DEFECTS IDENTIFICATION ON SEMICONDUCTOR WAFER FOR YIELD IMPROVEMENT USING MACHINE LEARNING

PEDRAM TABATABAEEMOSHIRI

FACULTY OF ENGINEERING
UNIVERSITI MALAYA
KUALA LUMPUR

2025

DEFECTS IDENTIFICATION ON SEMICONDUCTOR WAFER FOR YIELD IMPROVEMENT USING MACHINE LEARNING

PEDRAM TABATABAEEMOSHIRI

DISSERTATION SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF ENGINEERING SCIENCE

FACULTY OF ENGINEERING
UNIVERSITI MALAYA
KUALA LUMPUR

2025

UNIVERSITI MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: **PEDRAM TABATABAEEMOSHIRI**

Matric No: **S2029802/1**

Name of Degree: **Master of Engineering Science**

Title of Project Paper/Research Report/Dissertation/Thesis ("this Work"):

Defects Identification on Semiconductor Wafer for Yield Improvement Using Machine Learning

Field of Study: **Electronics and Automation**

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the Universiti Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date: 5/2/2025

Subscribed and solemnly declared before,

Witness's Signature Date: 5/2/2025

Name:

Designation:

DEFECTS IDENTIFICATION ON SEMICONDUCTOR WAFER FOR YIELD IMPROVEMENT USING MACHINE LEARNING

ABSTRACT

The semiconductor industry underpins modern technology, with its products embedded in almost every electronic device. As semiconductor devices grow increasingly intricate, ensuring their quality and reliability becomes more challenging. Electrical testing is crucial to semiconductor wafer quality assurance, designed primarily to identify fabrication defects. However, the testing process itself can inadvertently introduce new defects that may go undetected by subsequent inspection methods such as manual and visual inspection. When these defects escape detection, defective wafers may reach customers, leading to rejection and return to the manufacturer, resulting in significant yield losses and operational inefficiencies. This study addresses the urgent issue of detecting hidden defects in semiconductor wafers that conventional methods overlook. This work presents a novel graph-based semi-supervised learning (GSSL) algorithm designed for wafer defect detection. The proposed methodology involves collecting wafer inspection data, extracting relevant features, and applying the GSSL algorithm to identify the hidden defects. The approach constructs a graph representation of the wafer, leveraging its physical layout and test configuration, and integrates domain-specific knowledge. The method uses weighted edges to represent the likelihood of defect propagation between dies, optimized through extensive experimentation, followed by an iterative label propagation process to uncover hidden defects. Experimental results demonstrate the effectiveness of our method, achieving a 68% accuracy in detecting hidden defects across multiple product categories in real-world semiconductor manufacturing environments. The algorithm showed consistent performance across different wafer types and test configurations, outperforming traditional detection methods

with improved computational efficiency. This study offers valuable insights into the semiconductor industry, providing an advanced tool to enhance yield management and quality control processes.

Keywords: Semiconductor, Wafer Testing, Defect Identification, Test-Induced Defects, Graph-Based Semi-Supervised Learning

PENGENALPASTIAN KECACATAN PADA WAFER SEMIKONDUKTOR UNTUK PENINGKATAN HASIL MENGGUNAKAN PEMBELAJARAN MESIN

ABSTRAK

Industri semikonduktor merupakan tunjang teknologi moden, dengan produknya yang terkandung dalam hampir semua peranti elektronik. Seiring dengan peningkatan kerumitan peranti semikonduktor, usaha memastikan kualiti dan kebolehpercayaan menjadi semakin mencabar. Pengujian elektrik merupakan komponen penting dalam jaminan kualiti wafer semikonduktor yang direka terutamanya untuk mengenal pasti kecacatan fabrikasi. Walau bagaimanapun, proses pengujian itu sendiri boleh secara tidak sengaja menghasilkan kecacatan baharu yang mungkin tidak dapat dikesan oleh kaedah pemeriksaan seterusnya seperti pemeriksaan manual dan visual. Apabila kecacatan ini terlepas daripada pengesanan, wafer yang cacat mungkin sampai kepada pelanggan, menyebabkan penolakan dan pemulangan kepada pengilang, mengakibatkan kerugian hasil dan ketidakcekapan operasi yang ketara. Kajian ini menangani isu mendesak dalam pengesanan kecacatan tersembunyi pada wafer semikonduktor yang tidak dapat dikesan oleh kaedah konvensional. Kerja ini membentangkan algoritma pembelajaran separa terselia berasaskan graf (GSSL) baharu yang direka khusus untuk pengesanan kecacatan wafer. Metodologi yang dicadangkan melibatkan pengumpulan data pemeriksaan wafer, pengekstrakan ciri-ciri berkaitan, dan penggunaan algoritma GSSL untuk mengenal pasti kecacatan tersembunyi. Pendekatan ini membina perwakilan graf wafer, memanfaatkan susun atur fizikal dan konfigurasi ujian, serta mengintegrasikan pengetahuan domain khusus. Kaedah ini menggunakan sisi berpemberat untuk mewakili kebarangkalian perambatan kecacatan antara dai, yang dioptimumkan melalui eksperimen menyeluruh, diikuti dengan proses perambatan label berulang untuk mendedahkan kecacatan tersembunyi. Keputusan eksperimen menunjukkan keberkesanan kaedah kami, mencapai ketepatan 68% dalam mengesan kecacatan tersembunyi merentasi pelbagai kategori

produk dalam persekitaran pembuatan semikonduktor sebenar. Algoritma ini menunjukkan prestasi yang konsisten merentasi pelbagai jenis wafer dan konfigurasi ujian, mengatasi kaedah pengesanan tradisional dengan kecekapan pengkomputeran yang lebih baik. Kajian ini menawarkan pandangan berharga kepada industri semikonduktor, menyediakan alat termaju untuk meningkatkan pengurusan hasil dan proses kawalan kualiti.

Keywords: Semikonduktor, Pengujian Wafer, Pengenalpastian Kecacatan, Kecacatan Aruhan-Ujian, Pembelajaran Separa Terselia Berasaskan Graf.

ACKNOWLEDGEMENTS

First and foremost, I would like to extend my sincere appreciation to my supervisor, Associate Prof. Dr. Narendra Kumar for an invaluable experience to work under his mentorship. His vast knowledge and experience in industry were instrumental in guiding my research. Additionally, I am thankful for the opportunity he facilitated to collaborate with NXP Semiconductors Malaysia, which significantly enriched my research experience.

I would also like to express my gratitude to my co-supervisor, Associate Prof. Dr. Anis Salwa Mohd Khairuddin, for her consistent guidance throughout my studies. Her expertise and knowledge in Machine Learning, allowed me to refine my research for optimal results. I am especially grateful for her support in guiding me through the publication process.

My sincere thanks go to NXP Semiconductors Malaysia, for their role as an industrial collaborator. I would like to send my sincere regards to all the employees of NXP Semiconductors Malaysia, especially the Probe department, for their warm welcome and assistance. I would like to thank Mr. Daniel Ting and Mr. Vivek Regeev for their guidance and support on this project. Their industrial perspective significantly enhanced this work, enabling a real data simulation. I also wish to acknowledge Siao Hui, Praveen, Adi, Karimah, Jessica, Guna, Yugan, Suhaila, Azlan, and other members of NXP Semiconductor Malaysia who made my experience much more enjoyable.

Last but not least, I would like to thank my family for providing daily support, motivation, and guidance through my studies and life. I am sure that I would not be here today to finish my studies if it wasn't for their sacrifice and motivation.

To all those mentioned and to those whose names I may have inadvertently omitted, I offer my sincere thanks. Your contributions have been instrumental in the successful completion of this work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

Abbreviations:

AOI	:	Automated Optical Inspection
CNN	:	Convolutional Neural Network
CLIQUE	:	Clustering In Quest
DBSCAN	:	Density-Based Spatial Clustering of Applications with Noise
DUT	:	Device Under Test
FM	:	Foreign Material
GBILI	:	Graph Based on Informativeness of Labelled Instances
GRF	:	Gaussian Random Fields and Harmonic Functions
GNN	:	Graph Neural Network
GSSL	:	Graph-based Semi-Supervised Learning
IC	:	Integrated Circuit
IoT	:	Internet of Things
kNN	:	k-Nearest Neighbors
LGC	:	Local and Global Consistency
LNP	:	Linear Neighborhood Propagation
ML	:	Machine Learning
OPTICS	:	Ordering Points To Identify the Clustering Structure
PC	:	Personal Category
RGCLI	:	Robust Graph that Considers Labelled Instances
SLP	:	Special Label Propagation
SOM	:	Self-Organizing Map
SPC	:	Statistical Process Control
SSL	:	Semi Supervised Learning

SVM	:	Support Vector Machine
VLSI	:	Very Large-Scale Integration

Symbols:

A	:	Area of interest within the clustering parameters
c_i	:	Class of node i
E	:	Set of edges
e_{ij}	:	Edge that connects nodes v_i and v_j
D	:	Set of defect locations
dut_k	:	Site-number associated with the k^{th} position in each stepping
dut_k^*	:	Site-number with the largest number of collisions with defects
G	:	Graph
$isCollising_{j,k}$	:	Collision status of k^{th} site-number coordinates in j^{th} stepping with defect locations
N	:	Total number of dies
N_D	:	Total number of defects
N_S	:	Total number of steppings
$\mathcal{N}(i)$	:	Set of neighbours of node i
M	:	Number of sites tested in each stepping
PC-#	:	Product Category
R_{dut_k}	:	List to store defect locations that collide with the k^{th} site-number
$R_{\mathrm{dut}_k^*}$	:	Longest list of defect locations for k^{th} site-number
S	:	Set of stepping coordinates
T_{dut_k}	:	List to store stepping coordinates and their collision status for the site-number
V	:	Set of nodes
v_i	:	Node i in the graph
v_j	:	Node j in the graph

w_D	:	Weightage of defective die
w_G	:	Weightage of good die
w_{ij}	:	Weight of edge that connects nodes v_i and v_j
w_P	:	Weightage of potentially defect die
x_{AOI}	:	x-coordinate of wafer map generated by AOI
x_{AOI_New}	:	x-coordinate of wafer map generated by AOI after Rotation and Translation
$x_{\text{AOI\_Rotated}}$	:	x-coordinate of wafer map generated by AOI after Rotation
x_{AOI_TR}	:	Top right x-coordinate of wafer map generated by AOI after Rotation
x_{Defect}	:	x-coordinate of any defect in the wafer map generated by AOI
x_{Defect_New}	:	x-coordinate of any defect in the wafer map generated by AOI after Rotation and Translation
$x_{Defect\_Rotated}$	:	x-coordinate of any defect in the wafer map generated by AOI after Rotation
x_i (3.6)	:	x-coordinate of the i^{th} defect
x_i (3.7)	:	Feature vector associated with node v_i
$x_{j,k}$	:	x-coordinate of the k^{th} site in j^{th} stepping
x_{max}	:	Maximum x boundary of the area of interest
$x_{\text{max\_cluster}}$	:	Maximum x boundary of the cluster
x_{min}	:	Minimum x boundary of the area of interest
$x_{\text{min\_cluster}}$	:	Minimum x boundary of the cluster
$x_{\text{TestData\_TR}}$	:	Top right x-coordinate of wafer map generated by Physical Test Data
y_{AOI}	:	y-coordinate of wafer map generated by AOI

y_{AOI_New}	:	y-coordinate of wafer map generated by AOI after Rotation and Translation
$y_{AOI\_Rotated}$	:	y-coordinate of wafer map generated by AOI after Rotation
y_{AOI_TR}	:	Top right y-coordinate of wafer map generated by AOI after Rotation
y_{Defect}	:	y-coordinate of any defect in the wafer map generated by AOI
y_{Defect_New}	:	y-coordinate of any defect in the wafer map generated by AOI after Rotation and Translation
$y_{Defect\_Rotated}$	:	y-coordinate of any defect in the wafer map generated by AOI after Rotation
y_i (3.6)	:	y-coordinate of the i^{th} defect
y_i (3.7)	:	Label associated with node v_i
$y_i^{(t+1)}$	:	Updated label of node i at iteration $t+1$
$y_j^{(t)}$	:	Label of the neighbouring node j at iteration t
$y_{j,k}$	:	y-coordinate of the k^{th} site in j^{th} stepping
$y_{\max}$	:	Maximum y boundary of the area of interest
$y_{\max\_cluster}$	:	Maximum y boundary of the cluster
y_{min}	:	Minimum y boundary of the area of interest
$y_{min\_cluster}$	:	Minimum y boundary of the cluster
$y_{\text{TestData\_TR}}$	:	Top-right y-coordinate of the wafer map in Physical Test Data
ϵ	:	Predefined threshold that determines convergence status
Δx	:	x component of the translation vector
Δy	:	y component of the translation vector
θ_{AOI}	:	Direction angle for the AOI Machine
θ_{rotation}	:	Rotation degree between directions of AOI and physical test data

$\theta_{TestData}$: Direction angle for the physical test data

CHAPTER 1: INTRODUCTION

1.1 Introduction

The semiconductor industry has been the driving force behind the remarkable technological advancements witnessed in recent decades. Semiconductor devices, forming the bedrock of modern electronics, are essential components in a vast array of technologies. From the ubiquitous smartphones in our pockets to the powerful computer systems that drive artificial intelligence and scientific discovery, these devices rely on the intricate patterning of microscopic features on a silicon wafer substrate. This miniaturization process, though revolutionary, presents a complex challenge: ensuring that these features are formed perfectly at ever-shrinking scales.

In the semiconductor industry, wafers are known as one of the most valuable assets to manage. In recent years, the ever-growing demands of faster development and availability to market with the advent of high-end integrated systems and Internet-of-Things (IoT) make a compelling case for semiconductor manufacturers to produce wafers of high product quality by yield enhancement and cost reduction. The fabrication of very large-scale integration (VLSI) circuits is very complicated, with many manufacturing steps executed on the same wafer, hence it is critical for wafers to be tested and inspected with the highest accuracy to identify the failing dies before the packaging process. Otherwise, there could be defective semiconductor dies that will go through the assembly process and therefore lead to unnecessary expenses at the end of the manufacturing process. Monitoring the defects ensures manufacturers that the assets are managed in an efficient manner to increase profit.

Traditional methods for identifying defects on semiconductor wafers have relied heavily on human inspection. Whether the probing and identification are done by machine

and verified by humans or entirely performed by human experts in the field, is not only time-consuming and labour-intensive but also susceptible to human error and inconsistencies. Moreover, the complexity of modern integrated circuits and the ever-decreasing feature sizes have rendered manual inspection increasingly challenging and inefficient. In this context, Machine Learning (ML) techniques have emerged as a promising solution, offering the potential to automate the defect detection process while improving accuracy and throughput.

The application of machine learning algorithms to defect identification on semiconductor wafers has garnered considerable attention from researchers and industry professionals alike. By leveraging the powerful pattern recognition and classification capabilities of these algorithms, it becomes possible to detect and classify defects with unprecedented precision. This not only enhances the overall yield but also facilitates root cause analysis, enabling manufacturers to identify and address the underlying sources of defects, thereby improving the overall manufacturing process.

1.2 Motivation

While the test process is designed to find the defects and avoid unnecessary expenses at the end of the manufacturing process, the test process itself may induce defects to good dies, which reduces manufacturing yield. Furthermore, they lead to the test economics problem, as it usually takes a lot of effort from process engineers and test engineers to find the problem. Besides, if test-induced defects cannot be properly identified, the accuracy of root-cause analysis will be adversely affected. Therefore, it is desirable to detect this problem as soon as possible.

With the introduction of Machine Learning and their ability to analyse the wafer maps more accurately than humans, manufacturers are able to identify faulty test equipment faster and easier. However, the current methods and technologies are being used for the known defects detected by the wafer test and advanced optical inspection machines, while the limitation of named detectors on detecting the defects, leads to having hidden defects on the wafers.

Amidst the pursuit of robust solutions for identifying elusive and hidden defects on semiconductor wafers, the emerging field of semi-supervised graph-based machine learning techniques presents a promising avenue for exploration. Traditional supervised learning methods, while effective for known defect patterns, often falter when confronted with the complexities and nuances of previously unseen or rare defect manifestations. Conversely, purely unsupervised approaches may lack the guidance necessary to discern meaningful patterns from the vast expanse of data inherent to wafer maps.

Semi-supervised graph-based algorithms offer a compelling middle ground, harnessing the strengths of both paradigms. By leveraging the representational power of graphs to capture the intricate relationships and dependencies within wafer data, these techniques can effectively propagate label information from a limited set of labelled instances to a broader collection of unlabelled data points. This synergistic approach not only capitalizes on existing knowledge but also enables the discovery of novel defect signatures, facilitating the identification of previously undetected anomalies.

1.3 Problem Statement

In the semiconductor industry, maintaining high yield is crucial for profitability and competitiveness. However, defects consistently pose a significant threat to maintaining a high yield. Wafer testing is designed to identify and mitigate these defects to ensure the functionality of semiconductor devices. However, paradoxically, the testing process itself can introduce defects. These test-induced defects often go undetected by conventional methods, leading to substantial yield losses and financial burdens. While recent advancements have been made in detecting visible test-induced defects (Cheng et al., 2021), a critical gap remains in identifying these hidden defects, which can have profound implications for device reliability and overall yield. Manual inspection, a traditional method for detecting defects, heavily relies on the expertise and vigilance of the inspector. However, this approach is prone to human error and subjectivity, making it difficult to identify hidden defects that require recognizing intricate patterns and subtle anomalies. Machine learning techniques have been applied to optical inspection systems to automate the defect detection process. However, these techniques face challenges in adapting to the constantly changing contrast of hidden defects, making it arduous to develop a robust detection algorithm that can handle these unique and rare cases.

To address these limitations, graph-based semi-supervised learning (GSSL) algorithms have emerged as a promising approach for detecting hidden defects. However, current GSSL algorithms (Song et al., 2023) have their own limitations when applied to this specific problem. Constructing an appropriate graph that effectively captures the complex relationships between visible and hidden defects is a critical step in GSSL. Moreover, the

label propagation process in GSSL may not always accurately assign labels to hidden defects due to their unique nature and potential dissimilarity to visible defects.

There is, therefore, a pressing need for innovative approaches that can effectively uncover these hidden defects by leveraging both the spatial and topological information inherent in wafer maps. Such methods must be capable of operating with limited labelled data, as extensive labelling in semiconductor manufacturing is often costly and time-consuming.

1.4 Objectives

The objectives of the proposed research are as follows:

i. To investigate the intricate relationship between defect occurrence and the configuration of testing methodologies to unveil underlying patterns of hidden defects.
ii. To develop a machine learning-based method that demonstrates at least 60% average accuracy in predicting the precise location of hidden defects on semiconductor wafers.
iii. To optimize the graph-based algorithm for improved accuracy and reduced time complexity in hidden defect localization.

1.5 Thesis Contribution

This thesis endeavours to make significant contributions to the field of semiconductor manufacturing by pioneering novel machine learning techniques that address the critical challenge of detecting test-induced defects on wafers. Through a collaborative effort with NXP Semiconductor Malaysia, a leading global manufacturer of integrated circuits, this

research has been driven by real-world industrial needs and informed by domain expertise.

The outcomes of this work hold the promise of delivering substantial impact, both in academic and practical domains. From a theoretical standpoint, the development of advanced semi-supervised graph-based algorithms for defect identification will contribute to the expanding body of knowledge in machine learning and its applications in semiconductor yield enhancement. Concurrently, the proposed methodologies have the potential to translate into tangible benefits for the semiconductor industry, empowering manufacturers to overcome the limitations of conventional detection methods and unlock new levels of yield improvement.

By leveraging the synergy between cutting-edge research and industrial collaboration, this thesis aims to bridge the gap between academic exploration and real-world implementation. The resulting techniques, validated through rigorous experimentation and evaluation on industrial datasets provided by NXP Semiconductor Malaysia, will offer a robust and scalable solution for identifying elusive test-induced defects, thereby mitigating yield losses, streamlining root cause analysis, and ultimately enhancing the overall efficiency and profitability of semiconductor manufacturing operations.

1.6 Thesis Outline

This dissertation is framed into five chapters as follows:

Chapter 1 provides an overview of the research background, states the problem statement and main focus of the research, and points out the research objectives.

Chapter 2 gives a detailed background on the wafer test, the procedures and limitations. This chapter provides a technical review of the conventional methods of defect identification both for fabrication-induced and test-induced defects. The literature review on the available machine learning methods, especially on the semi-supervised graph-based algorithms is covered in this chapter.

Chapter 3 describes the flow of designing a semi-supervised graph-based algorithm to identify the location of the defect on the Wafer Map. This includes the procedure of detection of test-induced defects on the wafer through the analysis of defects. Moreover, the root-causing method of test-induced defects is described here.

Chapter 4 presents the development, validation, and characterization of a novel semi-supervised graph-based algorithm, along with the simulation and experimental results achieved. Various types of semiconductor wafers were examined under different testing configurations, with algorithm parameters systematically adjusted to determine the most optimized combination. The effectiveness of the developed algorithm is further assessed by comparing its defect identification results with actual evaluations conducted by NXP engineers. Additionally, a comparative analysis between the proposed approach and the existing reported literature is provided to highlight its advancements and contributions.

Chapter 5 represents the conclusion of this research work, summarizes the findings, contributions and design challenges. The potential improvement of the proposed algorithm is highlighted for future work.

