Algorithm Development for Defect Detection in Silicon Wafer Production

Omri Drori

Abstract

This document outlines a home task for developing an algorithm to detect defects in silicon wafers. The task's objective, challenges, and a two-pronged method of approach are described, highlighting the importance of precision and the difficulties encountered due to limited sample sizes and noise in image data.

1 Introduction

The objective of this task was to develop an algorithm capable of identifying defects in silicon wafers during their production process. Such an algorithm has significant practical implications in industrial manufacturing, offering the potential to automate defect detection, thereby enhancing efficiency and reducing the reliance on manual inspection.

2 Challenges and Initial Approaches

A primary challenge encountered was the limited number of sample images available for analysis. This constraint hindered the application of advanced deep learning techniques typically used for detection and segmentation tasks. An initial approach involved training a neural network to analyze the distribution of image patches in both reference and test images, aiming to identify deviations indicative of defects. While this method showed promise, it faced a significant hurdle due to the high noise levels in the images and the minuscule size of some defects. Excessive noise reduction risked missing these small defects, leading to potential false negatives. figure 1 show the unsuccessful results on case 2.

3 Alternative Strategies

Subsequently, another approach was explored: predicting pixel values in the inspected image using patches from the reference image. However, this method also faced challenges due to the precision required and the presence of noise in the images.

4 Alignment and Its Challenges

A fundamental and straightforward idea involved aligning the reference and inspected images, where the differences between them could indicate defects. This process, however, was impeded by difficulties in accurately aligning the images. Despite numerous attempts to identify significant features for alignment, the results were not sufficiently accurate, leading to many false positives. The similarity of features within the images caused challenges in feature matching, and adjustments in parameters like Lowe's ratio test did not yield satisfactory results. This problem can be seen in image 2 An assumption of Affine transformation (no perspective distortion) improved results but did not fully resolve the alignment issues.

5 Two-Pronged Approach to Defect Detection

Given the imperfect alignment, it was necessary to adopt a two-part strategy for defect detection:

5.1 Defects Within Wafer Components

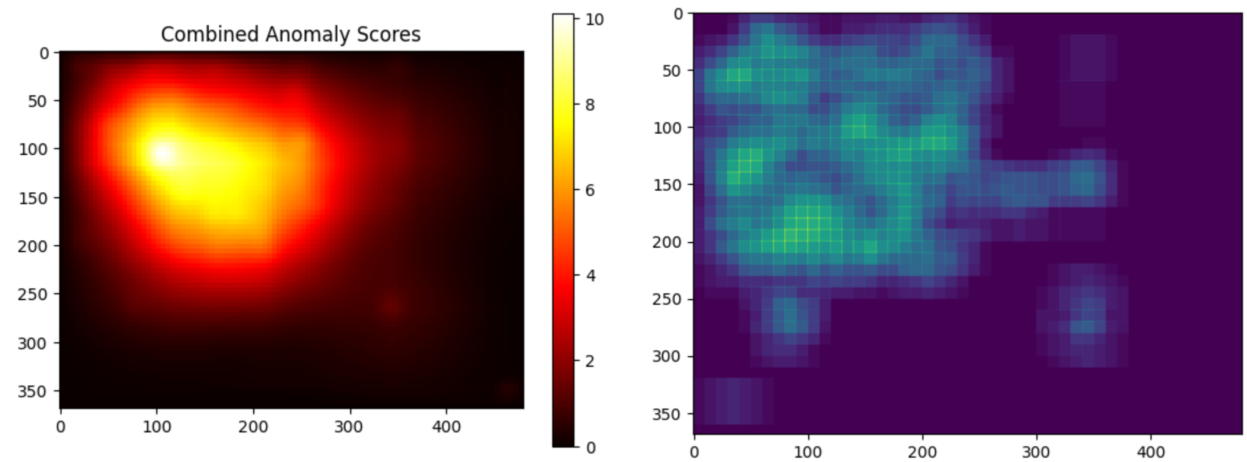
This part was more challenging, involving noise reduction through non-local means to preserve potential subtle defects. Otsu thresholding followed by morphological operations helped in subtracting residual noises and creating a mask to minimize noise in subsequent steps.

