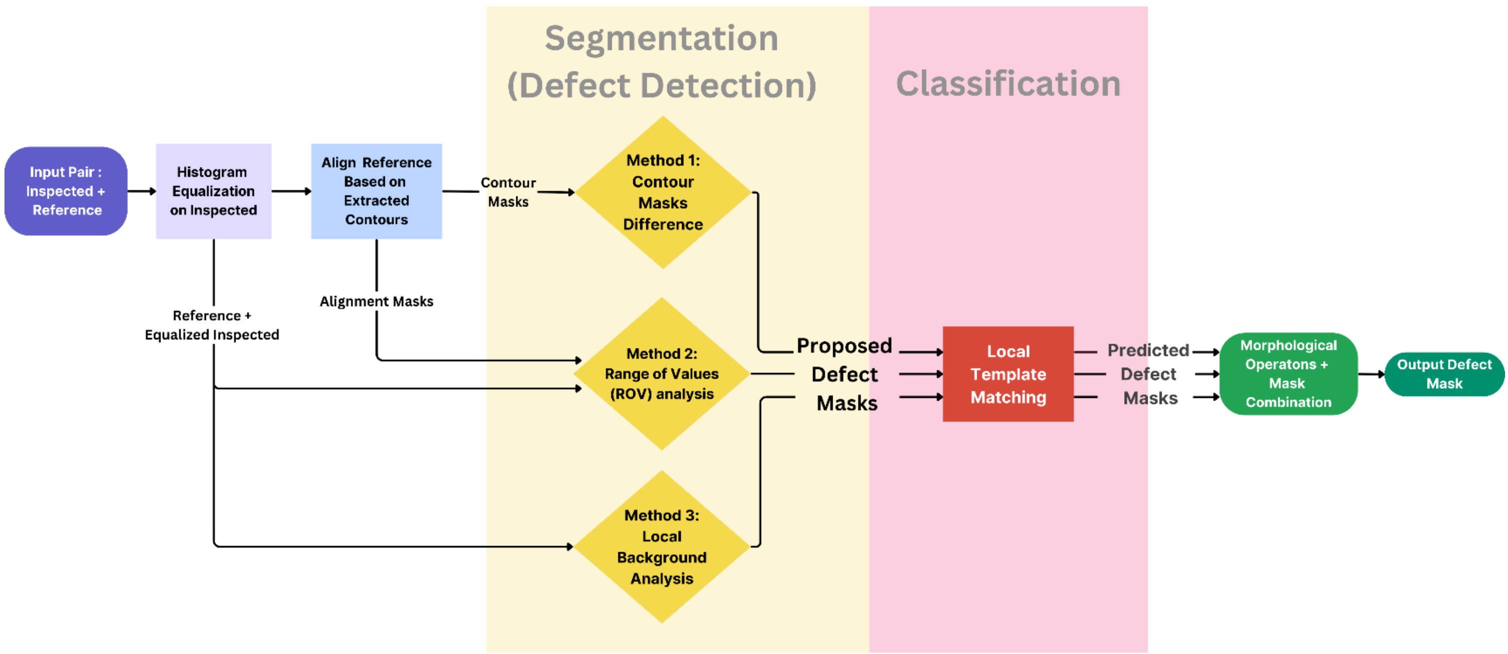
How it works:
We align the reference and subtract it from the inspected image to get the difference image. Then, we calculate the mean and standard deviation (μ, σ) of this difference image, and apply threshold to keep pixels that are 1.5 standard deviations above the mean. Contouring the result, we end up with filled contours that are suspected of being defects. Along each contour, we want to compare the mean intensity in the inspected image to the intensity of its local background. We define the local background as the subtraction of the inspected image from its dilation (the pixels that are bright here are the pixels just around an object). For each suspected contour we locally threshold its local background with OTSU's method and get a mask of pixels that are the local background of the suspected pixels. Finaly, we compare the mean intensity of the inspected image along the contour and along its local background, and segment a contour as defect only if it is 10% brighter or darker than its local background. We end with some morphological operations.

Advantage – can capture defects with gradual variation

Classification

Given a mask of suspected defects, I use local template matching to classify which proposed defect really is a defect. This approach is based on the assumption that the images are almost perfectly aligned. If we have a small region in the inspected image with suspected defect, we can simply perform template matching on its near area in the reference image. We would expect a high score (>0.80) if the template appears in the reference, and a lower score (<0.80) if there is a defect that obscures the real values there. We note that for a suspected region that is too large and might contain other suspected regions, regular template matching might be problematic, and we therefore use masked template matching.