CHAPTER 2: LITERATURE REVIEW

2.1 Overview

2.1.1 Importance of defect identification in semiconductor manufacturing

The semiconductor industry is a vital cog in the global economy, producing the essential components that power everything from smartphones to supercomputers. As technology advances and devices become more sophisticated, the demand for high-performance semiconductors continues to skyrocket. To meet this demand, manufacturers are constantly pushing the boundaries of miniaturization and complexity, making the identification and localization of defects on semiconductor wafers more challenging than ever before.

Defects in semiconductor wafers can have catastrophic consequences. Even a minuscule flaw can lead to a device failure, rendering it unusable. This can result in significant financial losses for manufacturers, as scrapped wafers and rework costs can quickly add up. In addition, product quality can be compromised, undermining the reputation of the manufacturer and leading to lost market share.

Given the high stakes involved, there is a pressing need for robust and accurate defect detection techniques. These techniques must be able to identify and localize defects with high precision, even on the most advanced semiconductor wafers. Failure to do so can have a devastating impact on the bottom line of semiconductor manufacturers.

In addition to the financial implications, defects in semiconductor wafers can also have a negative impact on the environment. The manufacturing process for semiconductors is complex and resource-intensive, and any defects that result in scrapped wafers or rework

can lead to wasted materials and energy. By developing more efficient defect detection techniques, manufacturers can reduce their environmental footprint and contribute to a more sustainable future.

The development of cutting-edge defect identification methodologies is an ongoing pursuit for semiconductor companies. By investing in research and development, manufacturers can stay ahead of the curve and maintain their competitive edge in the global market. The ability to identify and localize defects with high precision is essential for producing high-performance, reliable, and cost-effective electronic devices that meet the needs of consumers and businesses alike.

2.1.2 Overview of the semiconductor wafer production process

The semiconductor wafer manufacturing process is a meticulously orchestrated series of steps aimed at transforming a pure silicon crystal into a complex array of electronic circuits that can perform myriad functions. The process can be broadly categorized into three primary stages: fabrication, testing, and assembly. Each stage is crucial, employing sophisticated technology and stringent quality controls to ensure the functionality and reliability of the final products. Fabrication involves multiple processes such as photolithography, doping, and etching to create the intricate layers and circuit patterns on the silicon wafer. Testing, often referred to as probing, ensures each chip functions correctly and meets the desired specifications. Finally, assembly involves cutting the wafer into individual chips, packaging these chips into protective casings, and preparing them for shipment to customers. Figure 2.1 shows the top-view process-flow of semiconductor wafer manufacturing, based on (Srivastava, 2021).

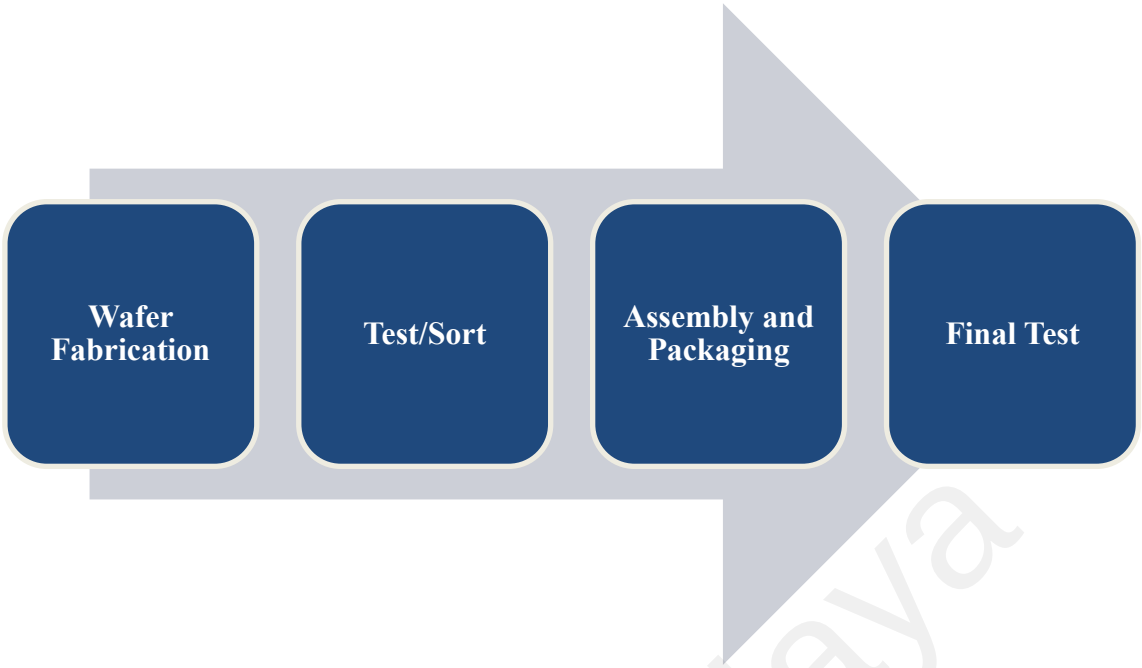


Figure 2.1: Process-flow of semiconductor wafer manufacturing (Srivastava, 2021)

As described by (Quirk & Serda, 2001) in their comprehensive book "Semiconductor Manufacturing Technology" the process typically begins with the growth of a cylindrical ingot of highly purified silicon crystal. The fundamental fabrication steps involve photolithography, where patterns are transferred onto the wafer surface using light-sensitive photoresist materials, followed by various deposition, etching, and implantation processes to create the desired device structures. These wafers undergo a series of critical steps, including polishing, cleaning, and epitaxial deposition, to prepare the surface for subsequent processing.

The fabrication process is the heart of semiconductor manufacturing, where the intricate device structures are created on the wafer surface. This stage involves a series of steps, such as deposition of various materials (e.g., silicon dioxide, polysilicon, metals), photolithographic patterning using light-sensitive photoresists, etching to selectively

remove materials, and ion implantation to introduce dopants. These steps are repeated multiple times in a specific sequence, with each layer building upon the previous one to form the complex three-dimensional structures of modern integrated circuits.

After fabrication, the individual die on the wafer undergoes wafer-level testing and probing, a critical step for identifying functional and non-functional circuits. This process involves electrically probing each die on the wafer using specialized equipment and testing protocols. The probing data is used to generate wafer maps, which indicate the location and distribution of defective die, allowing for yield assessment and enhancement.

The wafers are then singulated, separating the individual die through techniques like dicing or scribing. These individual dies are subsequently packaged, a process that involves encapsulating the die in protective materials and forming external connections, such as wire bonding or flip-chip assembly. The packaging stage is crucial for protecting the delicate die from environmental factors and facilitating integration into printed circuit boards or other electronic systems.

2.1.3 Wafer Test

Wafer testing is a crucial step in the semiconductor manufacturing process, ensuring the quality and functionality of the integrated circuits (ICs) before they are packaged and sent to customers. This crucial process unites electrical testing and optical inspection, meticulously scrutinizing each die on the wafer's surface to identify and isolate potential defects. Through this rigorous evaluation, wafer testing elevates the yield, reliability, and performance of the final products, ensuring that only the highest quality ICs reach the market.

2.1.3.1 Electric Test

The electrical test, also known as wafer probe or wafer sort, is performed using a probe machine, which consists of a prober and a tester. The prober is responsible for handling the wafer and aligning the probe card to the bond pads on each die, while the tester executes the test program and measures the electrical characteristics of the devices.

One crucial function of the prober is wafer handling. It securely holds the wafer during testing using a vacuum or electrostatic chuck. The prober can accommodate various wafer sizes, ranging from small to large diameters, to cater to different manufacturing processes. Automated wafer handling systems minimize human intervention and reduce contamination risks. Another vital function is probe card alignment. The prober precisely positions the probe card over the wafer's surface, ensuring accurate alignment with the bond pads on each die. This alignment is essential for proper electrical contact between the probes and the bond pads. Advanced probe systems employ high-resolution cameras and image processing algorithms to achieve sub-micron alignment accuracy. Once the probe card is aligned, the prober lowers it onto the wafer, establishing electrical contact between the probes and the bond pads. Probes are typically made of tungsten or gold-plated materials to ensure good conductivity and minimize wear. The prober applies a controlled amount of force to maintain consistent and reliable contact throughout the testing process. Figure 2.2 visualises a prober machine with how the electric tester is getting aligned on the wafer.

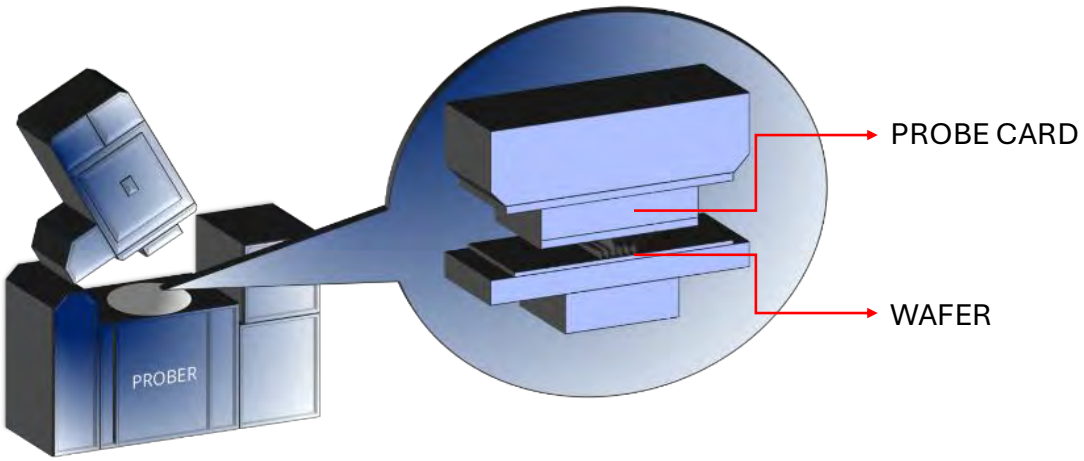


Figure 2.2: Prober machine.

The prober acts as an interface between the wafer and the electrical tester. It facilitates the transmission of test signals from the tester to the device under test (DUT) on the wafer. The prober also routes the electrical responses from the DUT back to the tester for analysis.

The prober enables die-by-die testing, allowing for the evaluation of individual devices on the wafer. It moves the probe card systematically across the wafer, testing each die sequentially. Defective dies can be identified and marked for further analysis or rework.

The prober collects electrical test data from each die and transmits it to the tester. The tester analyses the data to determine whether each device meets the specified electrical specifications. This collected data is crucial for statistical process control (SPC) and yield analysis.

By precisely controlling wafer handling, probe card alignment, and electrical contact, the prober plays a vital role in ensuring the accurate and reliable testing of wafer dies. It is an integral part of the wafer probe process, helping manufacturers identify and sort

functional devices from defective ones, ultimately contributing to the quality and yield of semiconductor products.

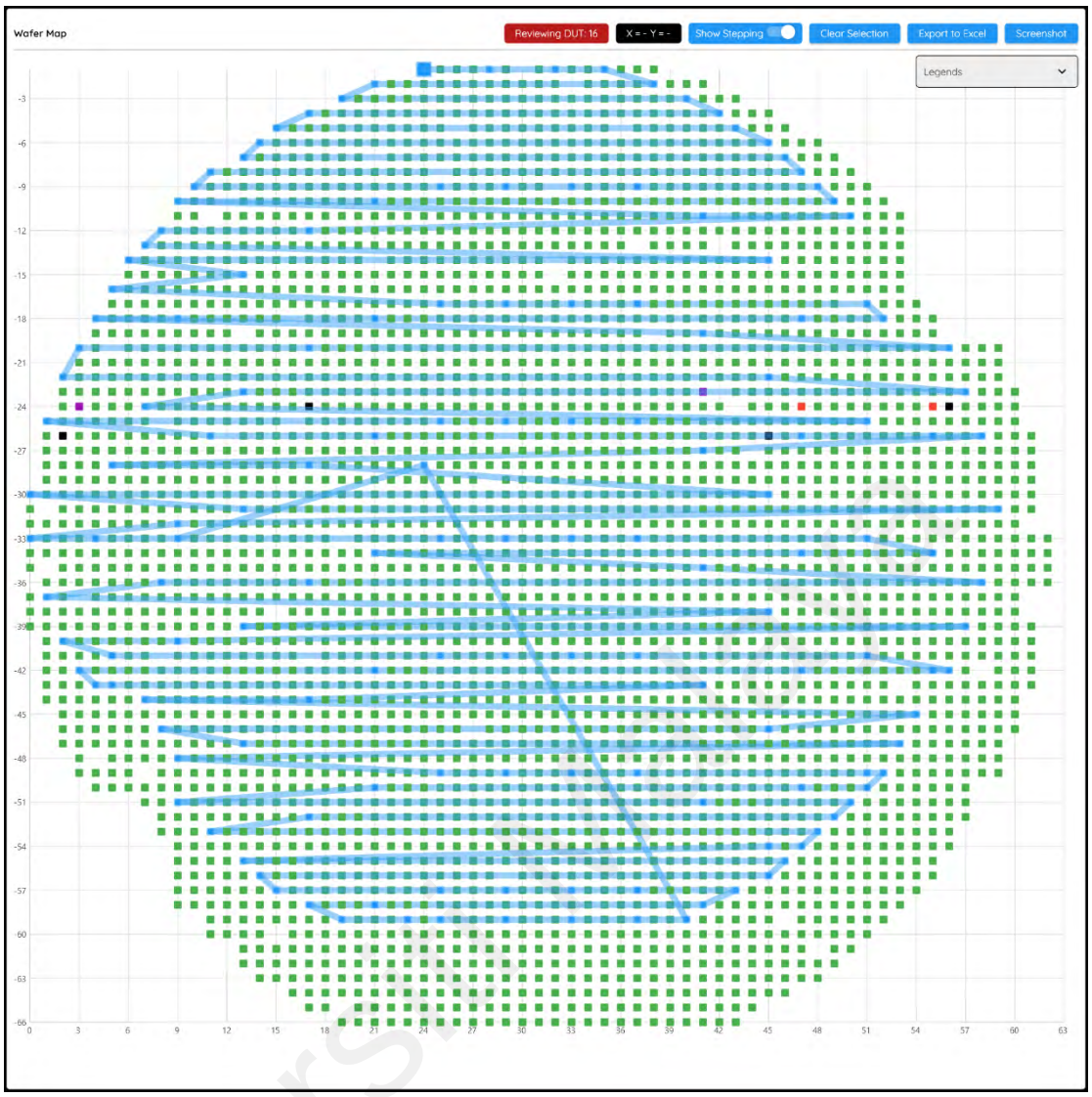
To increase testing efficiency, modern wafer test systems employ parallel testing, where multiple dies are tested simultaneously. The configuration of the electrical test includes the path and direction of movement for the prober, the number of dies being tested simultaneously, and the site number that identifies the index of the testing die. The prober moves in a predefined path, typically in a serpentine or zigzag pattern, to cover all the dies on the wafer. The number of dies tested at once depends on the probe card design and the tester's capabilities. Modern probe cards can have hundreds or even thousands of probes, allowing for parallel testing of multiple dies, which significantly reduces the overall test time.

Figure 2.3 (a) illustrates a typical wafer test configuration with multiple test sites. In this example, a probe card with 32 test sites is used to test a wafer. The test sites are arranged in a 4x8 matrix, allowing 32 dies to be tested concurrently. The test process follows a predetermined test flow, which defines the sequence of tests to be performed on each die, as shown by the blue line path in Figure 2.3 (b). The starting point of the testing process is marked by a prominent blue square at the top of the wafer. Each time the prober evaluates a group of dies, the initial die in the group is highlighted with a blue square. The blue line connecting these squares visually represents the test sequence, indicating the direction and path of the testing process.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32

(a) Site-number orientation of 32 sites

Figure 2.3: Test configuration example



(b) Example path of prober

Figure 2.3, continued: Test configuration example

The results of the electrical test are used to generate a wafer map, which visually represents the pass/fail status of each die on the wafer. This information is essential for yield analysis and process optimization.

2.1.3.2 Optical Inspection

Optical inspection techniques have emerged as a crucial tool for detecting and classifying defects in semiconductor wafer manufacturing. These methods have

significantly impacted the industry by enabling the identification of various defect types, leading to improved yield and quality control. The advent of machine vision and image processing technologies has revolutionized the way wafer inspection is performed, replacing manual inspection methods that were time-consuming, labour-intensive, and prone to human error (Ma et al., 2023). In their comprehensive review, (Ebayyeh & Mousavi, 2020) discussed the various automatic optical inspection (AOI) systems and algorithms used for detecting defects in electronic components such as semiconductor wafers, flat panel displays, printed circuit boards, and light-emitting diodes.

The history of optical inspection in semiconductor manufacturing dates back to the early days of the industry. Initially, manual inspection using microscopes was the primary method for detecting defects on wafer surfaces. However, as the complexity of integrated circuits increased and feature sizes decreased, manual inspection became increasingly challenging and inefficient. AOI systems have played a crucial role in identifying these defects, which can be categorized as random, systematic, or mixed-type defects (Ebayyeh & Mousavi, 2020). The ability to detect and classify these defects accurately is essential for yield enhancement and process optimization in semiconductor manufacturing.

AOI systems typically consist of an image acquisition unit, comprising cameras and illumination settings as used by (Hara et al., 1988) back in 1988, and an image processing unit that analyses the captured images using various algorithms. The selection of cameras and lenses depends on factors such as the field of view, resolution, and depth of field required for the specific application Illumination settings are also critical in AOI systems, as they can significantly impact the contrast and visibility of defects in the captured images. Figure 2.4 illustrates the normal setup of the camera and wafer for an AOI system.

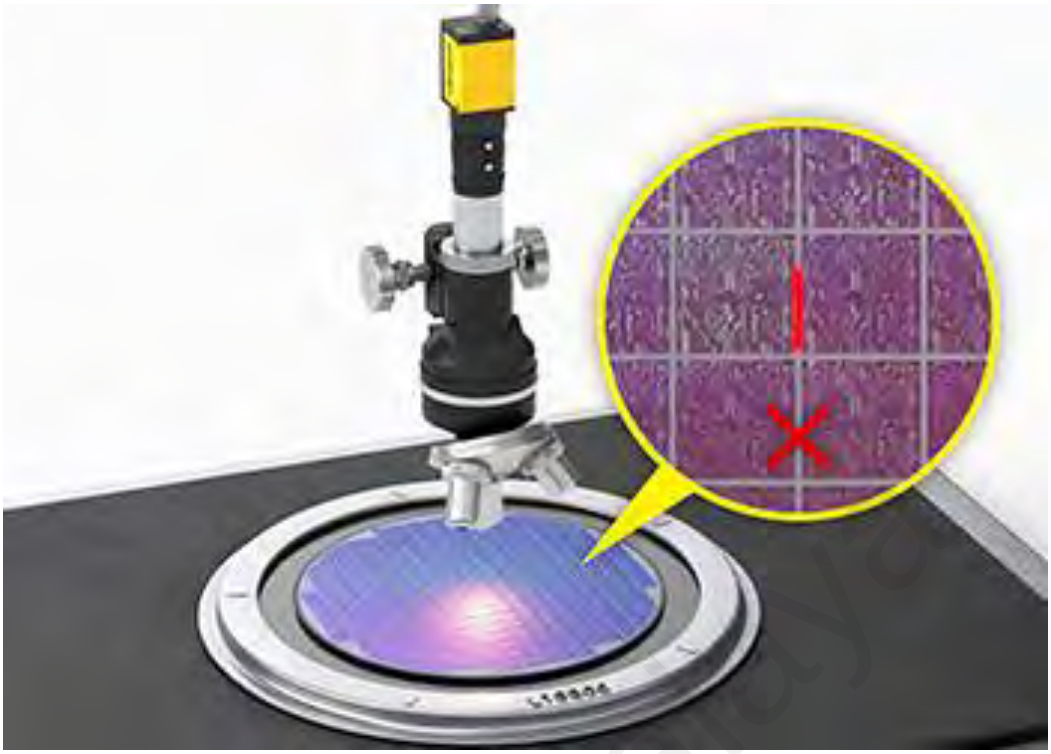


Figure 2.4: Semiconductor wafer defect inspection (Cognex, 2024)

The image processing unit of AOI systems employs a wide range of machine learning algorithms for defect detection and classification. These algorithms can be broadly categorized into supervised and unsupervised learning methods. Supervised learning algorithms, such as support vector machines (SVMs) and convolutional neural networks (CNNs), require labelled training data and are widely used for defect classification tasks. Unsupervised learning algorithms, such as k-means clustering and self-organizing maps (SOMs), do not require labelled data and are often used for anomaly detection and pattern recognition tasks in semiconductor wafer inspection.

One of the most famous works in the field of wafer surface defect detection is the WM-811K dataset, which consists of 811,457 wafer images with eight different defect types

(Ma et al., 2023). This dataset has been widely used as a benchmark for evaluating the performance of various defect detection and classification algorithms.

Recent advancements in deep learning have further enhanced the capabilities of AOI systems in semiconductor wafer inspection. For example, (Nakazawa & Kulkarni, 2018) proposed a CNN-based approach for wafer map defect pattern classification. In another study, (Saqlain et al., 2020) developed a deep CNN for wafer defect identification on an imbalanced dataset, demonstrating the potential of deep learning in handling real-world challenges in semiconductor manufacturing.

Despite the advancements in optical inspection techniques, there are still limitations and challenges. One major limitation is the difficulty in detecting sub-surface or non-visible defects, as optical methods rely on surface imaging. Additionally, the increasing complexity of wafer patterns and the need for higher-resolution imaging pose challenges in terms of computational resources and inspection time. Moreover, the presence of mixed-type defects and the need for a large amount of labelled data for training machine learning models are other obstacles that need to be addressed.