5.2 Defects Outside Wafer Components

With lesser noise interference, simpler anomaly detection algorithms could be applied. Several options were implemented, including the isolation forest, mean variation algorithm, and one-class SVM. The choice of algorithm depended on the specific characteristics of the sample images.

6 Results

The figures presented below illustrate the outcomes of the defect detection task. While the algorithm successfully identified defects in the provided samples to a certain extent, it is important to note that this success was heavily reliant on fine-tuning specific parameters through an iterative process of trial and error. Consequently, the customized nature of these parameters, particularly those related to morphological operations and kernel sizes aimed at noise reduction, may limit the algorithm's applicability to new, unseen samples. The methodology, especially when applied to areas within the components, depended largely on empirical adjustments rather than a standardized approach, which suggests that the current configuration might not be as effective on different datasets.



(a) First autoencoder result

(b) Second autoencoder result

Figure 1: Results of fitting autoencoders to patches from case2 example. It is evident that the noise adversely affects performance, as seen in both plots.

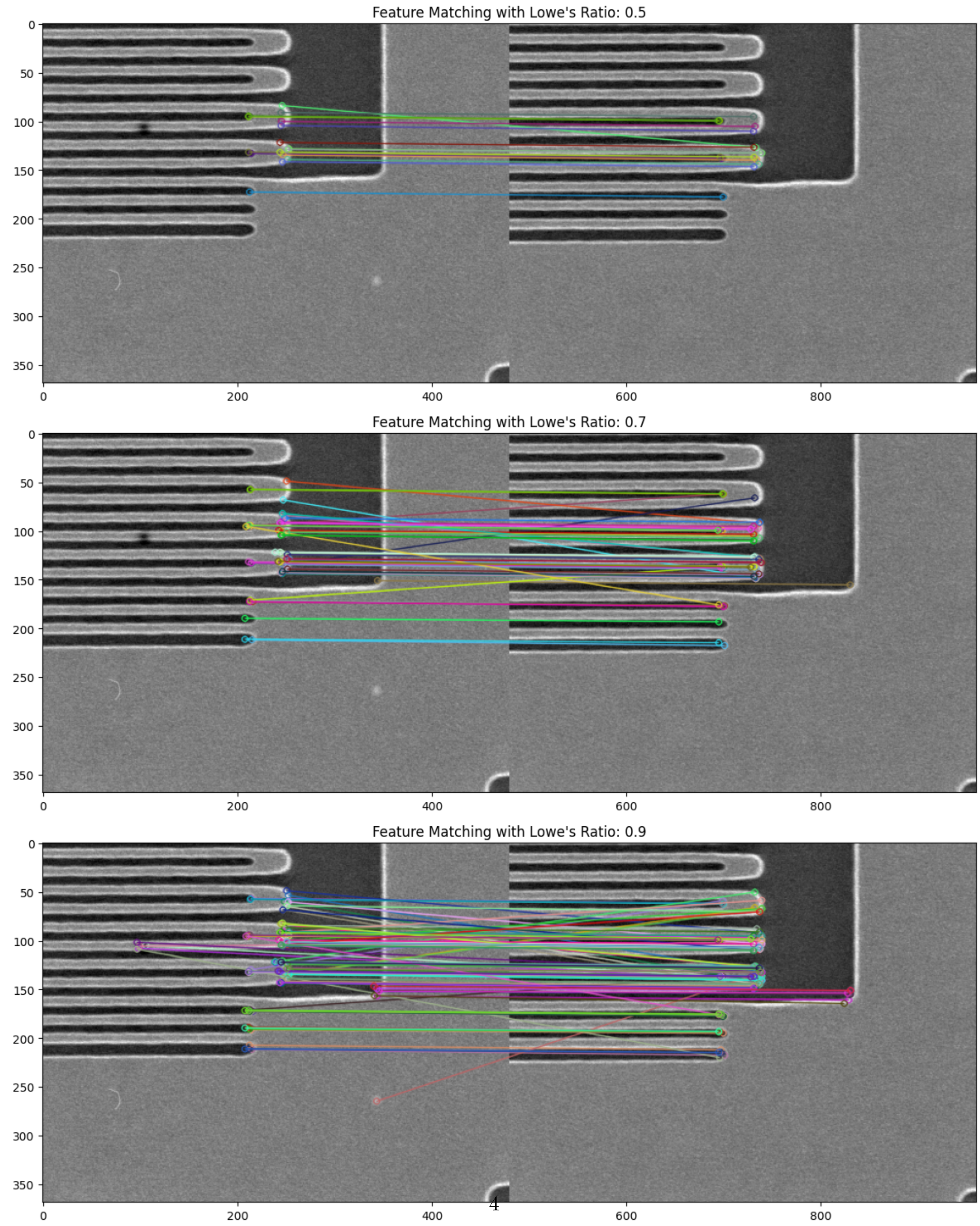
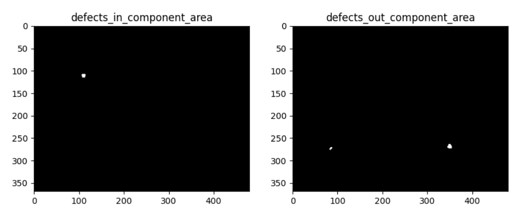
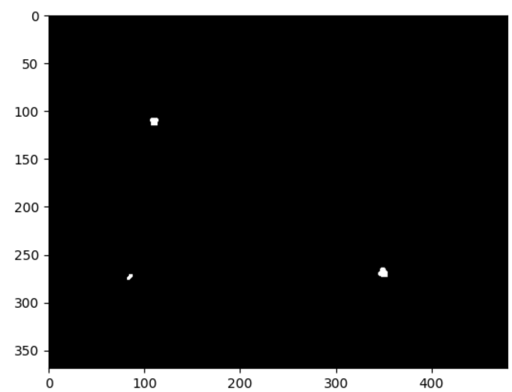