The entire pipeline is visualized in the following flow chart:

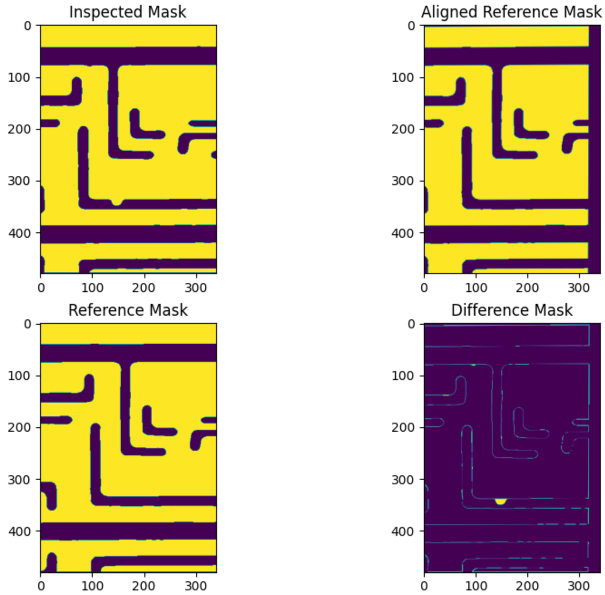


Results

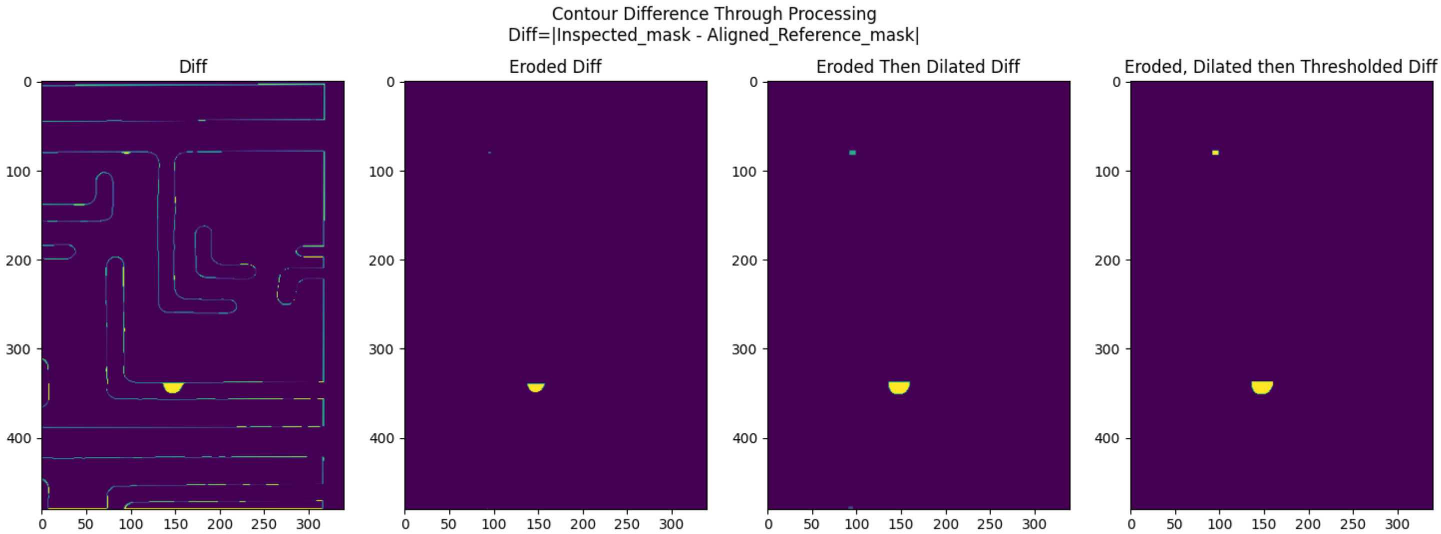
For each case we present the proposed defects by each method, and the final output by the local template matching classifier.