2.2 Types and sources of defects in semiconductor wafers

2.2.1 Defect in semiconductor wafers

In the realm of semiconductor manufacturing, a defect on a wafer refers to any deviation from the desired structure or composition that compromises the functionality, reliability, or yield of the semiconductor device. These defects can manifest in various forms, ranging from minute imperfections at the atomic scale to visible anomalies on the wafer surface. Common types of defects include point defects, dislocations, surface

roughness, particulates, and process-induced irregularities, each with its unique impact on device performance. Defects can arise from multiple stages of the fabrication process, including crystal growth, wafer preparation, deposition, etching, and lithography. They can also occur during testing, where physical contact with the wafer surface may inadvertently damage delicate structures or introduce contamination. Understanding and mitigating defects are paramount in semiconductor manufacturing, as even minor deviations can lead to significant yield losses and compromise the functionality of electronic devices.

2.2.2 Fabrication Defects

This category encompasses defects arising from various fabrication processes, including lithography, deposition, and etching. Lithographic defects, such as mask defects, exposure issues, or resist residues, can lead to pattern distortions, line edge roughness, or incomplete pattern transfer. Deposition defects, like non-uniform thickness, pinholes, or voids, can impact device performance and reliability. Etching defects, caused by improper etching conditions or etch residues, can result in issues like undercut, trenching, or incomplete pattern transfer. (Shinde et al., 2022). The defects in this category originate from a variety of fabrication processes, each posing unique challenges.

As semiconductor technology advances and device dimensions continue to shrink, the impact of fabrication defects becomes more severe. Defects that were once considered tolerable can now cause significant yield losses and device failures in advanced integrated circuits. To maintain high yields and product quality, semiconductor manufacturers must implement stringent defect control strategies and continuously monitor the fabrication process for the presence of defects.

(Ebayyeh & Mousavi, 2020) stated that normally three types of wafer defects occur, which are random defects, systematic defects and mixed defects. Random defects are scattered haphazardly across the wafer. These defects are caused by unpredictable factors, such as particles in the clean room, and exhibit no discernible pattern.

Systematic defects, in contrast, follow a regular pattern and are typically attributed to issues within the manufacturing process itself. (Wu et al., 2015). Examples include the ring pattern resulting from misalignment in the storage-node process, and the scratch pattern caused by agglomerated particles during the chemical mechanical process.

Lastly, mixed defects, as the name suggests, combine elements of both random and systematic defects on a single wafer map. This type of defect is the most prevalent in the semiconductor manufacturing process. Figure 2.5 shows the most recognized defect patterns.

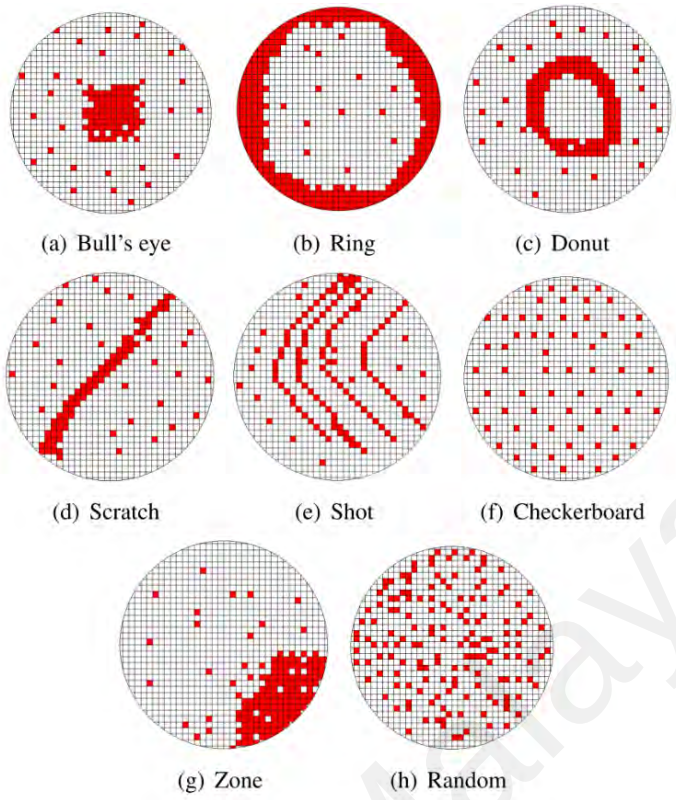


Figure 2.5: Fabrication defect pattern types (Ebayyeh & Mousavi, 2020)

Automated wafer inspection systems and defect classification techniques play a crucial role in identifying and mitigating fabrication defects. Recent advancements in machine learning, particularly deep learning algorithms such as convolutional neural networks, have shown promise in improving the accuracy and efficiency of defect detection and classification. However, challenges remain in detecting sub-resolution defects and adapting to the ever-evolving landscape of defect types that emerge with each new technology node.

2.2.3 Test-Induced Defects

Test-induced defects are another major category of defects that can occur in semiconductor wafers during the manufacturing process. Unlike fabrication defects that

originate from issues in the wafer fabrication steps, test-induced defects are introduced during the wafer testing phase. (Cheng et al., 2021; Cheng et al., 2020) provide a comprehensive overview of test-induced defects, their causes, and their impact on semiconductor manufacturing yield.

One of the key characteristics of test-induced defects is their spatial pattern on the wafer. Since dies are probed in a predetermined order during wafer testing, test-induced defects often exhibit specific patterns that can be captured in test paths. (Cheng et al., 2021) highlight that these patterns can be leveraged to effectively identify test-induced defects.

Another characteristic of test-induced defects is their repetitive nature over an area on the wafer. This phenomenon occurs when a faulty probe pin, responsible for making contact during testing, drags across the wafer's surface. As the faulty pin moves, it consistently causes similar defects at the same position on multiple dies throughout the wafer. These repeated defects share common visual properties, such as colour and pattern, making them easily identifiable. Figure 2.6 illustrates an example of repeated defects on the wafer. As can be seen, the defects are represented by 4 red dots, all the dots are on the same location of the die. Based on the images they also look almost the same in colour and pattern.

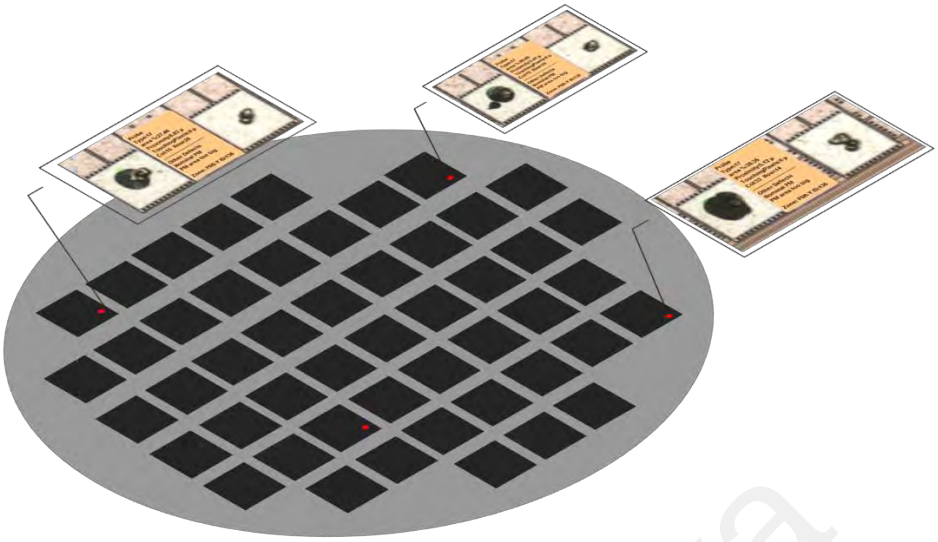


Figure 2.6: Repeated defects example

The primary causes of such repeated defects can be attributed to issues with the probe card itself. For instance, the presence of liquid on the probe pin can lead to contamination and subsequent defects. Similarly, shifted probe pins or pins with short tip lengths can also contribute to this problem.

In addition to probe card-related factors, defects on the wafer itself can also give rise to repeated test-induced defects. The presence of liquid FM (foreign material) on the wafer's surface can interact with the probe pins, resulting in the formation of similar defects at the same location on multiple dies. (Shankar & Zhong, 2005).

Identifying and mitigating test-induced defects is crucial for improving semiconductor manufacturing yield. By accurately classifying test-induced defects and distinguishing them from fabrication defects, manufacturers can take targeted corrective actions.

2.2.4 Comparison and Effects

Fabrication defects and test-induced defects, while both detrimental to semiconductor manufacturing yield, differ in their origins, characteristics, and impact on the final product.

Fabrication defects, as discussed earlier, arise from issues in various manufacturing processes such as lithography, deposition, and etching. These defects can manifest as pattern distortions, non-uniform thickness, or incomplete pattern transfer, among others. The impact of fabrication defects on device performance and reliability can be significant, especially as device dimensions continue to shrink. Even minor defects that were once considered tolerable can now lead to device failures in advanced integrated circuits.

On the other hand, test-induced defects are introduced during the wafer testing phase, where electrical tests are performed to identify functional dies. These defects often exhibit specific spatial patterns on the wafer, following the predetermined probing order. Test-induced defects can also be repetitive, occurring at the same location on multiple dies due to issues with the probe card or the presence of foreign material on the wafer surface.

While both types of defects can lead to yield losses, their impact on the final product may differ. Fabrication defects, if not detected and removed early in the manufacturing process, can propagate through subsequent steps and affect the functionality and reliability of the finished devices. Test-induced defects, however, primarily impact the yield by causing otherwise functional dies to be incorrectly identified as defective.

The different characteristics of fabrication and test-induced defects also necessitate distinct detection and classification approaches. This study focuses specifically on test-

induced defects, which have gained increasing attention as semiconductor devices become more complex and testing requirements more stringent. Identifying and mitigating these defects is crucial for improving overall yield and reliability, as well as reducing the risk of costly field failures and customer dissatisfaction.

2.3 Defect Detection Methods

2.3.1 Conventional Methods

Conventionally, defect detection on semiconductor wafers has relied heavily on manual inspection by experienced engineers. This process is not only time-consuming and labour-intensive but also prone to human errors and subjectivity. (Shankar & Zhong, 2005). As the complexity of integrated circuits increases with shrinking feature sizes, manual inspection becomes increasingly challenging and inadequate to meet the demands of modern semiconductor manufacturing (Huang & Pan, 2015).

Early attempts to automate defect detection primarily focused on simple rule-based systems and basic machine-learning techniques. In their work, (Shankar & Zhong, 2005) developed a template-based machine vision system for inspecting wafer die surfaces. Their method compared the wafer images with a golden template using pixel subtraction and applied rule-based defect specifications to distinguish between critical and non-critical defects. While this system demonstrated improved performance compared to manual inspection, it still had limitations in terms of adaptability and robustness (Huang & Pan, 2015).

One major challenge in conventional defect detection methods is the detection of hidden or latent defects, particularly those induced during the testing process. Test-

induced defects, such as probe marks or surface scratches, can be subtle and difficult to identify using traditional approaches. These defects may not cause immediate failures but can lead to reliability issues and affect the long-term performance of the semiconductor devices.

Conventional methods often fail to capture the intricate patterns and characteristics of test-induced defects, as they rely on pre-defined rules or limited feature representations. This limitation highlights the need for more advanced and adaptive techniques capable of learning from data and discovering hidden patterns.

2.3.2 Machine Learning for Defect Detection

Machine learning techniques have revolutionized the field of defect detection in semiconductor manufacturing, enabling the development of automated systems that can identify and classify defects with high accuracy. These approaches can be broadly categorized into supervised learning, unsupervised learning, semi-supervised learning and deep learning methods, each with its strengths and limitations. Recent reviews by (Li & Kang, 2023; Ma et al., 2023; Theodosiou et al., 2023) provides comprehensive insights into state-of-the-art machine learning techniques for wafer map defect pattern recognition. Here we review some of the works done recently in the field of detection and classification of wafer defects patterns and highlight their strengths and limitations.

2.3.2.1 Supervised Learning Methods

Supervised learning methods have been widely applied to tackle this challenge, leveraging labelled data to train models capable of accurately identifying and categorizing various types of defects.

(Baly & Hajj, 2012) employed an SVM classifier to categorize 1,150 wafer images into high-yield and low-yield classes, demonstrating superior performance compared to decision trees, k-nearest neighbor (KNN), partial least squares regression, and generalized regression neural networks. The nonlinear SVM model exhibited better classification accuracy for wafer defect patterns. Nonetheless, the binary classification approach may not provide fine-grained insights into specific defect types, including those caused by latent testing processes.

Building on the work of (Baly & Hajj, 2012), several researchers have further explored the application of SVM for wafer defect detection and classification. (Xie et al., 2014) proposed a wafer defect pattern detection scheme based on SVM algorithms. They utilized linear, Gaussian, and polynomial kernels, selecting the kernel with the smallest test error through cross-validation for SVM training. This method demonstrated robustness against false positives caused by image translation or rotation. However, like previous approaches, it may struggle with detecting subtle, test-induced defects that do not manifest as clear visual patterns.

(Saqlain et al., 2019) proposed a voting ensemble classifier consisting of logistic regression, random forest, gradient boosting machine, and artificial neural networks for wafer map defect pattern recognition. The ensemble approach combines the best results of all classifiers to obtain the final classification, demonstrating improved accuracy compared to individual models. However, the method relies on handcrafted features and may struggle to detect subtle defects caused by repeated testing. (Li et al., 2021) proposed an AdaBoost-based method for wafer defect pattern recognition, focusing on scratch patterns. The five-step methodology enhances pattern visibility and achieves over 89%

recognition rate for scratch patterns and over 94% for common defect patterns. However, the method's performance on hidden test-induced defects is not explicitly addressed.

2.3.2.2 Unsupervised learning methods

Unsupervised learning methods, such as clustering and anomaly detection, have been explored to overcome some of the limitations of supervised learning. These methods do not require labelled data and can potentially discover new defect types. A comprehensive review by (Naeem et al., 2023) show the bright potential of these algorithms.

One prominent unsupervised learning approach in wafer defect detection is the use of clustering algorithms. (Jin et al., 2019) proposed a novel framework based on the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm for wafer map defect pattern detection and classification. Their method selectively removed outliers according to defect pattern characteristics, enabling simultaneous detection of abnormal points and defect patterns. The DBSCAN-based approach showed promise in handling complex defect patterns and noise reduction. However, the authors noted that the performance of DBSCAN can be sensitive to parameter selection, particularly in cases where sample density is not uniform or the dataset is very large, potentially leading to long convergence times and suboptimal clustering results.

(Huang, 2007) proposed an innovative approach using self-supervised multilayer perceptrons for clustered defect detection in high-quality chips. This method aimed to enhance feature extraction capabilities through the use of multilayer perceptrons. The self-supervised nature of the approach allowed it to adapt to new defect patterns without the need for manual labelling. However, the performance of this method can be highly

dependent on the choice of activation functions, which may require careful tuning for optimal results.

2.3.2.3 Semi-Supervised Learning

Semi-supervised learning (SSL) approaches have emerged as a promising middle ground, combining the strengths of supervised and unsupervised learning. (van Engelen & Hoos, 2020; Y C a et al., 2018) SSL methods leverage both labelled and unlabelled data to improve defect detection accuracy while reducing the reliance on extensive labelled datasets. The core principle of SSL is to utilize the underlying structure and patterns in the unlabelled data to enhance the learning process. In the context of wafer defect detection, SSL algorithms can exploit the spatial and temporal relationships between defects, as well as the overall wafer map patterns, to better identify and classify defects even with limited labelled examples.

Recent work by (L. L. Y. Chen et al., 2021) proposed a novel SSL approach specifically tailored for wafer map defect pattern recognition. Their method combines a convolutional neural network (CNN) with few-shot and self-supervised learning techniques. By incorporating unlabelled data through self-supervised pretext tasks, they demonstrated improved performance on the WM-811K dataset, a benchmark for wafer map defect classification. While this approach shows promise in capturing complex spatial patterns, it may still struggle with rare or previously unseen defect types.

2.3.2.4 Deep Learning

Convolutional Neural Networks (CNNs), a type of deep learning model, have proven to be highly effective for this task. CNNs are specifically designed to process data that

has a grid-like structure, such as images. They consist of multiple layers of neurons, each of which is responsible for extracting different features from the input data. The initial layers of a CNN typically learn basic features, such as edges and lines, while the deeper layers learn more complex features, such as shapes and objects.

The application of CNNs in wafer defect detection has evolved significantly over recent years. (Nakazawa & Kulkarni, 2018) introduced a CNN for wafer map defect pattern recognition and image retrieval, demonstrating the potential of transfer learning by training on simulated wafer maps and achieving high classification accuracy on real wafer maps. Building upon this foundation, (Saqlain et al., 2020) developed a deep CNN model for wafer defect identification (CNN-WDI) that outperformed several previous models, achieving 96.2% classification accuracy on the WM-811K dataset.

(Tsai & Lee, 2020) developed a lightweight neural network for wafer map classification based on data augmentation. Their approach combined a CNN with innovative data augmentation techniques to improve classification performance while reducing computational complexity. This method showed promise in handling limited data scenarios and improving generalization. However, its focus on static defect patterns may limit its effectiveness in capturing the dynamic nature of gradually fading test-induced defects.

(S. Chen et al., 2021) explored the use of transfer learning with CNNs for wafer map defect recognition. They utilized a 29-layer deep CNN model pre-trained on the MNIST dataset, which was then fine-tuned on wafer map data. This approach aimed to minimize the need for large wafer-specific datasets, achieving over 94.9% accuracy on a balanced

dataset of seven defect patterns. While this method shows potential for adapting to new defect types with limited data, it may not explicitly address the challenge of detecting defects that gradually lose visibility over repeated testing.

Recent research has further advanced these techniques to handle more complex scenarios. (Wang et al., 2020) introduced a deformable convolutional network (DC-Net) for efficient mixed-type wafer defect pattern recognition. The DC-Net's innovative use of deformable convolution layers allows dynamic adaptation of the receptive field, enabling better handling of complex and mixed defect patterns. This approach achieved a classification accuracy of 93.2% on the WM-811K dataset, demonstrating significant progress in dealing with diverse defect types.

Addressing the persistent challenge of imbalanced datasets, (Geng et al., 2021) proposed a novel approach combining few-shot learning and self-supervised learning. Their method, utilizing a CNN with Inception blocks, incorporates a few-shot loss for labelled data and a self-supervised loss for unlabelled data. This innovative technique showed superior performance compared to traditional machine learning methods, particularly in scenarios with limited labelled data.

(Yu et al., 2021) further explored few-shot learning by proposing a hybrid self-attention mechanism and prototype network for wafer map defect recognition. This approach aims to improve classification performance with small amounts of data, which is particularly relevant in semiconductor manufacturing where new defect types may emerge frequently.

To tackle the complexity of mixed-type defect patterns, (Li & Wang, 2021) developed an improved Mask R-CNN model. By integrating a feature pyramid network and an enhanced non-maximum suppression algorithm, their model showed superior performance in detecting small objects and shallow defects, addressing a critical challenge in wafer defect detection.

(Kim et al., 2021) conducted a comparative study of various CNN architectures, including VGG16, ResNet, MobileNetV2, and ShuffleNet, for wafer defect pattern identification. They also incorporated out-of-distribution learning to handle undefined defect patterns, providing valuable insights into the trade-offs between model complexity and performance.

2.3.2.5 Graph Neural Networks (GNNs)

Graph Neural Networks (GNNs) have emerged as a promising advancement in deep learning, particularly for tasks involving graph-structured data (Scarselli et al., 2009; Wu et al., 2021). GNNs extend traditional deep learning architectures, such as convolutional neural networks (CNNs), to operate directly on graphs, enabling them to capture both the feature information of individual nodes and the structural information encoded in the graph topology (Bronstein et al., 2017). By learning node representations through the iterative aggregation of neighbourhood information, GNNs have shown remarkable performance in various applications, including node classification, link prediction, and graph classification (Zhou et al., 2018).

In the context of semiconductor wafer defect detection, GNNs offer a powerful framework for leveraging the spatial and hierarchical relationships present in wafer map

data. By representing the wafer as a graph, with nodes corresponding to individual dies and edges encoding their spatial connections, GNNs can learn rich, context-aware representations that capture both local and global defect patterns. Moreover, the message-passing mechanism of GNNs allows for efficient propagation of defect information across the wafer graph, potentially enabling the detection of subtle or hidden defects that may be missed by traditional methods (Wu et al., 2021).

Recent works have demonstrated the potential of GNNs for wafer defect classification and yield prediction, showcasing their ability to outperform conventional CNNs and other machine-learning approaches. However, while GNNs show promise in this domain, they may not be the most suitable approach for the specific problem of detecting test-induced hidden defects in semiconductor wafers, as discussed in the limitations section.

2.3.3 Limitations of Conventional Methods and Machine Learning approaches & hidden defects

Despite the advancements in conventional methods and machine learning approaches for defect detection in semiconductor manufacturing, there remain significant limitations in their ability to detect hidden or latent defects, particularly those induced during the testing process. As discussed in the previous sections, conventional methods heavily rely on manual inspection by experienced engineers, which is not only time-consuming and labour-intensive but also prone to human errors and subjectivity These methods often fail to capture subtle defects and struggle to keep pace with the increasing complexity of integrated circuits.

Machine learning techniques, including supervised and unsupervised, semi-supervised and deep learning methods, have shown promise in automating defect detection and classification tasks.