Figure 2: Impact of Lowe's ratio on feature alignment. A lower ratio results in fewer features being matched, which might be too restrictive for reliable alignment. A higher ratio, on the other hand, can lead to many ambiguous matches, making the alignment prone to errors.

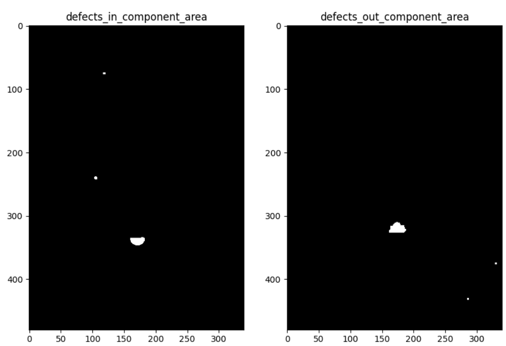


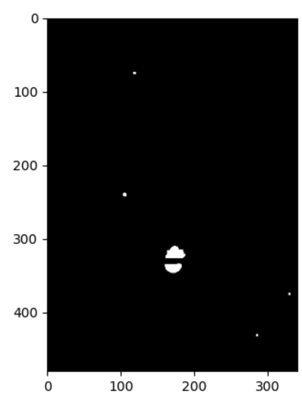


(b) final result on case1

(a) The intermediate result from each operation on case2 between and outside components area

Figure 3: Results of case 2 .

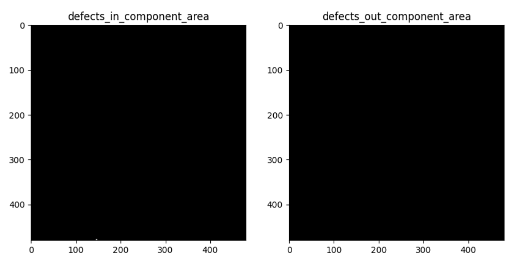


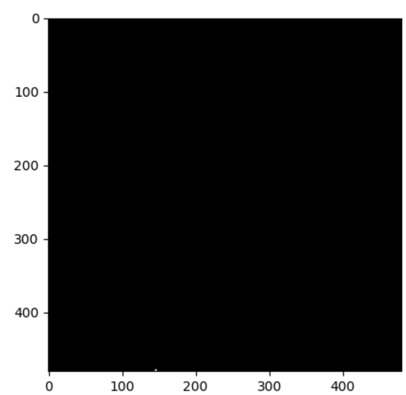


(b) final result on case1

(a) The intermediate result from each operation on case1 between and outsiide components area

Figure 4: Results of case 1 . it can be seen that results not prefect.





(b) final result on case1

(a) The intermediate result from each operation on non-defective case between and outsiide components area

Figure 5: Results of non defectve case