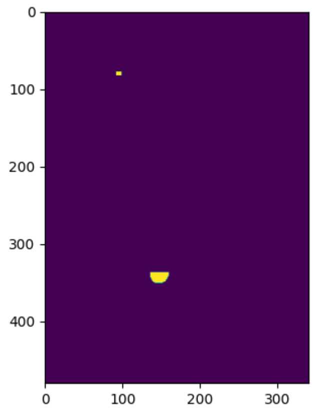
Case 1:

<u>Method1 (Contour Masks Difference):</u>



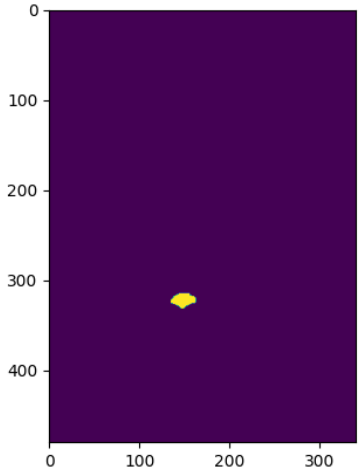
Aligned contour
masks + difference





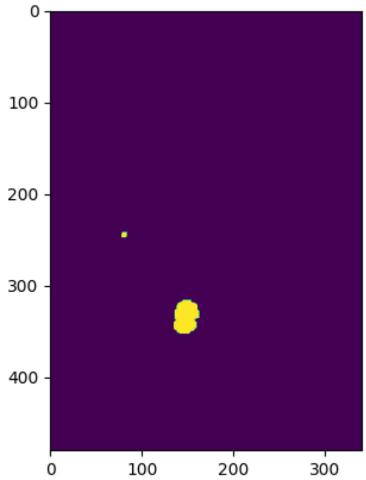
**Method 1 - Proposed defects
after morphological operations**

Method2 (ROV analysis):



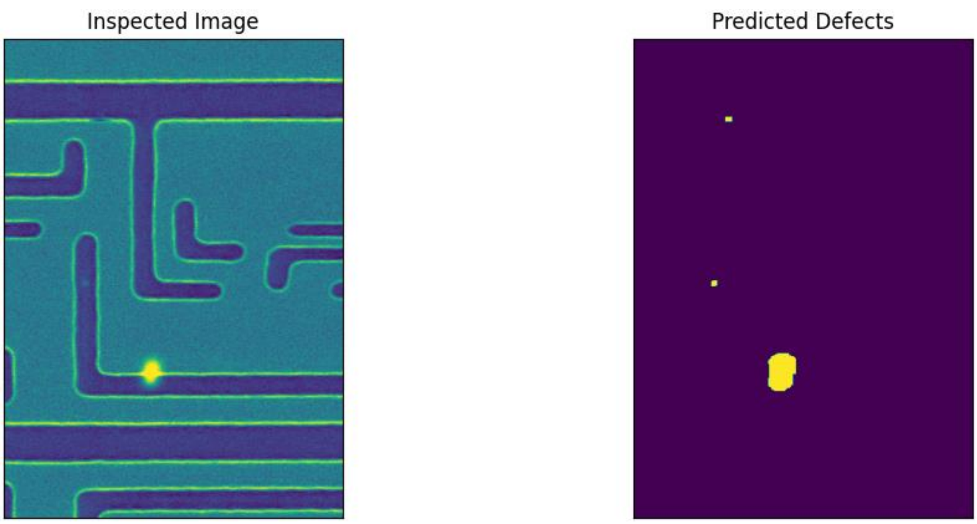
Method 2 - Proposed defects after morphological operations

Method 3 (Local Background Analysis)



Method 3 - Proposed defects after morphological operations

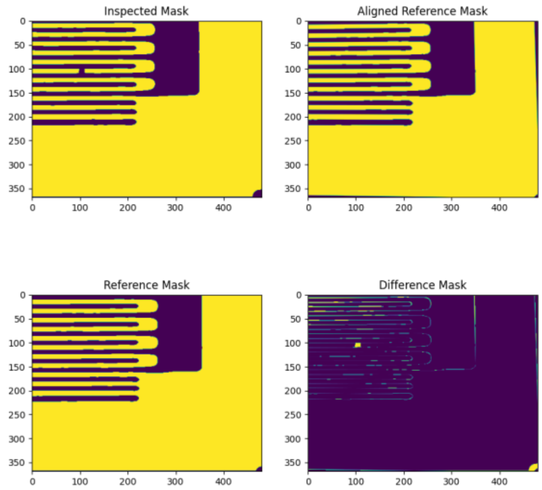
Local Template Matching Classification Result:



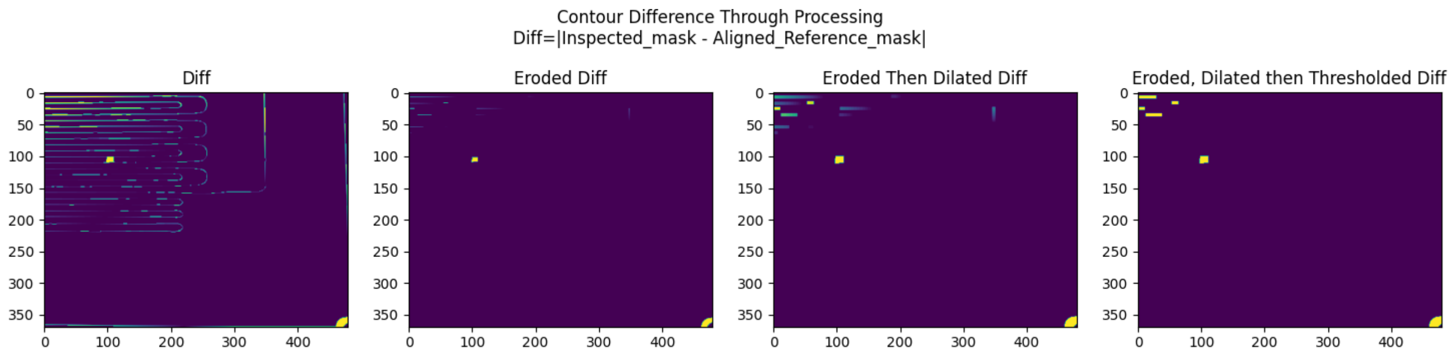
Case 1: Predicted Binary Defect Mask

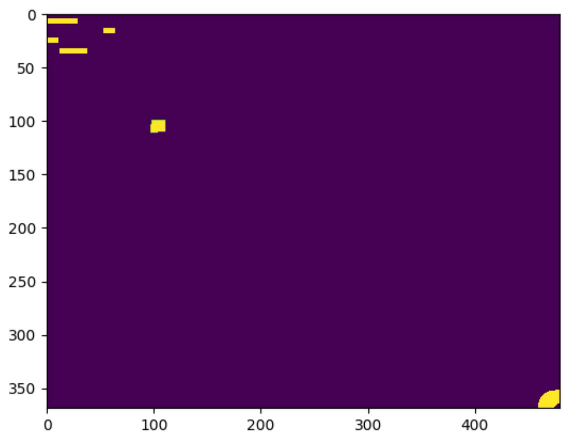
Case 2:

Method1 (Contour Masks Difference):



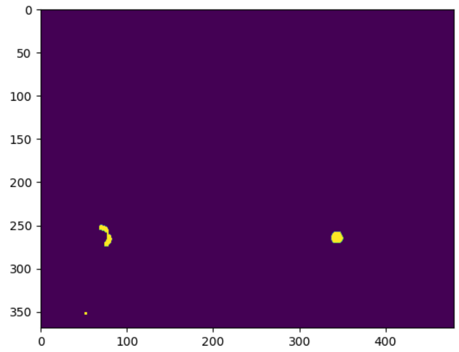
Aligned contour masks + difference





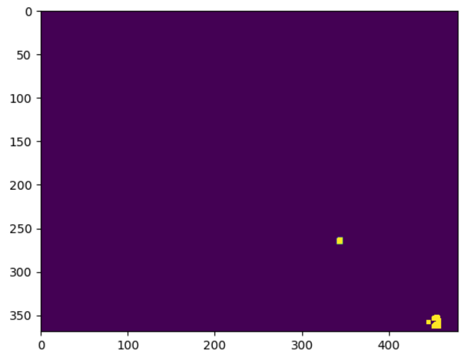
Method 1 - Proposed defects after morphological operations

Method 2 (ROV analysis):