Supervised learning such as support vector machines (SVM), and deep learning methods such as convolutional neural networks (CNN) and Graph Neural Networks (GNNs), have demonstrated high accuracy in identifying known defect types. However, supervised learning methods have several limitations. Firstly, they require a large amount of labelled data for training, which can be time-consuming and costly to obtain in a manufacturing setting. Secondly, these methods are limited to detecting known defect types and may struggle with identifying novel or previously unseen defects. Lastly, and most importantly, as the prober moves throughout the wafer, the faulty site-number that caused repeated defects would gradually lose the effect, resulting in less visible defects that supervised learning methods fail to detect due to the contrast and lighting settings in the defined algorithm.

On the other hand, unsupervised learning methods also have their drawbacks. They may struggle to distinguish between critical defects and benign process variations, leading to high false positive rates. Additionally, the performance of unsupervised learning methods often depends on the choice of hyperparameters and the quality of the input data, which can be challenging to optimize in a production environment.

While the recent work on using semi-supervised learning has shown promising results, several challenges remain. The complexity and variability of test-induced defects pose significant difficulties for SSL methods. These algorithms often struggle to capture subtle

differences between defect types, especially when the labelled data is limited or not representative of all possible defect patterns. Additionally, the dynamic nature of semiconductor manufacturing processes means that new defect types may emerge over time, requiring SSL models to adapt continuously. Furthermore, most SSL approaches assume that the unlabelled data follows a similar distribution to the labelled data. However, in semiconductor manufacturing, this assumption may not always hold due to process variations and evolving defect patterns. This mismatch can lead to degraded performance or even negative transfer, where incorporating unlabelled data harms the model's accuracy.

One of the major challenges in defect detection is the identification of hidden or latent defects, such as those induced during the testing process. Test-induced defects, including probe marks or surface scratches, can be subtle and difficult to detect using conventional methods and machine learning approaches. These defects may not cause immediate failures but can lead to reliability issues and affect the long-term performance of semiconductor devices.

The limitations of existing methods in detecting hidden defects can be attributed to several factors. Firstly, these methods often rely on pre-defined rules or limited feature representations, which may not effectively capture the intricate patterns and characteristics of test-induced defects. Secondly, as the prober moves throughout the wafer, the faulty site-number that caused repeated defects would gradually lose its effect, resulting in less visible defects.

2.4 Graph-Based Semi-Supervised Learning

2.4.1 Overview of semi-supervised learning and its advantages

Graph-based Semi-Supervised Learning (GSSL) is a powerful approach that leverages the inherent structure of data to propagate label information from labelled to unlabelled instances. A comprehensive review done by (Chong et al., 2020; Song et al., 2023) provides insights and detailed information into the development and application of GSSL.

By representing the data as a graph, where nodes correspond to instances and edges capture the similarity between them, GSSL can effectively exploit the underlying manifold structure to improve classification performance.

The history of GSSL can be traced back to the early 2000s, with (Zhu et al., 2003) introduced the Gaussian Fields and Harmonic Functions (GRF) method, which formulates the semi-supervised learning problem as a Gaussian random field on the graph. (Zhou et al., 2004) proposed the Local and Global Consistency (LGC) method, which balances the local and global consistency of the classification function on the graph.

The core idea behind GSSL is to construct a graph that encodes the relationships between instances and then use this graph to guide the learning process. Figure 2.7 and Figure 2.8 demonstrate a sample graph construction and label propagation.

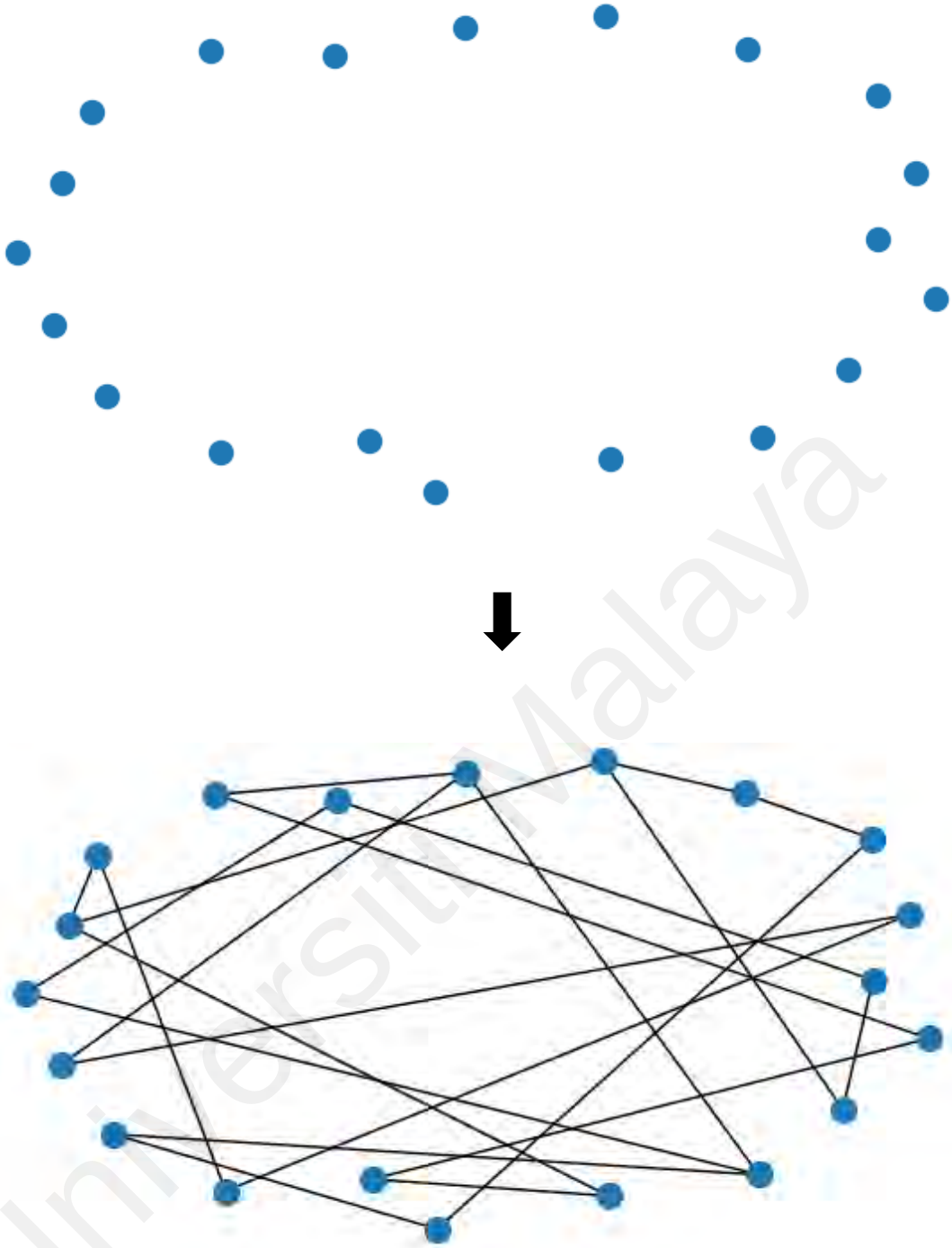


Figure 2.7: Sample of graph construction

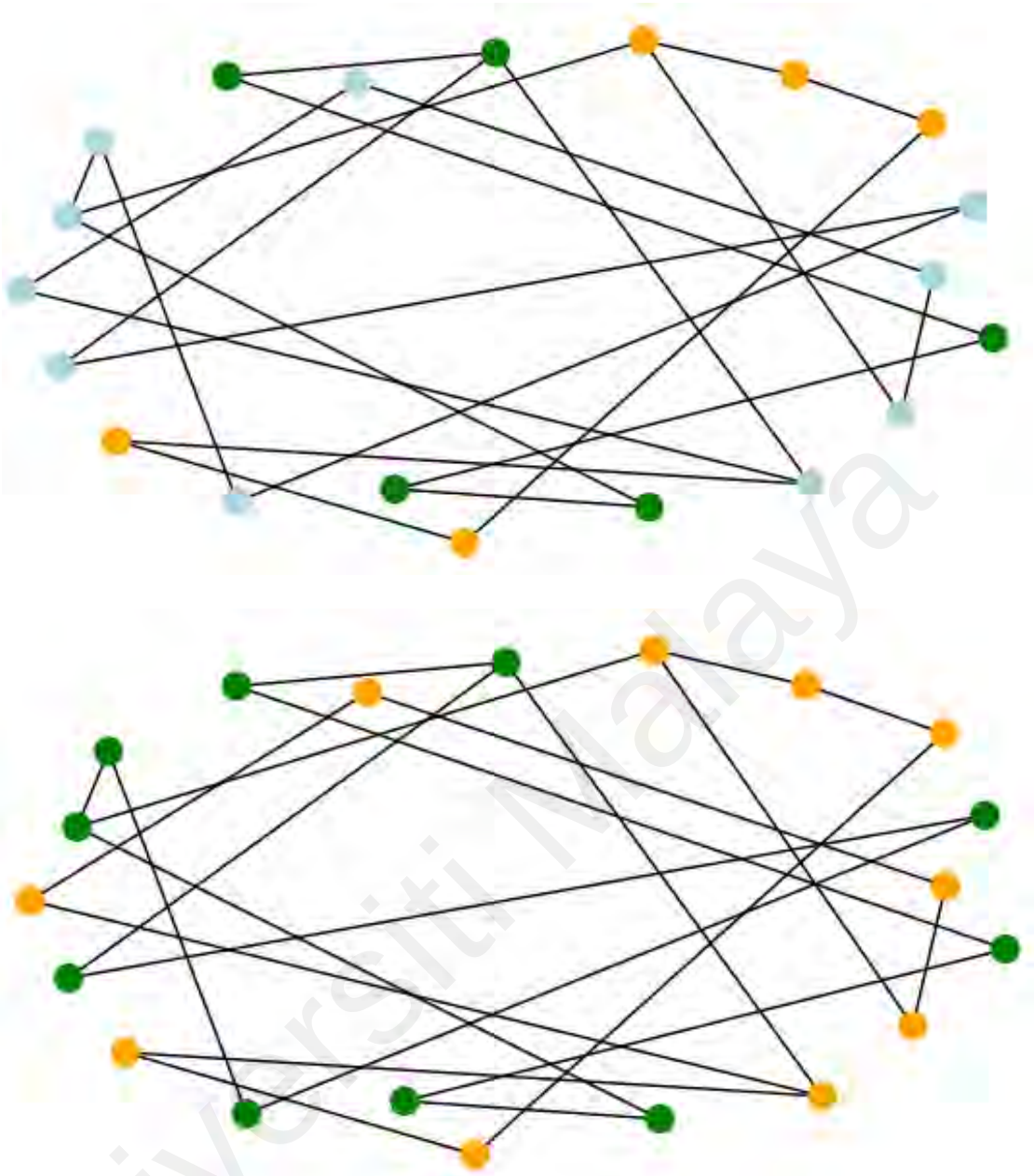


Figure 2.8: Sample of label propagation

One of the key advantages of GSSL is its ability to capture complex data structures and relationships that may not be apparent in the original feature space. This is particularly useful in semiconductor wafer defect detection and classification, where defects often exhibit intricate patterns and dependencies. By constructing a graph that represents the similarities between different wafer regions or defect types, GSSL can

effectively propagate label information from known defects to unknown ones, improving overall classification accuracy.

2.4.2 Graph construction techniques

In GSSL, graph construction plays a crucial role in capturing the relationships between instances. Several techniques have been developed to construct graphs that effectively capture the relationships between data points. This section focuses on five key methods: k-Nearest Neighbors (kNN), b-matching, Linear Neighborhood Propagation (LNP), Robust Graph that Considers Labelled Instances (RGCLI), and Graph Based on Informativeness of Labelled Instances (GBILI).

The k-Nearest Neighbors (kNN) approach is one of the most classic and widely used methods for graph construction in GSSL (Song et al., 2022). In this method, each node is connected to its k nearest neighbors based on a predefined distance metric. While simple and intuitive, kNN can lead to irregular graphs where nodes have varying degrees, potentially impacting the label propagation process.

To address the irregularity issue of kNN, b-matching was introduced (Jebara et al., 2009). This method ensures that every node in the resulting graph has exactly b neighbors, creating a more balanced graph structure. The b-matching approach formulates graph construction as an optimization problem, which can lead to more consistent label propagation.

Linear Neighborhood Propagation (LNP) (Wang & Zhang, 2008) takes a different approach by assuming that each node can be linearly reconstructed by its neighbors. This method not only produces a new way of graph construction but also derives a

straightforward extension to the inductive setting, allowing for the handling of out-of-sample data.

Recognizing the potential value of labelled instances in graph construction, the Graph Based on the Informativeness of Labeled Instances (GBILI) method was developed (Berton & Lopes, 2014). GBILI utilizes the label information during the construction phase, potentially leading to graphs that are more informative for the subsequent label propagation step.

Building upon GBILI, the Robust Graph that Considers Labelled Instances (RGCLI) method was proposed to create more robust graphs (Berton et al., 2017). RGCLI solves an optimization problem that incorporates labelled data, aiming to produce graphs that are more resilient to noise and better suited for semi-supervised learning tasks.

However, these measures may not effectively capture the complex spatial relationships and hierarchical structures present in wafer map data, limiting their ability to accurately propagate label information for detecting hidden defects.

2.4.3 Label propagation algorithms

Label propagation algorithms in GSSL aim to assign labels to unlabelled nodes by minimizing a cost function that encourages label smoothness over the graph. Popular label propagation methods include Gaussian Random Fields (GRF), Local and Global Consistency (LGC), Special Label Propagation (SLP).

Gaussian Random Fields (GRF), introduced by (Zhu et al., 2003), is one of the earliest and most popular approaches in GSSL. GRF formulates the label propagation problem as

estimating a function f on the graph that satisfies two criteria: it should be as close as possible to the given labels on labelled nodes, and it should be smooth on the entire graph. The method uses a quadratic energy function to ensure smoothness, which leads to a harmonic function solution. GRF enforces a hard constraint that the labels of labelled nodes remain fixed during propagation, which can be beneficial when label information is highly reliable.

Local and Global Consistency (LGC), proposed by (Zhou et al., 2004), extends the GRF approach to a multi-class setting and introduces some key modifications. Unlike GRF, LGC relaxes the hard constraint on labelled nodes, allowing their labels to change slightly during the propagation process. This flexibility can be advantageous when there's potential noise in the initial labels. LGC balances local consistency (neighbouring nodes should have similar labels) and global consistency (the overall label distribution should be smooth) through its objective function. The method also introduces a normalization factor based on node degrees, which helps regulate the influence of high-degree nodes in irregular graphs.

Special Label Propagation (SLP), developed by (Nie et al., 2010), addresses a limitation of previous methods by introducing the capability to detect outliers or discover novel classes in the data. SLP achieves this by adding an extra label category, giving the algorithm the flexibility to assign nodes to a new, previously unseen class. This feature is particularly valuable in real-world scenarios where the unlabelled data might contain instances that don't belong to any of the known classes, such as new types of defects in semiconductor wafer inspection.

2.4.4 Opportunities for extending GSSL to semiconductor wafer analysis

Graph-based semi-supervised learning techniques have the potential to be applied to the problem of detecting defects in semiconductor wafers. By representing the wafer as a graph, where nodes correspond to individual dies and edges represent the spatial relationships between dies, GSSL algorithms can propagate defect information from labelled defective dies to unlabelled dies, enabling more accurate and efficient defect detection.

To extend GSSL to semiconductor wafer analysis, several challenges need to be addressed:

i. **Graph construction:** Designing effective graph construction techniques that capture the spatial and structural relationships between dies on the wafer. This may involve incorporating domain knowledge about the wafer layout and the defect patterns.
ii. **Label propagation:** Adapting label propagation algorithms to handle the specific characteristics of semiconductor wafer data, such as the sparsity of labelled defective dies and the potential imbalance between defective and non-defective classes.
iii. **Incorporation of wafer-specific features:** Integrating additional features, such as process parameters or electrical test measurements, into the graph-based models to improve defect detection performance.
iv. **Scalability:** Develop scalable GSSL algorithms that can handle large-scale wafer data and enable real-time defect detection and analysis.

By addressing these challenges and leveraging the power of graph-based semi-supervised learning, novel approaches can be developed to enhance the accuracy and efficiency of defect detection in semiconductor wafers, leading to improved yield and quality control in the manufacturing process.

2.5 Research Gap

The current literature on defect detection in semiconductor manufacturing has two main gaps. First, there is a lack of focus on detecting hidden defects induced by the testing process itself, which can significantly impact the quality and reliability of semiconductor devices. Existing methods fail to effectively leverage the spatial and topological information inherent in visible test-induced defect patterns to uncover these hidden defects.

Second, the limitations of existing graph-based semi-supervised learning (GSSL) methods in addressing the unique challenges posed by hidden defect detection in wafer maps. Current GSSL methods often rely on simple similarity measures for graph construction, which may not effectively capture the complex spatial relationships and hierarchical structures present in wafer map data. Moreover, these methods often lack consideration for domain-specific knowledge, adaptive graph refinement mechanisms, and weighted edge initialization based on the unique characteristics of semiconductor wafer testing.

To address these gaps, this work proposes a novel graph-based semi-supervised learning algorithm specifically designed for test-induced hidden defect detection in semiconductor wafers. The proposed approach integrates three key innovations:

i. a specialized graph construction mechanism that incorporates domain knowledge of wafer testing patterns and defect propagation behaviour.
ii. an adaptive edge weighting scheme that dynamically adjusts based on both spatial relationships and test configuration parameters.
iii. an iterative label propagation process that leverages information from visible defects to identify hidden ones.

By combining these elements, the algorithm aims to capture the complex spatial and hierarchical relationships more effectively in wafer map data while maintaining computational efficiency. The method utilizes minimal labelled data and incorporates test-specific characteristics such as probe card movement patterns and site number configurations to enhance detection accuracy. This comprehensive approach promises to improve both the identification of test-induced hidden defects and the overall reliability of semiconductor manufacturing quality control.

CHAPTER 3: METHODOLOGY

3.1 Introduction

In this study, a machine-learning based solution is proposed to effectively identify hidden defects in semiconductor wafers. The proposed method involves a multi-step process designed to enhance the identification accuracy of these defects. The core part of the methodology is the development of a graph-based semi-supervised algorithm which is to construct the graph and build the label propagation. The graph construction requires the information on wafer map and visible defects, and label propagation requires information on test configuration.

In order to achieve the quantitative research, the project collaborated with NXP Semiconductor Malaysia to study the wafer's defects, recognize the pattern of the defects, and identify the location of the hidden defects. The proposed methodology leverages the power of data analysis and machine learning to identify these hidden defects. The methodology flowchart is shown in Figure 3.1, and it involves several key steps:

i. **Data Acquisition:** Defect data will be collected from wafers tested by the Automated Optical Inspection (AOI) machine at the fabrication plant of NXP Malaysia. Data from a variety of products manufactured over the past two years will be included. A dedicated software tool will be developed to integrate and preprocess this data.
ii. **Test-Induced Defect Discovery:** Statistical analysis will be performed on the pre-processed data to identify recurring patterns in defect distribution. These visible defects are used as the backbone of the algorithm to find the hidden ones.

iii. **Defect Clustering:** Clustering techniques will be employed as a separate step to identify specific areas of interest on the wafer where defects are more likely to occur. This analysis focuses the investigation on regions with a higher concentration of defects.

iv. **Root-Cause Analysis:** Data analysis will be conducted to establish connections between defect types and test configurations, revealing the underlying causes of these defects. This initial analysis lays the groundwork for subsequent hidden defect pattern recognition.

v. **Hidden Defect Localization with Graph-based Semi-supervised Learning:** Once defect root causes are identified, a novel graph-based semi-supervised learning algorithm will be developed. This algorithm will leverage the spatial relationships between defect patterns on the wafer, represented as a graph structure. It will exploit both labelled data (known defects) and unlabelled data (suspected hidden defects) to achieve high accuracy in pinpointing the locations of hidden defects.

vi. **Validation and Results:** The effectiveness of the proposed algorithm will be validated by presenting the results on comprehensive wafer maps, along with supporting statistical analysis. This will demonstrate the algorithm's ability to uncover previously undetected defects.

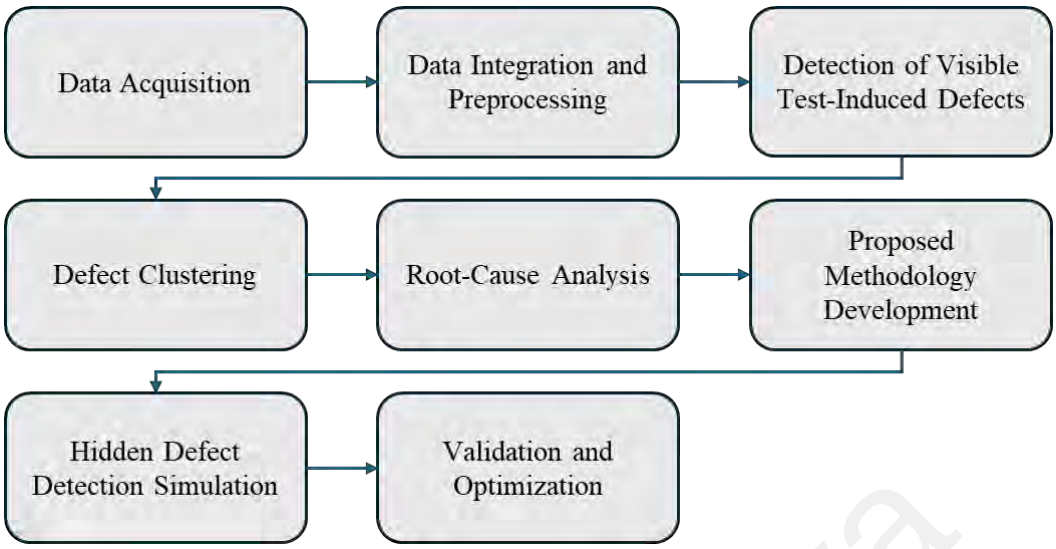


Figure 3.1: Methodology overall process flow

3.2 Data Acquisition

3.2.1 Types and Sources of Data

This project is collaborated with NXP Semiconductor Malaysia for sharing the knowledge and work, thus the dataset is provided by the probe department. The dataset is the collection of the wafers tested and analysed by the team from the Probe Department. The data collected from NXP are as follows:

i. **AOI Machine Report:** The Automated Optical Inspection (AOI) Machine employs high-resolution image capture and computer vision algorithms to scan and analyse the wafer surface. The report provides valuable information and images of defects detected by the AOI machine during the manufacturing process. This data is crucial for discovering visible test-induced defects that may have occurred during testing. The report is in text format and contains detailed information about the location, size, and type of defects classified by the AOI machine. The primary focus of this data is

to utilize the defect location data to gain insights into the distribution and patterns of test-induced defects.

ii. **Physical Test Configurations & Results:** The Physical Test Configurations & Results data encompasses information about the test configurations used for each wafer and the corresponding test results. As mentioned in Chapter 2, each test performed on a wafer is based on a configuration file designed by test engineers in the plant. These configuration files contain crucial details such as the path and direction of movement for the prober and the number of dies tested simultaneously. The configuration files are accessible in Excel format, along with the test results. By extracting information from the Excel files, we can correlate the index of each die in the test with the corresponding AOI defect data.

iii. **Previous Cases of Analysis on Hidden Defects:** The third type of data collected pertains to previous cases of hidden defects that were discovered through manual analysis by engineers. This analysis, documented in both Excel and PowerPoint formats, provides valuable insights into the number of hidden defects found and the time spent on manual inspection. The analysis is typically conducted after the rejection and return of the product by the customer. This data serves as a validation benchmark for the developed algorithm. It's important to note that the number of hidden defect cases is relatively small compared to fabrication defects, as test-induced defects occur less frequently.

3.2.2 Data Centralization and Script Development

During the data collection phase, a significant challenge encountered was the disparate storage of required information across different databases. To address this challenge, a custom script was developed to centralize the information for each wafer. This script efficiently merges data from the AOI report and test configuration data into a unified database. The same database is utilized to store the analysis results, which facilitates further trend visualization and in-depth analysis.

3.3 Data Preprocessing

3.3.1 Data Cleaning

The report for each wafer generated by the AOI machine is a text file format and contains the information about each wafer and its defects, such as the defect class type and location. The first step was to read this data in the software and convert the lines written by the AOI machine into data types to be used later in the algorithms.

Besides the conversion, the data needed to be cleaned before being able to use it in the analysis. The report for each wafer contained a duplication of defects (due to the recipe defined for the AOI machine). Therefore, after converting, the software cleans the dataset to ensure no duplication is in the list of defects, as this would cause huge errors in finding the repeated defects, and consequently root-causing and hidden defect identification.

3.3.2 Conversion

As mentioned earlier, the data collected from AOI machine, and test configurations are from two different databases, and consequently required to have correlation preprocessing, to ensure the data acquired from both sources are in the same coordinates.

However, this correlation is challenging due to the data originating from different databases and the coordinate systems used for plotting the defect locations and die indices being different.

To overcome these challenges, two main steps were undertaken:

i. **Rotating the AOI Data to Match the Physical Test Data Direction:** The first step involved rotating the AOI data to ensure that it was aligned in the same direction as the Physical Test Data. This was achieved by comparing the direction information from the Physical Test Data and the AOI machine. A mapping was created to associate the direction strings with their corresponding direction angles in degrees:

UP → 0°

RIGHT → 90°

DOWN → 180°

LEFT → 270°

Let the direction angle for the physical test data be represented as $\theta_{TestData}$, and the direction angle for the AOI machine be represented as θ_{AOI}.

The required rotation degree, $\theta_{rotation}$, was calculated as the difference between θ_{AOI} and $\theta_{TestData}$:

$$\theta_{rotation} = \theta_{AOI} - \theta_{TestData} \quad (3.1)$$

If $\theta_{rotation}$ was non-zero, the AOI wafer map coordinates (x_{AOI}, y_{AOI}) and defect coordinates (x_{Defect}, y_{Defect}) were rotated accordingly using the following transformations:

- For $\theta_{\text{rotation}} = \pm 90°$:

$$x_{\text{AOI\_Rotated}} = y_{\text{AOI\_Rotated}} \quad (3.2)$$

$$y_{\text{AOI\_Rotated}} = -x_{\text{AOI\_Rotated}} \quad (3.3)$$

$$x_{\text{Defect\_Rotated}} = y_{Defect} \quad (3.4)$$

$$y_{Defect\_Rotated} = -x_{Defect} \quad (3.5)$$

- For $\theta_{\text{rotation}} = \pm 180°$:

$$x_{\text{AOI\_Rotated}} = -x_{\text{AOI}} \quad (3.6)$$

$$y_{\text{AOI\_Rotated}} = -y_{\text{AOI}} \quad (3.7)$$

$$x_{\text{Defect\_Rotated}} = -x_{\text{Defect}} \quad (3.8)$$

$$y_{\text{Defect\_Rotated}} = -y_{\text{Defect}} \quad (3.9)$$

- For $\theta_{\text{rotation}} = \pm 270°$:

$$x_{AOI\_Rotated} = -y_{AOI} \quad (3.10)$$

$$y_{AOI\_Rotated} = x_{AOI} \quad (3.11)$$

$$x_{Defect\_Rotated} = -y_{Defect} \quad (3.12)$$

$$y_{Defect\_Rotated} = x_{Defect} \quad (3.13)$$

After rotating the AOI data, it was aligned in the same direction as the Physical Test Data.

ii. **Transforming the AOI Coordinate System to Match the Physical Test Data Coordinate System:** After rotating the AOI data, the next step was to transform its coordinate system to match the Physical Test Data coordinate system. This was achieved by finding the top-right coordinates of the wafer map in both the AOI and Physical Test Data, and then calculating the

translation vector required to align them. For the AOI data, let the top-right coordinates be represented as (x_{AOI_TR}, y_{AOI_TR}). For the Physical Test Data, let the top-right coordinates be represented as $(x_{\text{TestData\_TR}}, y_{\text{TestData\_TR}})$. The translation vector, $(\Delta x, \Delta y)$, was calculated as:

$$\Delta x = x_{\text{TestData\_TR}} - x_{AOI\_TR} \tag{3.14}$$

$$\Delta y = y_{\text{TestData\_TR}} - y_{AOI\_TR} \tag{3.15}$$

If the translation vector was non-zero, the rotated AOI wafer map coordinates $(x_{AOI\_New}, y_{AOI\_New})$ and defect coordinates $(x_{Defect\_New}, y_{Defect\_New})$ were translated using the following transformations:

$$x_{AOI\_New} = x_{AOI\_Rotated} + \Delta x \tag{3.16}$$

$$y_{AOI\_New} = y_{AOI\_Rotated} + \Delta y \tag{3.17}$$

$$x_{Defect\_New} = x_{Defect\_Rotated} + \Delta x \tag{3.18}$$

$$y_{Defect\_New} = y_{Defect\_Rotated} + \Delta y \tag{3.19}$$

By performing these two steps, the AOI data was rotated and translated to match the direction and coordinate system of the Physical Test Data, respectively. This enabled accurate correlation of the die indices from the Physical Test Data with the defect locations from the AOI machine, as they were now in the same coordinate space and aligned in the same direction.

3.4 Detection of Visible Test-Induced Defects

Since we are collecting our wafer map and defects information from the AOI machine, we need to come up with another procedure to find the test-induced defects than

mentioned in (Cheng et al., 2021). As mentioned in the 2nd chapter, the AOI machine is capable of detecting some of the test-induced defects, however as it goes through the general classification, it is just classified as a defect and not specifically a test-induced one.

As mentioned in the literature review, one of the primary indicators of test-induced defects is their repetitive nature, as a defective probe pin on the test card can cause the same defect pattern to be replicated across different die locations. Therefore, one of the main ways of identifying the test-induced defects detected by the AOI machine is to find the repeated defects on the wafer. The main characteristic to define the existence of repeated defects is to have multiple defects on the same area on the dies in the wafer. To achieve this, we define the area on each die, where the size of each area which is also known as the zone is defined by the AOI report. Figure 3.2 shows how the die is divided into zones.

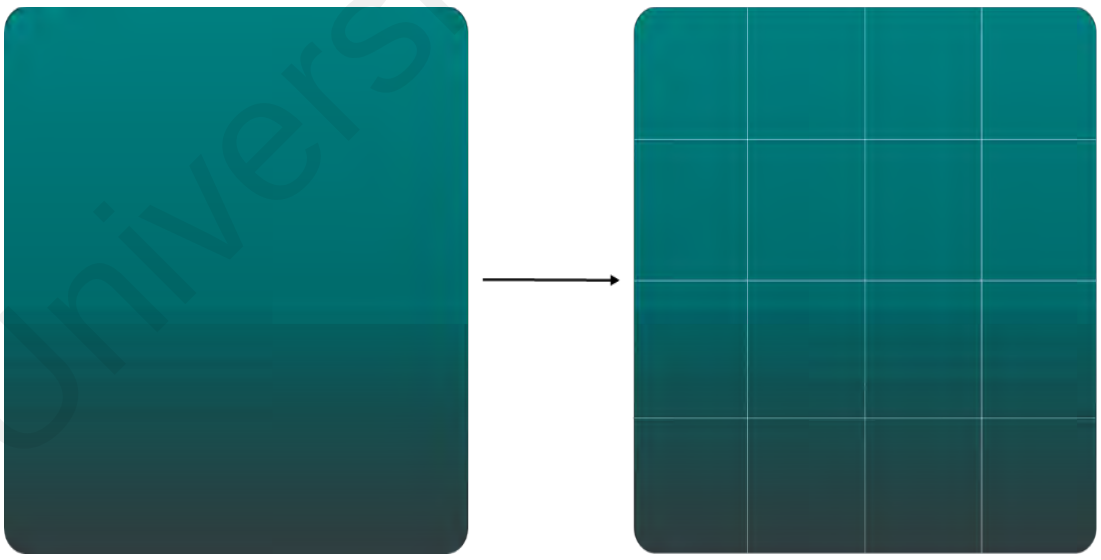


Figure 3.2: Zone division on a Die

In order to find the repeated defects, AOI defect coordinate data is fed into these zones. The more defects that fall at a certain location when stacked into a heatmap will produce red zones that are an early indicator of a repeated defect. Figure 3.3 visualizes how the zones are turned into a heatmap based on the density of defects in each zone. The threshold for identifying the zone as indicator of repeated defect, varies for each device, based on the number of dies in the wafer.

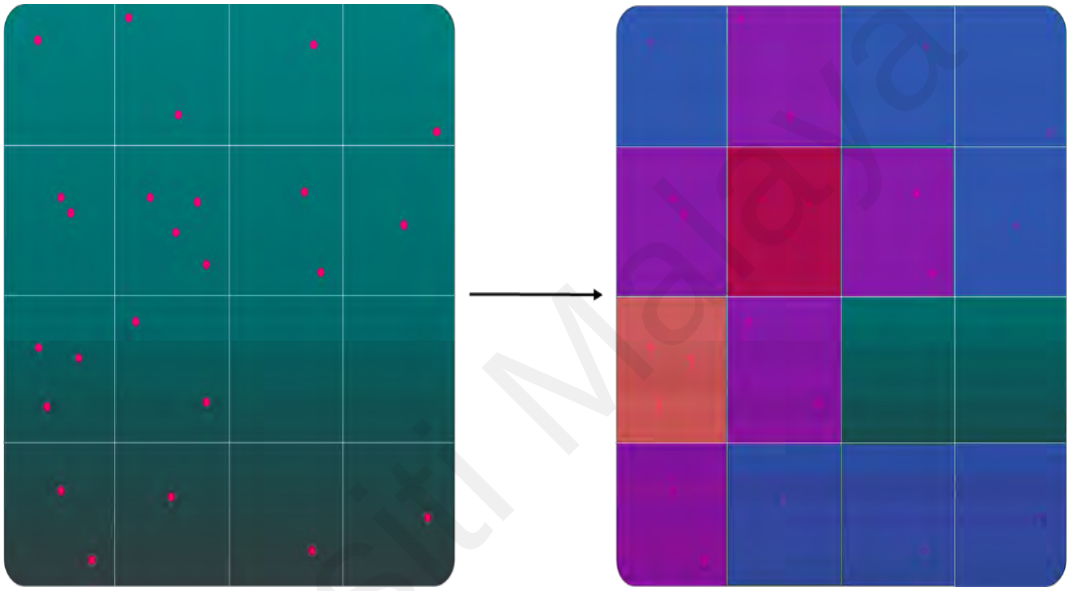


Figure 3.3: Die Heatmap

The more defects that fall at a certain location when stacked into a heatmap will produce red zones that are an early indicator of a repeated defect. This is the very first step in identifying the hidden defects, as we need to identify the faulty wafer first.

3.5 Defect Clustering

Once the test-induced defects are identified, they can be root-caused to find the pattern between them and their relationship with the test configuration, to confirm if it is caused by the probe card.

Each test contains a number of rounds of testing multiple dies at once, which is known as stepping. On each round, the probe card will analyse the dies based on the index configured in the configuration file. This is known as site number.

To analyse the relationship between the defects and the site numbers of steppings, all the steppings that a probe card does for a wafer shall be analysed. However, this would not only take a long processing time to achieve but also require more computation power. To reduce and optimize this process, a clustering algorithm can be used to identify the area of interest on the wafer, to reduce the size, thus reducing the number of steppings going to be analysed for the comparison. Cluster algorithms' goal is the divide a dataset into groups of similar objects that are called clusters. While these algorithms are extremely helpful in finding similar defect types and shapes in the wafer map, they are also used in removing the unwanted defects in the wafer map known as the noise.

To achieve this task, various types of clustering algorithms can be applied. (Djouzi & Beghdad-Bey, 2019; Ezugwu et al., 2022), where each is based on the type of technique chosen for finding the similarities. As the defects do not have a preset number of groups they can join and with the characteristics of the defects, only a few algorithms were selected to be tested based on the review done by (Mahdi et al., 2021) as shown in Figure 3.4.

Clustering Type	Large Dataset Size	High Dimensionality	Stream Processing	Handling Spatial	Handling Noise	Arbitrary Shape	Sequence Independence	Scalable	Average Parameters	Scalability			Algorithm Complexity			Data Type		
										High	Mid.	low	High	Mid.	low	Num.	Cat.	Mix
Density	88%	29%	46%	38%	83%	92%	33%	13%	2.54	21%	38%	42%	29%	21%	46%	71%	21%	8%
Grid	93%	53%	27%	33%	93%	67%	53%	20%	2.07	33%	20%	47%	7%	33%	53%	73%	0%	27%
Hierarchical	58%	21%	5%	11%	21%	74%	53%	16%	1.47	37%	21%	42%	58%	11%	16%	37%	58%	5%
Model	28%	22%	6%	11%	50%	78%	6%	0%	2.17	0%	17%	83%	61%	17%	11%	56%	17%	28%
Partitioning	77%	23%	27%	8%	15%	42%	19%	42%	1.69	46%	23%	31%	46%	23%	19%	65%	27%	8%

Poor
Scale
Good

Figure 3.4: The chosen algorithms for clustering (Mahdi et al., 2021)

Two main parameters can be given to the clustering algorithms:

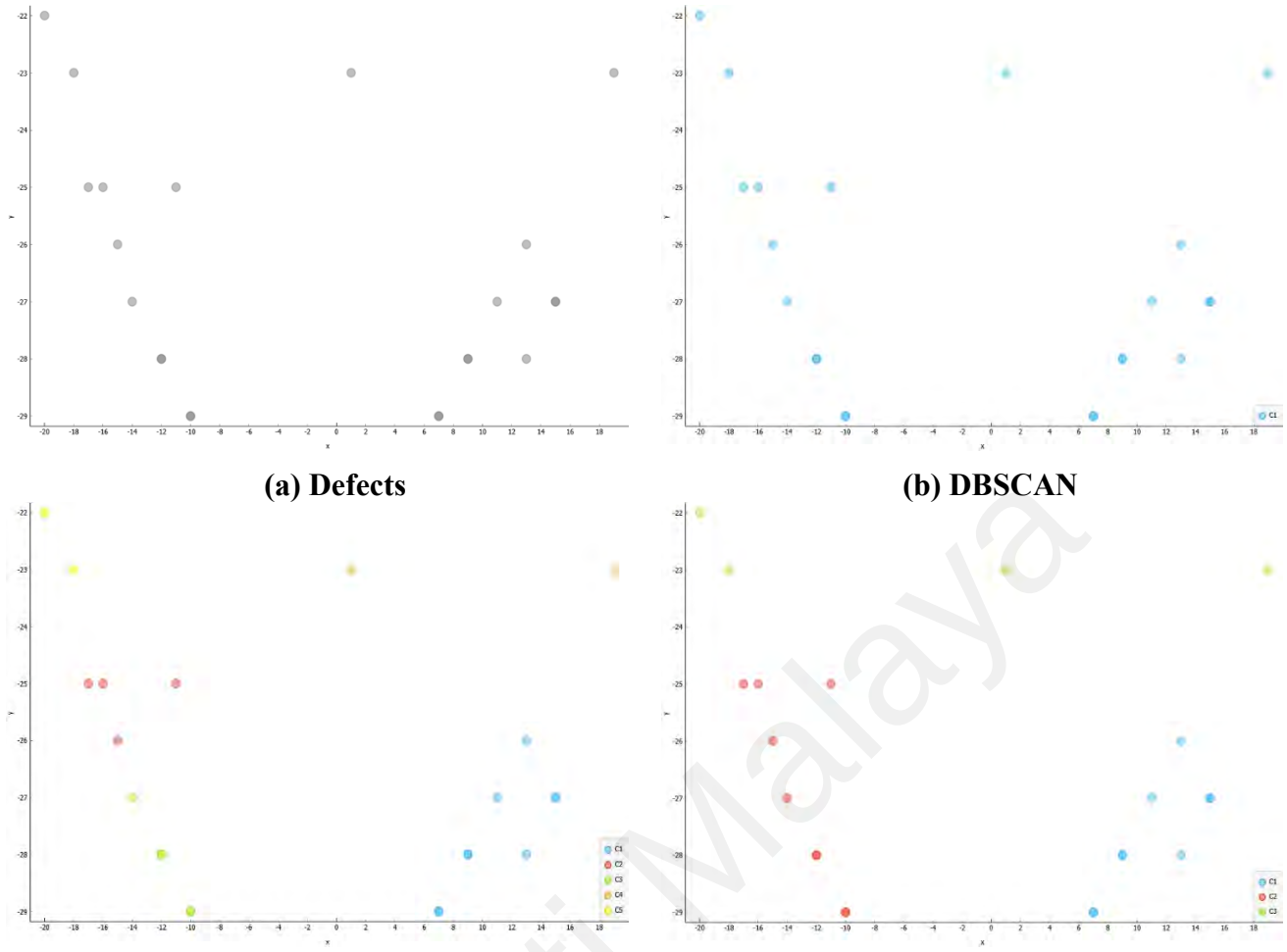
i. The distance between the two defects to form a cluster is the same as the stepping size.

ii. There must be at least two defects in a neighbourhood to define them as a cluster.

To evaluate the effectiveness of different algorithms, a subset of wafers exhibiting repeated defects was subjected to multiple clustering techniques. Among the algorithms considered, Clustering In Quest (CLIQUE) (Jain & Dubes, 1988), Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester et al., 1996) emerged as the most suitable choice for this specific application while Ordering Points To Identify the Clustering Structure (OPTICS) (Ankerst et al., 1999) was close.

To evaluate the effectiveness of different algorithms, a subset of wafers exhibiting repeated defects was subjected to multiple clustering techniques. Among the algorithms considered, DBSCAN emerged as the most suitable choice for this specific application. Figure 4 shows an example of a comparison between DBSCAN (b) and non-density-based algorithms like K-Means (c) and Louvian (d) on a wafer map with defects (a).

DBSCAN effectively groups nearby defects into a single cluster, represented by points of the same colour. In contrast, K-Means and Louvian tend to split defects into multiple smaller clusters, assigning different colours to defects that are likely related. It can be seen that DBSCAN consolidates related defects into a single cluster, unlike non-density-based algorithms that fragment them, which increases computational complexity in subsequent steps. Also, due to the nature of the algorithm, DBSCAN doesn't require the number of clusters to be predefined as input, which makes it particularly suited for handling varying defect patterns. Additionally, DBSCAN's noise cancellation capability filters out irrelevant defects misclassified by AOI systems, enhancing the accuracy of clustering. These attributes make DBSCAN the most suitable algorithm for this application.



(a) Defects **(b) DBSCAN**

(c) K-Means **(d) Louvian**

Figure 3.5: Comparison of clustering algorithms for defect detection on a specific wafer: (a) Original defects; (b) DBSCAN; (c) K-Means; (d) Louvain.

With the chosen algorithm and clustering done, now the area of interest must be identified. As the goal of this study is to find the test-induced hidden defects, and as we know that test-induced defects are repeated throughout the wafer, it is considered that the area between two clusters is an indicator of having repeated defects. To ensure there's no hidden defect being missed, we consider a margin to form the area of interest. The choice of clustering margins was determined empirically based on the observed spatial distribution of defects in the data. Specifically, the margins were set to double the height

and width of defect clusters to ensure the inclusion of all potential test-induced defect areas. This choice balances sensitivity and specificity: narrower margins risk excluding significant defect regions, while overly broad margins could incorporate unrelated noise.