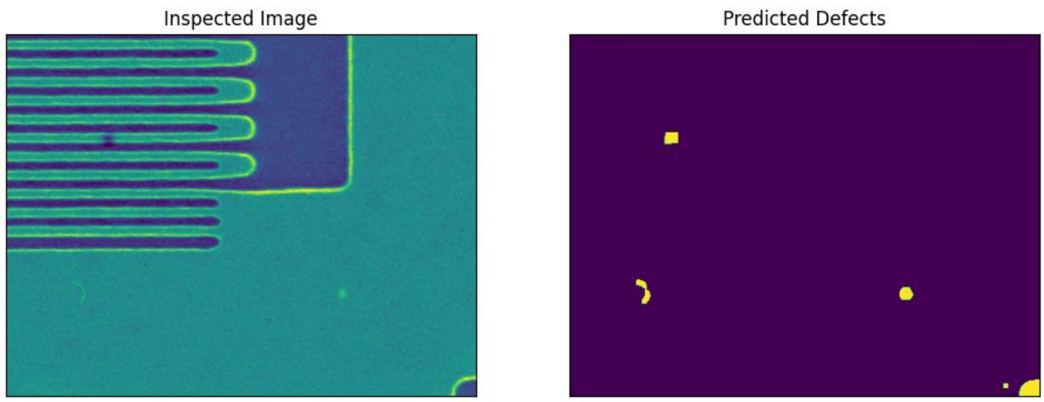
Method 2 - Proposed defects after morphological operations

Method 3 (Local Background Analysis)



Method 3 - Proposed defects after morphological operations

Local Template Matching Classification Result:

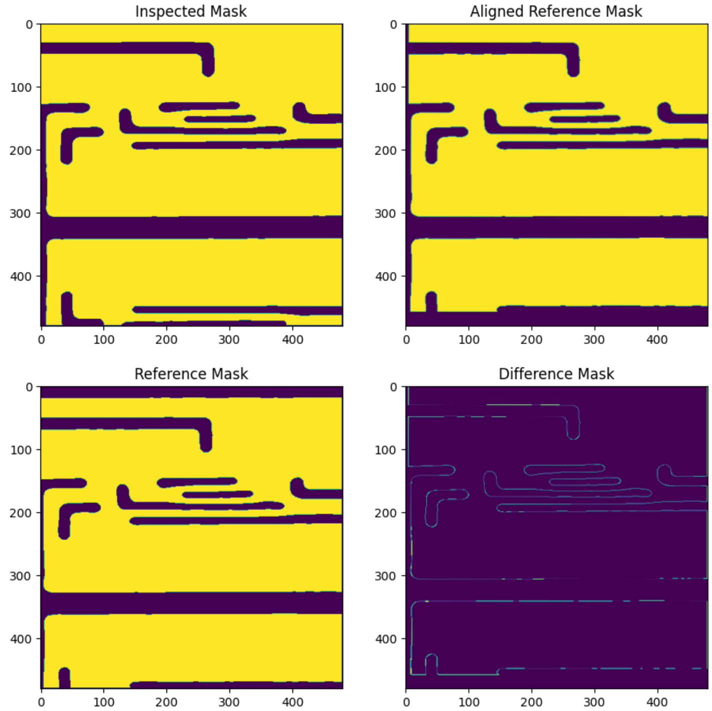


Case 2: Predicted Binary Defect Mask

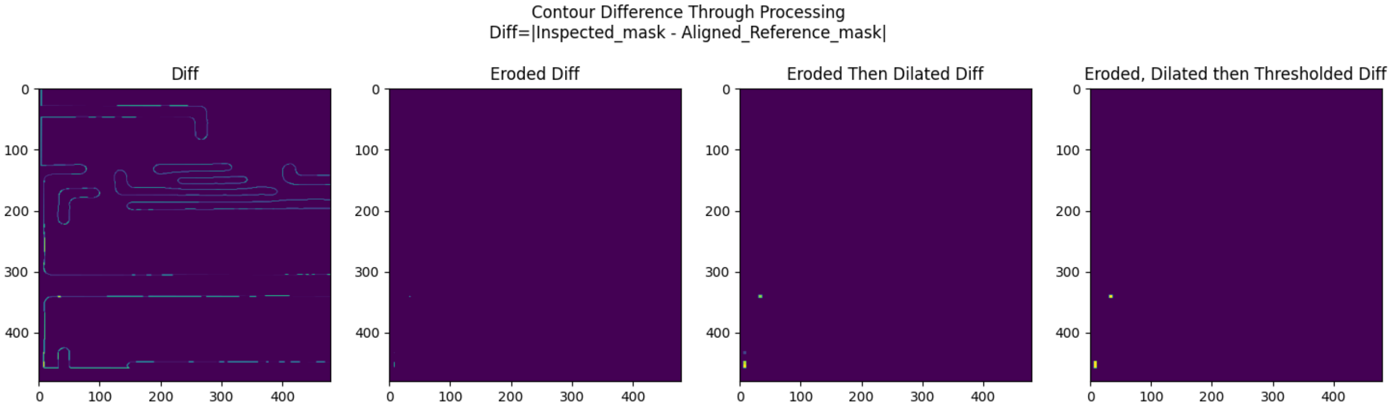
As evident, we captured the three main defects, but we also caught two false positives in the right corner.