Thus, to define the entire area on the wafers, boundaries of defined based on the location of clusters with a margin of double height/width:

$$y_{\max} = y_{\max\_cluster} + (2 \times height\ of\ testing\ area) \quad (3.20)$$

$$y_{min} = y_{min\_cluster} - (2 \times height\ of\ testing\ area) \quad (3.21)$$

$$x_{\max} = x_{\max\_cluster} + (2 \times width\ of\ testing\ area) \quad (3.22)$$

$$x_{\min} = x_{\min\_cluster} - (2 \times width\ of\ testing\ area) \quad (3.23)$$

3.6 Root-Cause Analysis

After finalizing the area of interest, the root-cause analysis aims to identify the site-number with the highest density of collisions with the defects. This site-number is considered the root cause of the test-induced defects, as concluded in (Cheng et al., 2021) And will be used as part of the label propagation algorithm structure in the next step. As mentioned earlier, the test-induced defects are repeatedly happening throughout the wafer. Therefore, we need to figure out the pattern of repeated defects through the site number representing the repeated defects. This will help us identify the potential coordinates of hidden defects.

The movement of the prober around the wafer for the testing is also called stepping data. The process involves iterating through the stepping data, with each stepping

representing the coordinates of each die and the site-number assigned to that die during the stepping process.

Let $D = \{(x_i, y_i)\}_{i=1}^{N_D}$ be the set of defect locations, where (x_i, y_i) represents the coordinates of the i^{th} defect and N_D is the total number of defects. Let $S = \{(x_{j,k}, y_{j,k}, \text{dut}_k) \mid j = 1, \dots, N_S; k = 0, \dots, M-1\}$ be the set of stepping coordinates, where N_S is the total number of steppings, M is the number of sites tested in each stepping, $(x_{j,k}, y_{j,k})$ represents the coordinates of the k^{th} site in the j^{th} stepping, and dut_k is the site-number associated with the k^{th} position in each stepping. The following lists are initiated for each site-number dut_k to hold their records:

i. $R_{\text{dut}_k} = \emptyset$: an empty list to store defect locations that collide with the site-number.
ii. $T_{\text{dut}_k} = \emptyset$: an empty list to store stepping coordinates and their collision status for the site-number.

The area of interest within the clustering parameters based on (1) will be defined as:

$$\text{A} = \{(x, y) \mid x_{\text{min}} \leq x \leq x_{\text{max}}, y_{\text{min}} \leq y \leq y_{\text{max}}\} \tag{3.24}$$

The data analysis process involves the following steps for each stepping coordinate $(x_{j,k}, y_{j,k}, \text{dut}_k) \in \text{S}$:

i. check if it falls within the area of interest A, otherwise skip the stepping:

$$(x_{j,k}, y_{j,k}) \in A \Leftrightarrow x_{\text{min}} \leq x_j \leq x_{\text{max}} \wedge y_{\text{min}} \leq y_j \leq y_{\text{max}} \tag{3.25}$$

ii. If the stepping coordinate falls within A, check for collisions of site-number coordinates with defect locations:

$$isCollising_{j,k}\begin{cases}1, & \exists(x_i, y_i) \in D \text{ such that } (x_i, y_i) = (x_{j,k}, y_{j,k}) \\ 0, & otherwise\end{cases} \quad (3.26)$$

iii. If $isColliding_{j,k} = 1$, update R_{dut_k} with relative defect location.

iv. Update T_{dut_k} with collision status.

After processing all stepping coordinates, the site-number dut_k^* with the longest list $R_{\mathrm{dut}_k^*}$ is considered the root cause of the test-induced defects:

Figure 3.5 shows an example of how the heatmap of site-number determines the faulty site-number, where the site-number #28 has 20 collisions with test-induced defects. This is a clear indication of the faulty site number. Therefore, coordinates of site-number #28 are the potential for hidden defects.

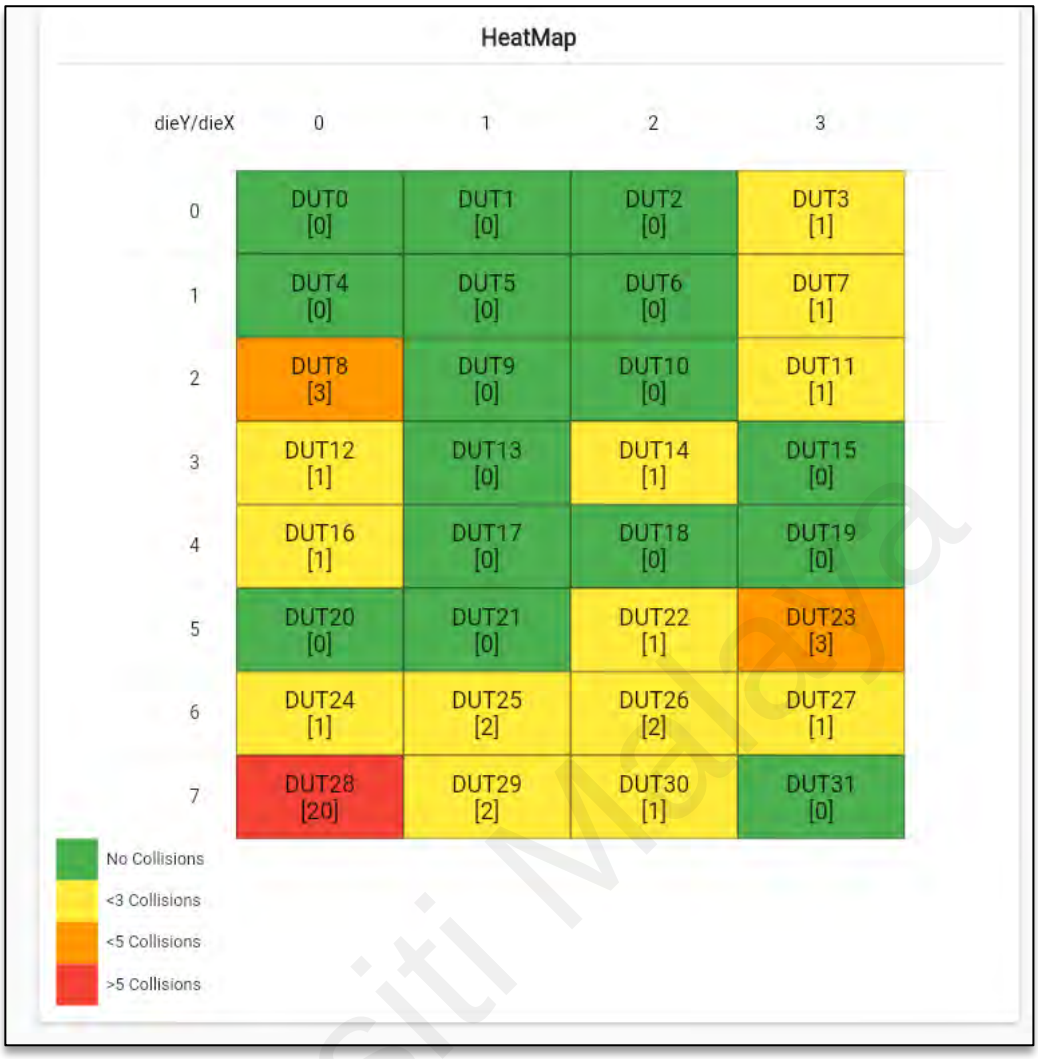


Figure 3.6: Site-number heatmap

3.7 The Proposed Methodology: Improved Graph-Based Semi-Supervised Learning for Defect Detection

With the knowledge of the root-cause, this work proposed a novel graph-based semi-supervised learning (GSSL). The proposed approach consists of two key steps: 1) Graph Modelling and 2) Label Propagation. By leveraging the structural information captured by the graph and requiring only a limited amount of labelled data, the algorithm

propagates defect information to unlabelled dies, effectively identifying hidden defects. Figure 3.6 shows a process flowchart of how the hidden defects are identified.

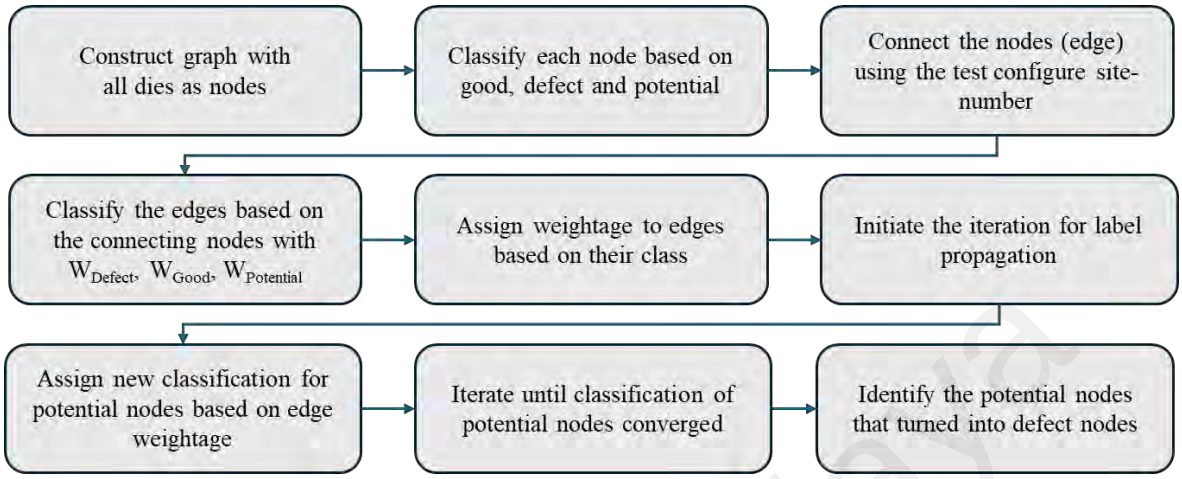


Figure 3.7: Proposed novel GSSL flowchart

3.7.1 Graph Modelling

In this step, we construct an undirected graph $G = (V, E)$, where V represents the set of nodes (dies) and E represents the set of edges connecting the nodes. Each node $v_i \in V$ is associated with a feature vector x_i and a label $y_i \in 0,0.5,1$, representing defective, potential, and good dies, respectively. The edges $e_{ij} \in E$ connect nodes v_i and v_j based on their site number information obtained from the root-cause analysis. To incorporate domain-specific knowledge, we introduce weighted edges that capture the relative importance of different connections in the context of semiconductor wafer testing. The edge weights are denoted as w_{ij} and are initialized based on three parameters: w_D, w_G, and w_P, which correspond to the weightage of defective, good, and potential dies, respectively. These weights are learned during the training stage to optimize the graph structure for accurate hidden defect detection. The graph construction process can be formalized as follows:

$$V = \{v_i | i = 1,2, \dots, N\} \tag{3.27}$$

where N is the total number of dies.

$$E = \{e_{ij} | v_i, v_j \in V, i \neq j\} \tag{3.28}$$

where e_{ij} is an edge connecting nodes v_i and v_j. The edge weights are initialized as:

$$w_{ij} = \begin{cases} w_D, & if \;\; y_i = 0 \;\; or \;\; y_j = 0 \\ w_G, & if \;\; y_i = 1 \;\; or \;\; y_j = 1 \\ w_P, & if \;\; y_i = 0.5 \;\; or \;\; y_j = 0.5 \end{cases} \tag{3.29}$$

These weights are critical in determining the influence of each type of die on its neighbours, reflecting the likelihood of defect propagation.

This graph construction approach differs from existing GSSL methods by incorporating domain-specific knowledge through weighted edges and considering the spatial proximity and site number information obtained from the root-cause analysis. Additionally, the proposed method introduces an adaptive graph refinement mechanism during the label propagation step, which will be discussed next.

3.7.2 Label Propagation

The label propagation step aims to propagate the label information from labelled nodes (good and defect dies) to unlabelled nodes (potential dies) iteratively until convergence. This approach is designed to identify hidden defects in semiconductor wafers by iteratively updating the classifications of potential dies based on their neighbours. This process leverages the graph structure constructed in the previous step and propagates labels through the network until convergence. At the start, each node v_i in the graph G is

assigned an initial label y_i. These initial labels are based on the root-cause analysis. The label assignment follows a normalized range between 0 and 1, where:

- $y_i = 0$ for defective dies, denoting nodes with confirmed defects.
- $y_i = 0.5$ for potential dies, indicating nodes with uncertain status.
- $y_i = 1$ for good dies, representing nodes with confirmed normal functionality.

This normalized range allows for intuitive interpretation, where values closer to 1 indicate higher likelihood of normal functionality, while values closer to 0 suggest higher probability of defects. The initial state of the labels can be represented as:

$$y_i^{(0)} = c_i \tag{3.30}$$

where c_i denotes the class of node i at the initial stage.

The label propagation process updates the label of each potential die iteratively. During each iteration, the label of a node is updated based on the weighted average of its neighbours' labels. The update rule for the label propagation is given by:

$$y_i^{(t+1)} = \frac{\sum_{j \in \mathcal{N}(i)} w_{ij} y_j^{(t)}}{\sum_{j \in \mathcal{N}(i)} w_{ij}} \tag{3.31}$$

where:

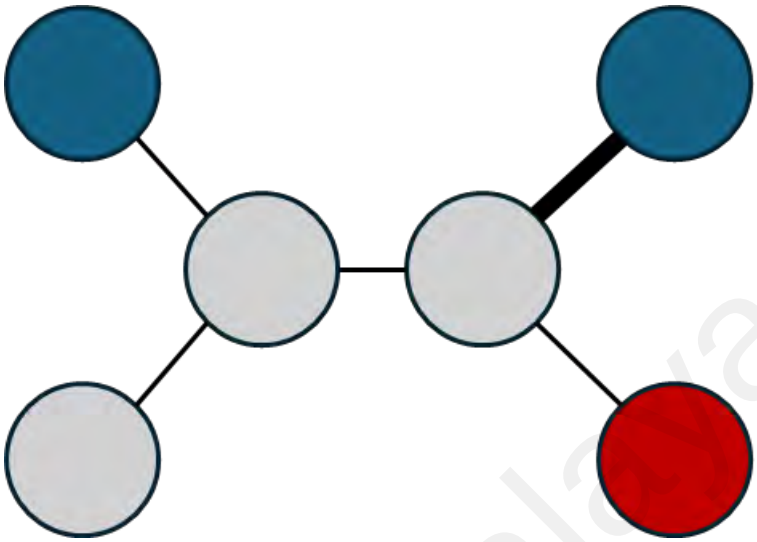
- $y_i^{(t+1)}$ is the updated label of node i at iteration $t + 1$,
- $\mathcal{N}(i)$ denotes the set of neighbours of node i,
- w_{ij} is the weight of the edge connecting nodes i and j,
- $y_j^{(t)}$ is the label of the neighbouring node j at iteration t.

The iterative process continues until the labels of the nodes converge, meaning that the change in labels between iterations falls below a predefined threshold ϵ. The convergence criterion is mathematically defined as:

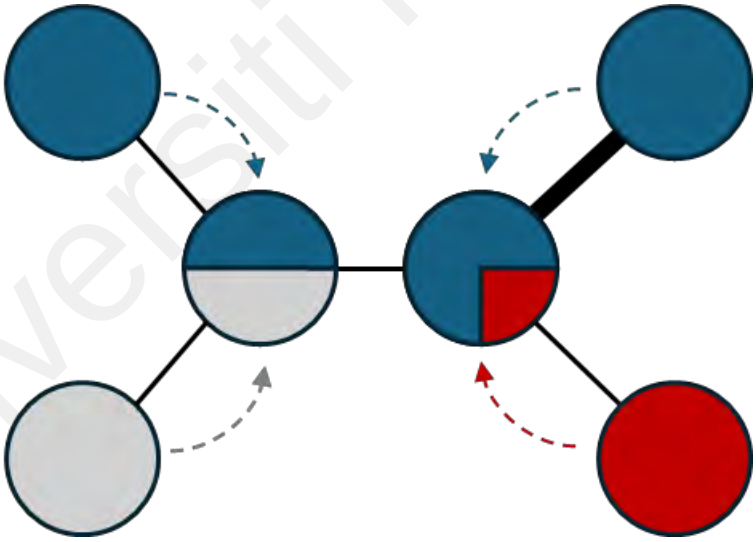
$$\left|y_i^{(t+1)} - y_i^{(t)}\right| < \epsilon \quad \forall i \tag{3.32}$$

Once convergence is achieved, the final labels indicate the probability of each potential die being a hidden defect.

Figure 3.7 shows the graphical representation of label propagation. Figure 3.7 (a) shows the initial state of each node, with weighted edges. Figure 3.7 (b) shows the nodes classification after 1 iteration, in which the potential nodes that were initially identified as grey, get new classification based on the weightage of edge and classes of neighbours. This figure shows the effect of weighted edge as the right potential node is more blue than red, due to higher weightage of the blue node's edge.



(a) Initial State



(b) After 1 iteration

Figure 3.8: Label Propagation example

3.8 Experimental Setup

To evaluate the effectiveness of the proposed graph-based semi-supervised learning algorithm for defect detection in semiconductor wafers, we collaborated with NXP Semiconductor Malaysia and conducted experiments on real product data that had already been analysed by the probe team. For data collection, we utilized both the prober and automated optical inspection (AOI) systems from the production line, as illustrated in Figure 3.8 and Figure 3.9. This comprehensive data acquisition approach allowed us to capture both electrical test results from the prober and visual & statistical defect and wafer map information from the AOI, providing a holistic view of the wafer condition.

Figure 3.9: AOI Machine at NXP Semiconductor Malaysia

Figure 3.10: Prober Machine at NXP Semiconductor Malaysia

The dataset consists of wafer maps with labelled defective dies, non-defective dies, and unlabelled dies, along with the test configuration information, including site numbers and relevant test parameters. Table 3.1 shows the information about the dataset. The experiments aim to assess the performance of the algorithm in detecting hidden defective dies that may have remained undetected by the manual analysis performed by the probe team. For each product category, we divided the wafers into a 60%-40% ratio for training and testing, respectively.

Table 3.1: Experiment Setup

Product Category	# Dies	Sites	# Wafers
PC-1	<2000	16, 28	10
PC-2	<3000	8, 64, 128	30
PC-3	<4000	32, 128	10

3.9 Optimization

We consider the weightage assigned to the edges connecting good dies w_G, defective dies w_D, and potential dies w_P as hyperparameters of the algorithm, as they play a crucial role in determining the algorithm's behaviour and performance. To optimize the performance of our proposed graph-based semi-supervised learning algorithm for wafer defect detection, we conducted an evaluation on these hyperparameters.

We designed four cases of normalized combinations for these hyperparameters to test wafers as shown by Table 3.2. These four cases were strategically chosen to represent a spectrum of weight distributions that align with our understanding of defect propagation in semiconductor wafers. Case 1 serves as a baseline where all die types are equally weighted, providing a neutral starting point. Cases 2 through 4 progressively decrease the weights of good w_G, and potential w_P, dies relative to defective dies w_D, reflecting the hypothesis that defective dies have the strongest influence on their neighbours. This gradual reduction allows us to assess the impact of increasingly emphasizing defective dies in the model. Importantly, these cases were not arbitrarily chosen but were informed by industry expertise and initial experimental results. While a more exhaustive search

could potentially yield marginally better results, our preliminary studies indicated that these four cases effectively capture the most significant variations in model performance.

The choice of these specific hyperparameter combinations is rooted in the physical characteristics of semiconductor wafer defects. In a typical wafer, defective dies tend to have a stronger influence on their neighbors compared to good or potential dies. This is because defects often propagate to adjacent dies due to the nature of the manufacturing process and the proximity of the dies on the wafer. By assigning higher weights to the edges connected to defective dies, the algorithm can more effectively capture this defect propagation behaviour. The progressive reduction of weights for good and potential dies in Cases 2 through 4 allows the model to focus more on the influence of defective dies while still considering the information provided by other die types. This approach aims to strike a balance between leveraging the most informative connections (i.e., those involving defective dies) and maintaining a comprehensive view of the wafer's overall defect landscape. By systematically evaluating these hyperparameter combinations, we can identify the optimal settings that maximize the algorithm's performance in detecting hidden defects, ultimately leading to improved yield and quality control in semiconductor manufacturing.

Table 3.2: Hyperparameter Cases

Case	w_D	w_G	w_P
Case 1	1	1	1
Case 2	1	0.5	0.5
Case 3	1	0.5	0.25
Case 4	1	0.33	0.17

3.10 Limitations of the Methodology

While the proposed graph-based semi-supervised learning methodology demonstrates promising capabilities in identifying test-induced hidden defects on semiconductor wafers, it is important to acknowledge and discuss the limitations inherent in this approach. These limitations stem from various factors, including the reliance on AOI reports, the complexity of test configurations, and the physical characteristics of the prober itself.

One notable limitation arises from the methodology's dependence on the AOI report as the primary source for identifying visible test-induced defects. Although AOI systems provide valuable information about defect locations and types, they may not capture all relevant defects, especially those that are subtle or fall outside the predefined classification criteria. This can lead to an incomplete representation of the defect landscape, potentially impacting the effectiveness of the subsequent steps in the methodology.

Another significant challenge stems from the inherent complexity of test configurations in semiconductor manufacturing. Modern testing processes often involve intricate overstepping patterns, where a single die may be tested multiple times by different site numbers. This redundancy can introduce ambiguity in associating defects with specific site numbers, complicating the root-cause analysis and the construction of the graph model. Overstepping also increases the computational complexity of the methodology, as it requires processing a larger volume of test data and handling more complex relationships between dies and site numbers.

These limitations highlight the need for careful consideration when applying the methodology in production environments and suggest potential areas for future enhancement. Understanding these constraints is crucial for proper implementation and interpretation of the results, particularly in cases where the testing scenario deviates from standard configurations.

CHAPTER 4: RESULTS & DISCUSSION

4.1 Detection of Test-Induced defects

A heatmap based solution has been created that provides the zonal defect density to indicate the location on the die where the most defects appeared. Based on the characteristics of the wafer, the die is divided into zones. The green indicates no defects have appeared on this section of the die throughout the wafer. Blue indicates that there's only one defect that appeared on this location of the die. Red demonstrated the location of the die with the highest number of defects that are usually more than 3.

Figure 4.1 shows different sets of heatmaps that appeared during the analysis. The axes in these heatmaps represent the physical dimensions of a semiconductor die in micrometres, with the x-axis showing the width and the y-axis showing the height. While the visual representation of the heatmaps may appear in different aspect ratios due to the visualization software's screen adaptation, the underlying coordinate system and measurements remain consistent. Each cell in the heatmap corresponds to a specific zone on the die, with the colour coding indicating the frequency of defects in that particular location.

Figure 4.1 (a) represents example of wafers that showed no repeated defects on their maps. This could be seen as they have no red indicator on their heatmaps. Figure 4.1 (b) shows the example of wafers that has only 1 repeated location on their heatmaps. This is not indicative of having only 1 defect, but it shows that all the defects happened on the exact same location on the die which is a clear characteristic of test-induced defects.

In more complex situations, Figure 4.1 (c) shows examples of devices that have multiple locations that have more than one defect frequency on the idea. This is shown by a colour range that goes from blue which was the indicator of 1, to red which is the indicator of the highest number, resulting in having shades of purple and red that indicate the different frequencies on each location. This means that the prober has multiple issues, and it could be harder to determine the location of hidden ones as it has a more complex pattern.

This results and the analysis on it have been verified by engineers of NXP Semiconductors Malaysia through verification of images of these defects.

(a) No repeated defects

Figure 4.1: Die Heatmap for Repeated Defect Detection

(b) 1 Location with a high frequency of defects

(c) Multiple locations with a high frequency of defects

Figure 4.2, continued: Die Heatmap for Repeated Defect Detection

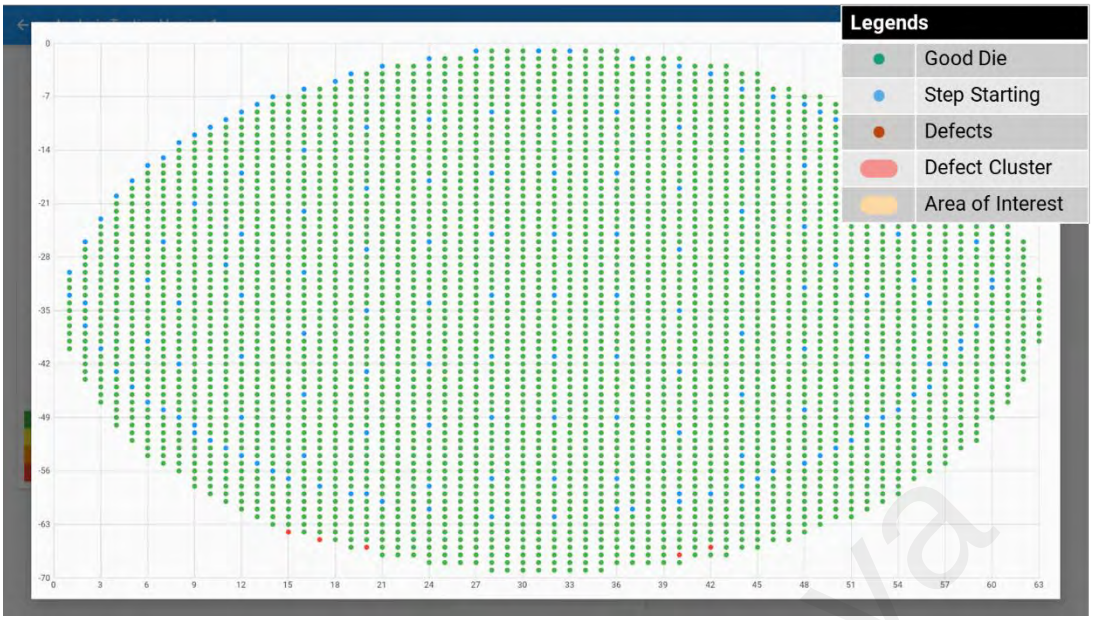
4.2 Area of Interest on the Wafer Map

Clustering visualization is developed based on the selected algorithm, and calculation is done to indicate the area of interest on the wafer that requires the root-causing and identification of hidden defects.

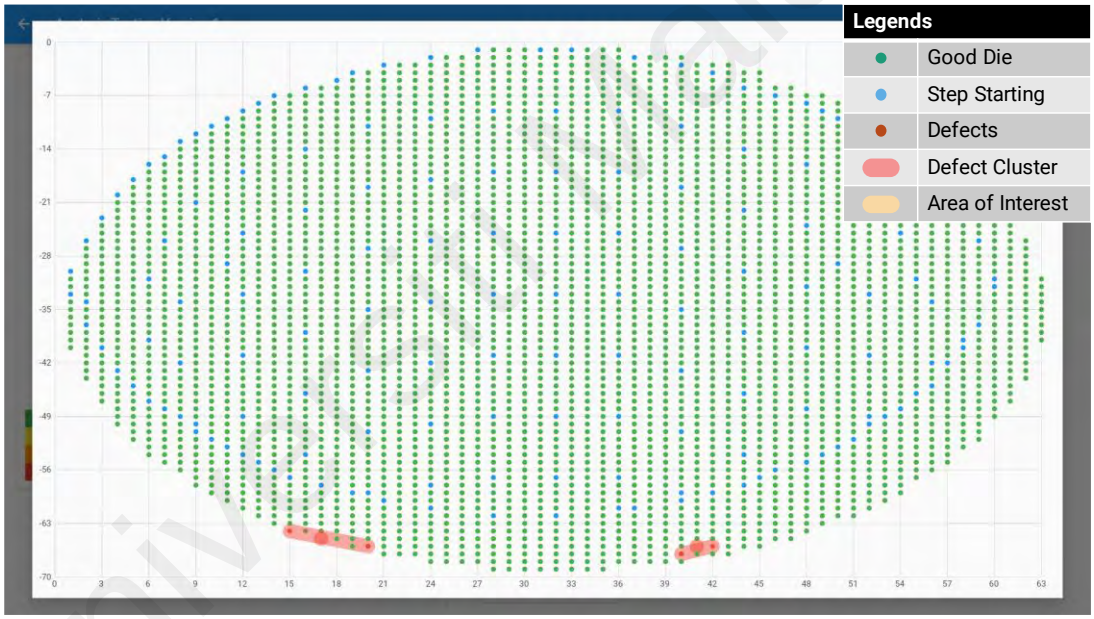
Figure 4.2 shows an example of the 3 steps of defining the area of interest on the wafer for one of the tested wafers. The axes in these wafer maps represent the spatial coordinates of individual dies on the wafer surface. These coordinate systems are referenced relative to the wafer's notch position, which serves as a physical orientation marker. While different software applications may employ varying coordinate representations based on their specific use cases, the methodology presented in this study includes coordinate transformation algorithms to ensure consistent alignment and synchronization across different environments. This standardization enables accurate correlation of die locations across multiple analysis platforms.

Figure 4.2 (a) indicates the initial wafer map, where the defects are shown in red dots. Figure 4.2 (b) shows 2 areas with red background, which is the result of the clustering algorithm (DBSCAN) applied on the defects locations. As it can be seen defects are categorized into 2 clusters, which indicates the efficiency of DBSCAN in identifying of clustering locations.

Figure 4.2 (c) shows the orange background shape appeared on the wafer map, which indicates the area of interest on this wafer map. As it can be seen, it is connecting to clusters and giving a margin that is proportional to the width/height of area that the testing prober tests on the wafer.

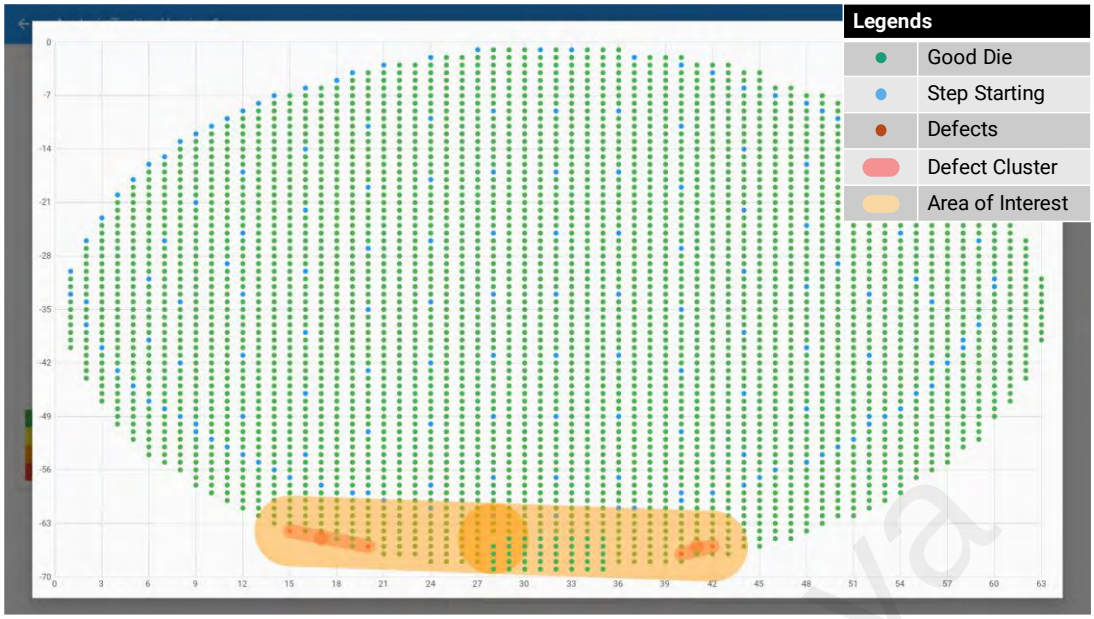


(a) Wafer map



(b) Clustering applied

Figure 4.3: Example of determination of the area of interest on a wafer map using clustering.



(c) Area of interest developed

Figure 4.2, continued: Example of determination of the area of interest on a wafer map using clustering.

4.3 Root-Causing

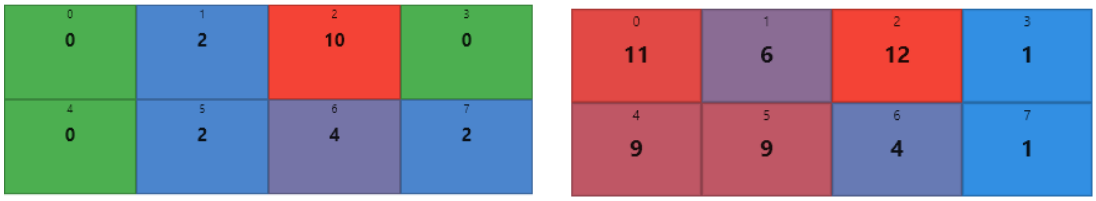
Similar to section 4.1, a heatmap is generated to indicate the root-cause of the repeated defects. Figure 4.3 shows examples of the heatmap for the root-causing representing the density of defects on each site number for each device. Each heatmap represents the collision frequency of site-numbers with defects. The higher the value is in the heatmap, the more it indicates that the relative site-numbers are responsible for test-induced defects.

As mentioned in Chapter 2, the number of dies that are simultaneously getting tested by the prober are different for each device. Figure 4.3 (a) shows the stepping with only 8 site-numbers. It can be seen in the left heatmap, that the faulty site-number can be

identified through the high number of collisions that site-number 2 (10 times) has with the defects.

In other examples illustrated by Figure 4.3 (b), Figure 4.3 (c) and Figure 4.3 (d), we can see steppings with 16, 32, 64 and 128 site-numbers.

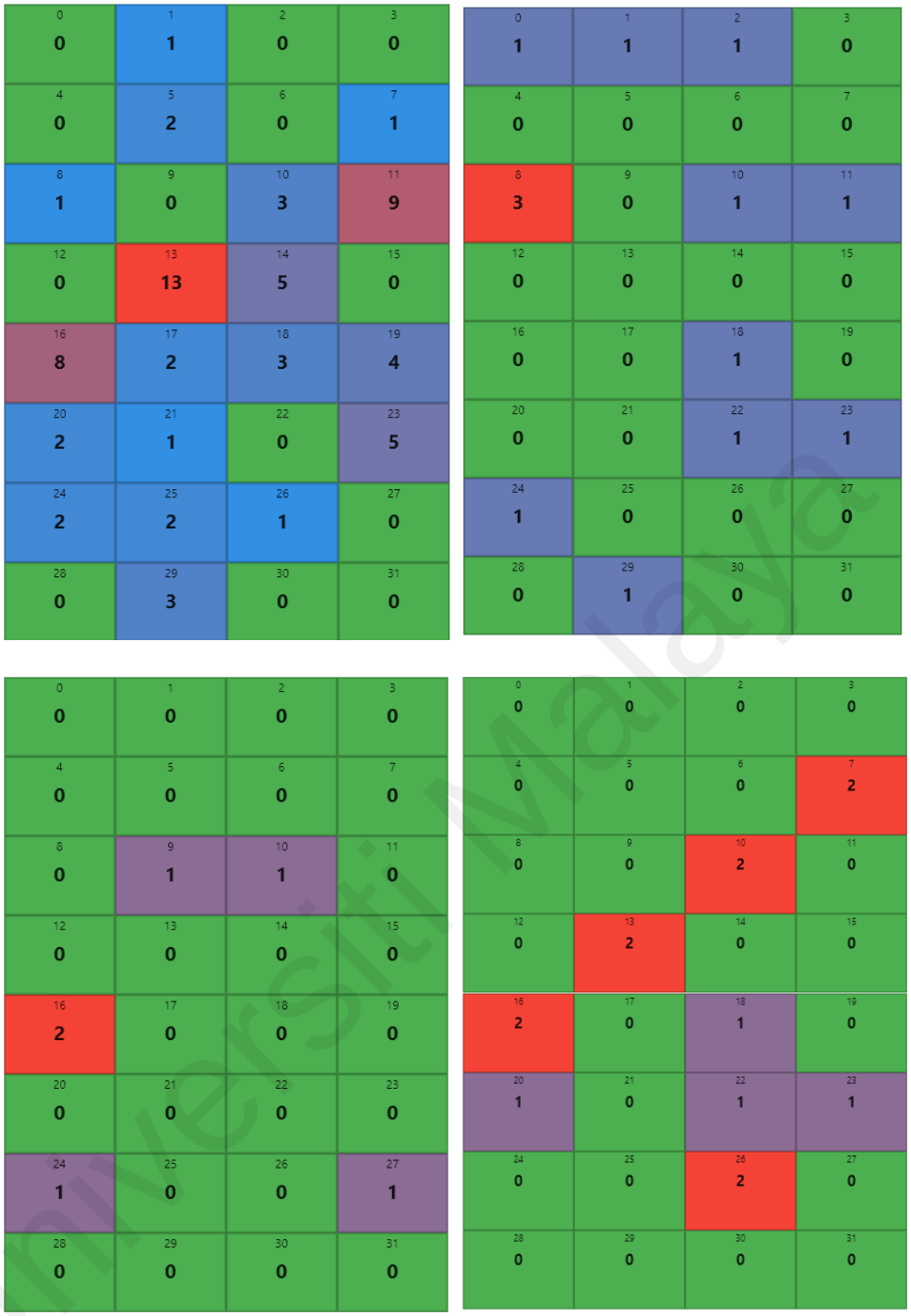
It can be seen from the distribution of the collisions among the site-numbers in different stepping configurations, that the lower the number of dies getting tested simultaneously, the higher the distribution is. It is due to the nature of probing of the wafer, that is the prober moves more with a lower number of simultaneous dies, resulting in engaging more dies should the prober carry a faulty needle. This could be also happening as in the design of the path of the stepping configuration, some of the dies in the wafer are getting tasted by multiple stepping, each time assigned with a different site-number. This, also known as cross stepping makes the pattern of test-induced defects more complicated and complex.



(a) Stepping with 8 site-numbers

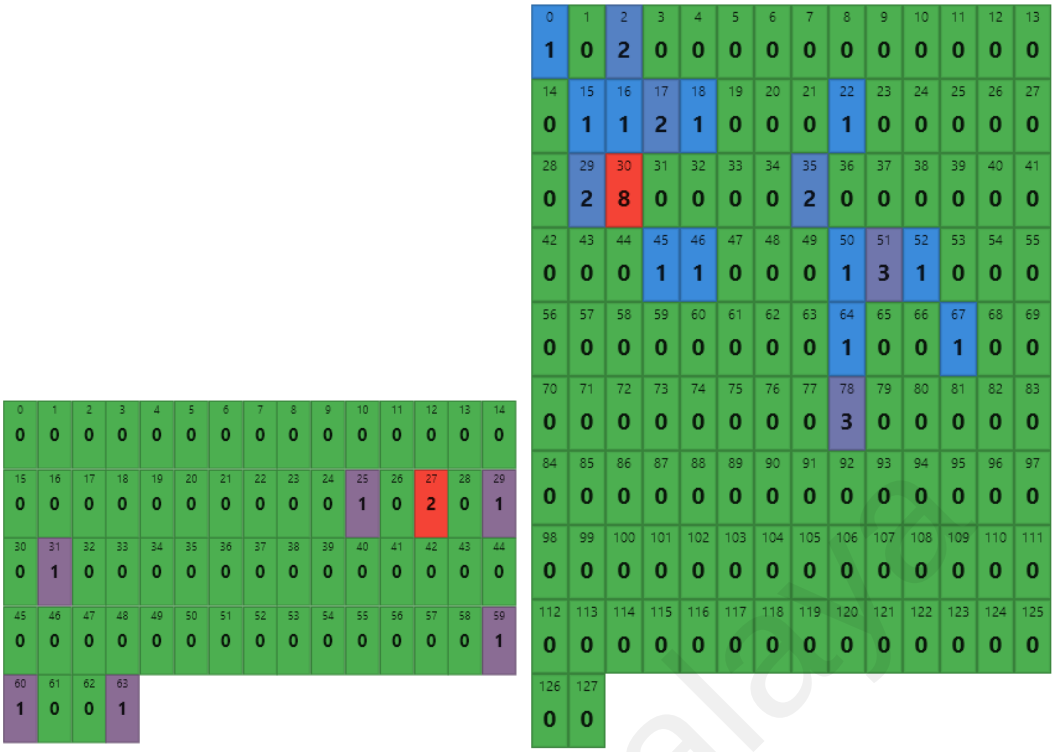
(b) Stepping with 16 site-numbers

Figure 4.4: Example of Root-Cause Heatmap, displaying defect density across different test configurations.



(c) Steppings with 32 site-numbers

Figure 4.3, continued: Example of Root-Cause Heatmap, Example of Root-Cause Heatmap, displaying defect density across different test configurations.



(d) Steppings with high 64 and 128 site-numbers

Figure 4.3, continued: Example of Root-Cause Heatmap, displaying defect density across different test configurations.

4.4 Identifying the Hidden Defects

To evaluate the performance of our algorithm, we employed standard metrics commonly used in classification tasks. Table 4.1 shows the results of each case test for each product category. The accuracy is the ability of the model to correctly offer the coordinates where the hidden defects are there.

Table 4.1: Algorithm Evaluation

Product	Case	Avg. Accuracy
PC-1	Case 1	58%
	Case 2	62%
	Case 3	68%
	Case 4	64%
PC-2	Case 1	50%
	Case 2	58%
	Case 3	64%
	Case 4	61%
PC-3	Case 1	59%
	Case 2	60%
	Case 3	60%
	Case 4	66%

The results from Table 4.1 demonstrate that our proposed algorithm achieves the goal of identifying the hidden test-induced defects for all products. The best performance for products with less than 2000 dies is obtained when the defect weight. W_D is set to 1, the good eight W_G is set to 0.5, and the potential knowledge weight W_P is set to 0.25. This indicates that placing a higher emphasis on the defect edge resulted in better accuracy.

For PC-1, the algorithm's performance improves steadily from Case 1 to Case 3, with accuracy increasing from 58% to 68%, before slightly declining to 64% in Case 4. This suggests that for this product category, assigning higher weights to defective dies while moderately reducing weights for good and potential dies (Case 3) yields the best results.

Accuracy in PC-2 wafers follows a similar trend as PC-1, with accuracy increasing from 50% in Case 1 to a peak of 64% in Case 3. This pattern confirms the effectiveness of the weighting scheme used in Case 3 for products with characteristics of PC-1 and PC-2.

However, that's not the best combination for PC-3. While accuracy marginally improves from Case 1 (59%) to Case 2 (60%), it remains stable in Case 3 before reaching its peak of 66% in Case 4. This suggests that a more extreme weighting scheme favouring defective dies is most effective for PC-3.

Through extensive consultation, a minimum acceptable accuracy threshold of 60% was established for practical implementation in production environments. This threshold was determined based on several factors, including the current manual detection rates, the economic implications of false positives and negatives in semiconductor manufacturing, and the complexity of detecting hidden test-induced defects that often escape conventional inspection methods. The 60% threshold represents a significant improvement over existing manual and automated methods for hidden defect detection while acknowledging the inherent challenges in identifying defects that gradually lose visibility during the testing process.