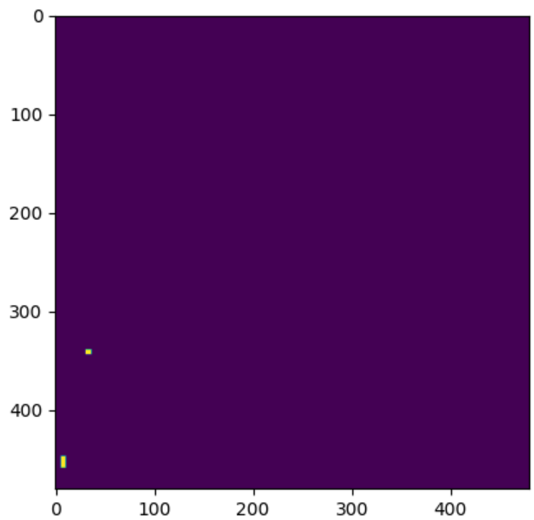
Case 3:

Method1 (Contour Masks Difference):



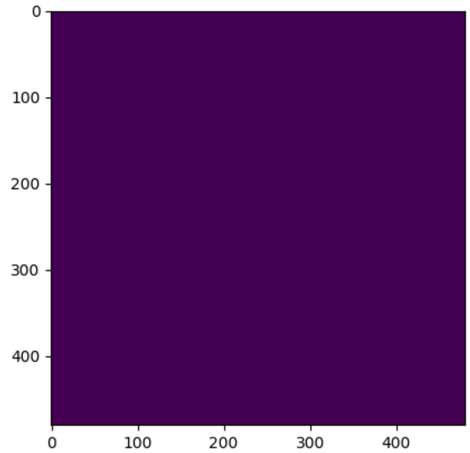
Aligned contour masks + difference





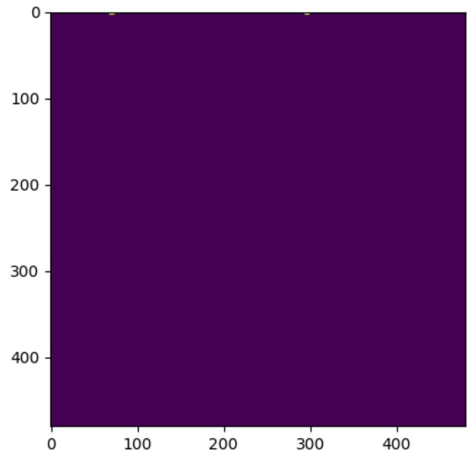
Method 1 - Proposed defects after morphological operations

Method 2 (ROV analysis):



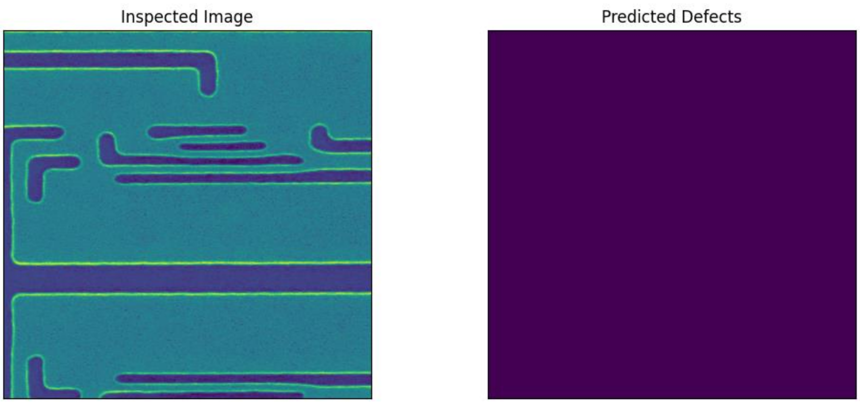
Method 2 - Proposed defects after morphological operations

Method 3 (Local Background Analysis)



Method 3 - Proposed defects after morphological operations

Local Template Matching Classification Result:



Case 3: Predicted Binary Defect Mask