It is worth to mention that the iteration of the experiment was also studied, and it varied based on the number of sites, as the more site-numbers in a test configuration, result in more potential coordinates and consequently more iterations to reach the convergence.

Figure 4.4 shows one of the experiment results from PC-1. All the possible defects after manual inspection have been plotted in blue colour, and some of the defects already been detected by AOI (visible test-induced defects) are shown in green colour. The remaining which are the hidden defects, are the point of an experiment to see if the algorithm could detect them with a combination of edge weights. This experiment

indicates the better accuracy of case #3 for this product category. Similarly, one of the results from PC-2 and PC-3 are illustrated in Figure 4.5 and Figure 4.6 respectively.

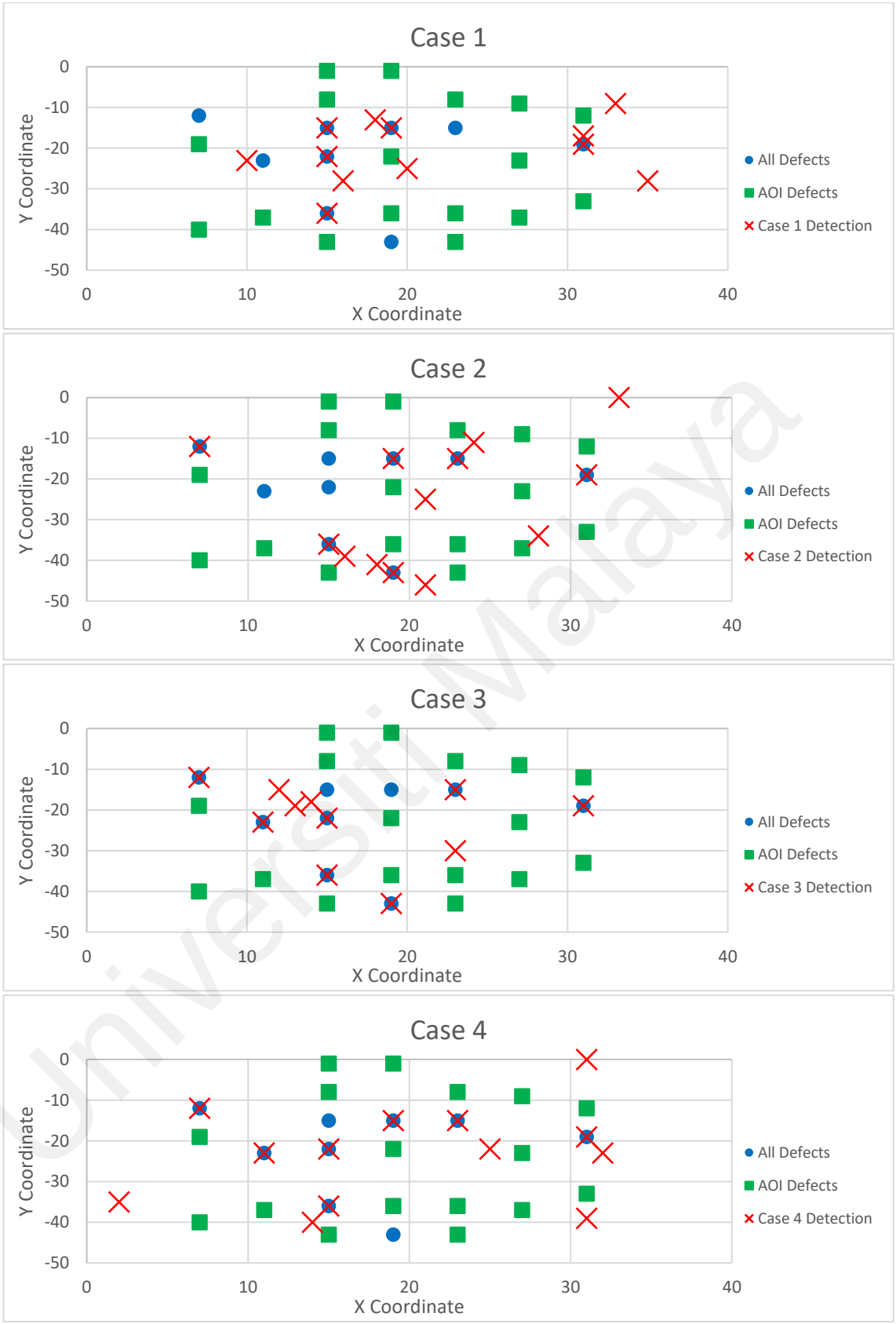


Figure 4.5: Experiment for one of the wafers in PC-1, for all 4 cases

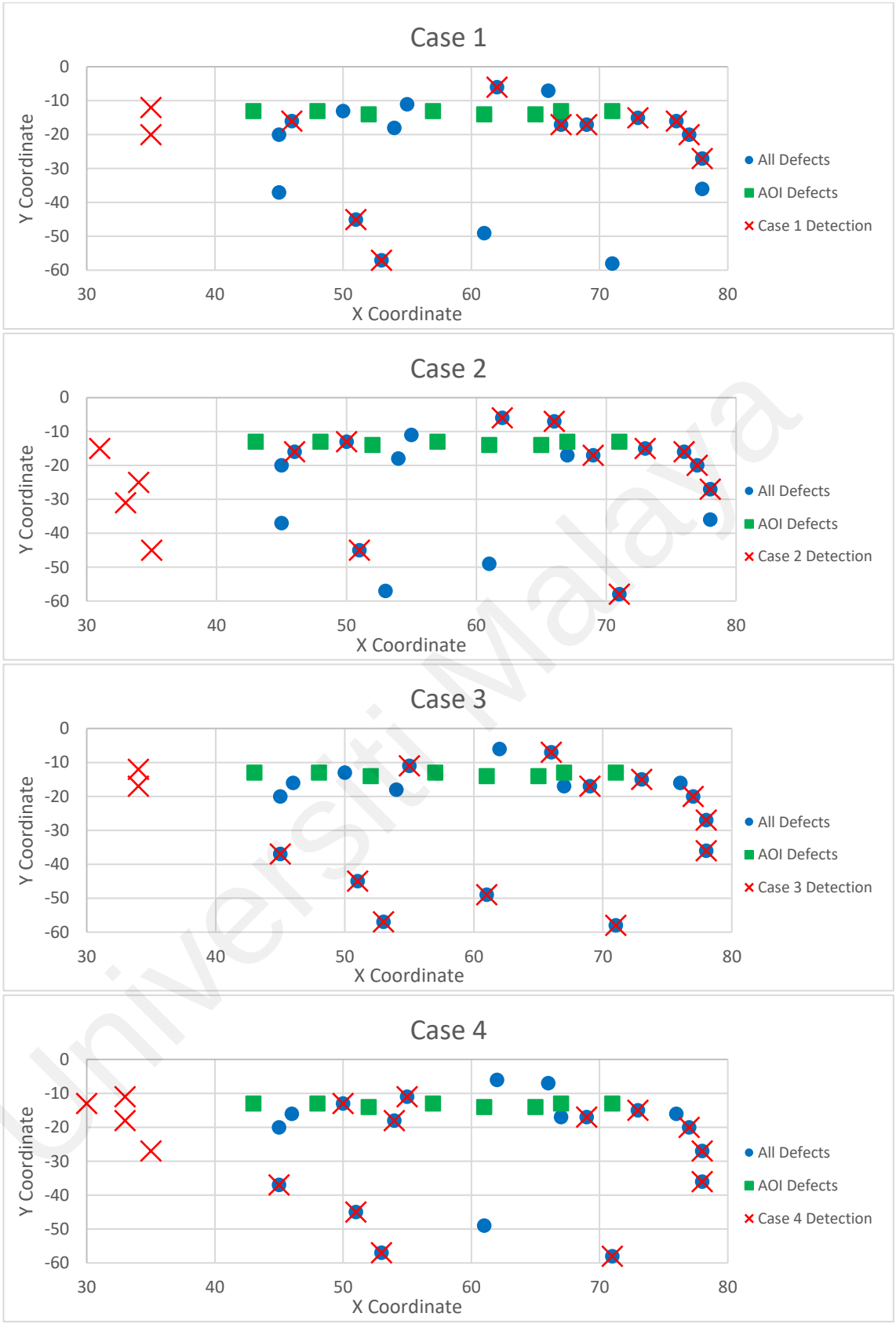


Figure 4.6: Experiment for one of the wafers in PC-2, for all 4 cases

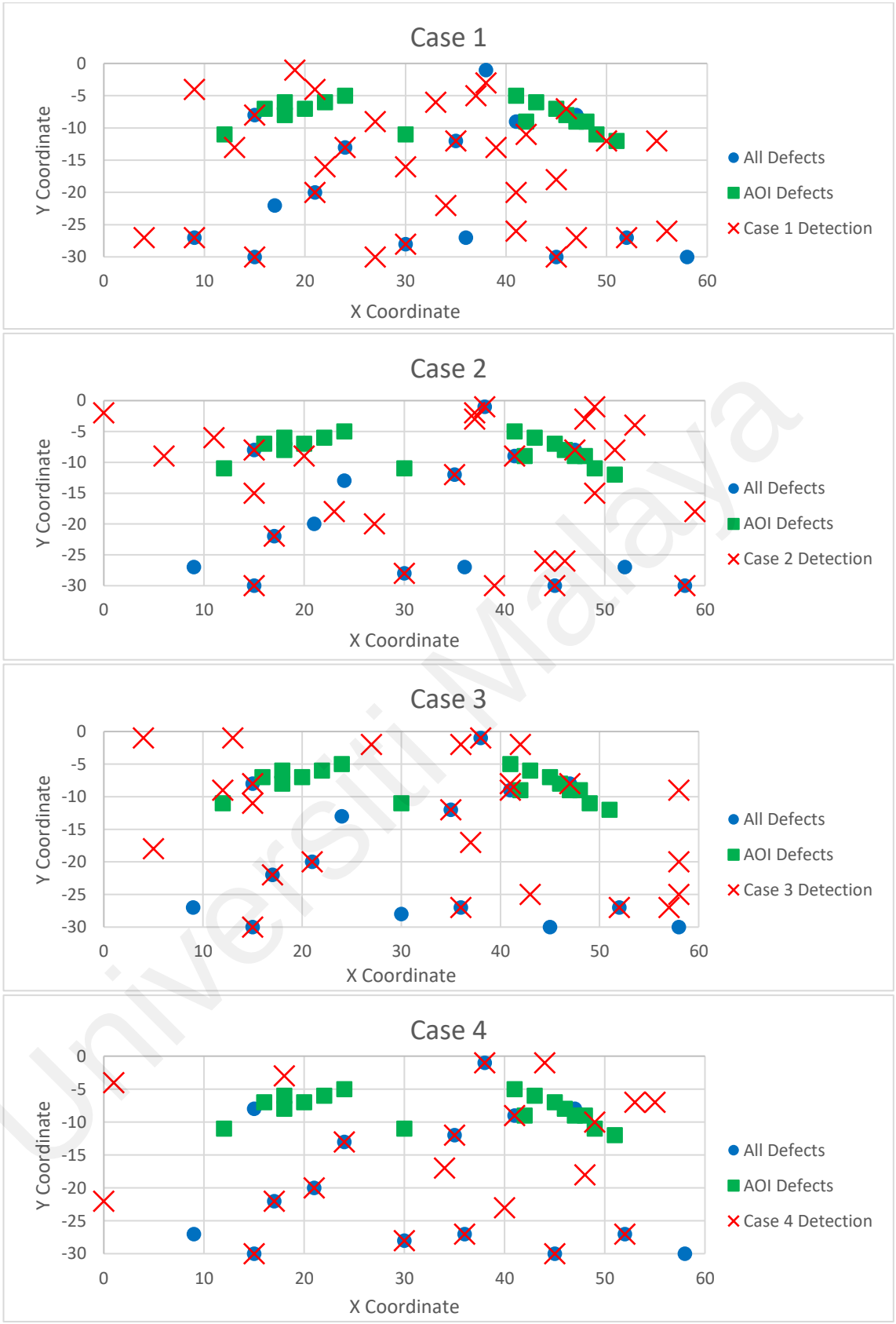


Figure 4.7: Experiment for one of the wafers in PC-3, for all 4 cases

4.5 Error Analysis

While the proposed GSSL algorithm demonstrates promising results in detecting test-induced hidden defects, achieving a maximum accuracy of 68%, our analysis revealed several key scenarios where the method's performance was suboptimal. Understanding these limitations is crucial for both practical implementation and future improvements.

4.5.1 Complex Defect Patterns

The algorithm's performance degrades when confronted with complex test configurations, particularly in scenarios with high site numbers or overlapping stepping patterns. This effect was particularly pronounced in cases where a single die location was assigned to multiple site numbers across different stepping iterations, as illustrated in Figure 12. In this example, the bottom-left region of the wafer shows overlapping steppings, causing a number of dies to be associated with multiple site numbers. To address this limitation, future work could explore more sophisticated methods for calculating and optimizing the weighted edges in the graph model. One potential approach is to incorporate additional information, such as the frequency and temporal order of site number assignments, into the edge weight calculation process. This could help the algorithm better distinguish between the relative importance of multiple site numbers assigned to a single die location.

4.5.2 Physical Probe Card

The physical nature of wafer testing introduces complexities that challenge our detection model. As probe cards degrade during testing, their impact pattern can evolve beyond the primary site number location. For example, in several cases from PC-2, we observed that a damaged probe needle not only affected its designated test site but also

created uncharacteristic defects in neighbouring dies, as it's shown in Figure 13 (c). These secondary defects often lacked the consistent spatial patterns that our algorithm relies upon for detection, leading to missed detections. To mitigate this issue, future work could focus on enhancing the root-cause analysis stage by incorporating temporal information about probe card health. By tracking the degradation of probe cards over time and correlating this information with the observed defect patterns, the algorithm could potentially adapt its detection strategy to account for the evolving impact of damaged probe needles. This could be implemented by maintaining a time-series database of probe card performance metrics (e.g., contact resistance, alignment accuracy) and incorporating these temporal features into the graph.

4.5.3 Novel Defect Types:

The proposed GSSL algorithm relies on defect patterns initially identified by the Automated Optical Inspection (AOI) system. Since test-induced defects caused by the prober typically follow established patterns, this approach works well in most cases. However, proposed method may struggle with novel defect types that the AOI system fails to detect. If a completely new defect type is introduced and not detected by the AOI system, the algorithm will be unable to identify these defects during the initial graph construction stage. To address this limitation, future work could explore the integration of Graph Neural Networks (GNNs) into the algorithm. GNNs can learn rich node representations by aggregating information from neighboring nodes through multiple graph convolution layers. This would enable the detection of previously unseen defect patterns by learning complex spatial and topological features directly from the wafer graph structure. Hence it would allow the model to identify subtle correlations between

defect patterns that may not be apparent through conventional similarity measures. The GNN could be trained in a semi-supervised manner on the existing labeled defects while simultaneously learning to detect anomalous patterns that deviate from known defect distributions. By incorporating GNNs, the algorithm could learn to identify new defect types that the AOI system may miss, adapting to evolving defect patterns and improving its overall detection accuracy.

4.6 Comparative Analysis

In this section, we present a comparative analysis of the proposed method for wafer test-induced hidden defect detection against state-of-the-art techniques in graph-based semi-supervised learning (GSSL) and wafer defect detection. The analysis focuses on two key aspects: (1) test-induced defect detection and root-causing in wafer manufacturing, and (2) graph construction and label propagation in GSSL.

4.6.1 Test-Induced Defect Detection and Root-Causing

The proposed method addresses the limitations of existing techniques by explicitly considering hidden defects with uncertain classifications. By leveraging a graph-based approach, the proposed method enables the detection of hidden defects through the propagation of information from known defective nodes to potential nodes. The edge weighting scheme, based on the defect status, provides informative connections that facilitate the identification of hidden defects. In contrast, the methods proposed in (Cheng et al., 2021) primarily focus on detecting and diagnosing visible test-induced defects, with limited emphasis on hidden defects. The proposed method goes beyond the scope of these techniques by specifically targeting hidden defects and utilizing a graph-based approach to uncover them.

4.6.2 Graph Construction

Table 4.2 shows the comparison between the proposed method and some of the techniques in this context mentioned in literature review.

Table 4.2: Comparison of Graph Construction Methods

Method	Label Information	Graph Type	Topology-aware	Adaptive Weighting	Prior Knowledge Incorporation
Proposed Method	Supervised	Weighted	Yes	Yes	Root-cause Analysis
KNN	Unsupervised	Unweighted	No	No	No
b-Matching	Unsupervised	Unweighted	No	No	No
LNP	Unsupervised	Weighted	No	No	No
GBILI	Supervised	Weighted	No	No	Label Information
RGCLI	Supervised	Weighted	No	No	Label Information

The proposed method distinguishes itself from other graph construction techniques by leveraging supervised label information, domain-specific knowledge, and an adaptive weighting mechanism to create a weighted, topology-aware graph tailored to the semiconductor wafer hidden defect detection problem. While KNN-based, b-Matching, and LNP methods are unsupervised and construct graphs based on general approaches such as k-nearest neighbours, optimization, or linear neighbourhood, the proposed method incorporates label information (0, 0.5, 1) and generates edges based on site number, which is specific to the semiconductor wafer domain. GBILI and RGCLI utilize label information but do not incorporate domain-specific weights. In contrast, the proposed method assigns weights (w_D, w_G, w_P) to the edges and adapts these weights during the iterative label propagation process to optimize the graph structure for accurate

hidden defect detection. Furthermore, the proposed method is topology-aware, considering the spatial arrangement of dies on the wafer, and incorporates prior knowledge from root-cause analysis to guide the graph construction process.

4.6.3 Label Propagation

The proposed method for identifying hidden defects in semiconductor wafers achieves a time complexity of $O(k|E|)$, where k is the number of iterations and $|E|$ is the number of edges in the graph. This complexity arises from the iterative label propagation process, which updates each node's label based on its neighbours' weighted average in each iteration. The algorithm's efficiency stems from its targeted approach, focusing only on suspicious site numbers and leveraging the typically sparse nature of wafer graphs. A comparison between proposed method and the traditional learning approaches is shown by Table 4.3.

Table 4.3: Comparison of Label Propagation Methods

Method	Learning Setting	Label Clamping	Time Complexity	Targeted Propagation	Per-Node Convergence	Weighted Propagation
Proposed Method	Transductive	Partial (good/ defective)	$O(k\|E\|)$	Yes	Yes	Yes
GRF	Transductive	Yes	$O(\|V\|^2)$	No	No	Yes
LGC	Transductive	No	$O(\|V\|^2)$	No	No	Yes
SLP	Transductive	No	$O(\|V\|^2)$	No	No	Yes

In contrast to our method, traditional graph-based semi-supervised learning approaches such as GRF, LGC, and SLP exhibit a time complexity of $O(|V|^2)$,where $|V|$ is the number of nodes, due to their matrix operations on the full graph. The proposed method's lower complexity translates to significant performance gains for large wafers,

especially when $|E| << |V|^2$. Beyond computational efficiency, the method introduces several key advantages: targeted propagation, per-node convergence checks, and domain-specific weighted propagation (w_D, w_G, w_P). These features, absent in GRF, LGC, and SLP, allow for more nuanced and context-aware label propagation. While maintaining a Transductive nature like its counterparts, the proposed approach's adaptive weighting scheme incorporates crucial domain knowledge about semiconductor defects, in contrast to the uniform weighting in traditional methods.

Table 4.4: Quantitative Comparison of Label Propagation Methods

Method	Product Category	Avg. Accuracy	Avg. Sensitivity
Proposed Method	PC-1	68%	64%
	PC-2	64%	63%
	PC-3	66%	64%
GRF	PC-1	46%	55%
	PC-2	46%	53%
	PC-3	44%	54%
LGC	PC-1	58%	60%
	PC-2	56%	57%
	PC-3	55%	58%
SLP	PC-1	60%	60%
	PC-2	55%	58%
	PC-3	61%	66%

Table 4.4 presents a quantitative comparison of the proposed label propagation method against other methods mentioned in Table 4.3 across three product categories (PC-1, PC-2, and PC-3). The performance is evaluated using two metrics of average accuracy and average sensitivity. The proposed method consistently outperforms the other approaches in terms of average accuracy, achieving 68%, 64%, and 66% for PC-1, PC-2, and PC-3, respectively. This demonstrates the effectiveness of the proposed method in correctly

identifying test-induced hidden defects across various product categories. While the proposed method maintains a competitive edge in average sensitivity, ranging from 63% to 64%, it is worth noting that SLP exhibits slightly higher sensitivity for PC-3 at 66%. This suggests that SLP may be more effective in detecting true positive hidden defects for this specific product category. However, considering the overall performance across all product categories and metrics, the proposed method demonstrates a clear advantage, striking a balance between accuracy and sensitivity.

The comparative analysis highlights the novelty and effectiveness of the proposed method in addressing the challenges of wafer test-induced hidden defect detection. By incorporating domain-specific knowledge, focusing on potential nodes with uncertain classifications, and leveraging a graph-based approach with informative edge weights, the proposed method offers a solution for detecting hidden defects in wafer testing. This approach has the potential to significantly improve the accuracy and efficiency of wafer defect detection and root-causing, ultimately leading to enhanced yield and quality in semiconductor manufacturing.

4.7 Solution Implementation

For the purpose of data integration, pre-processing, and detection of visible test-induced defects, root-causing and identification of hidden defects, a stand-alone software has been created from scratch using Flutter framework which is an open-source framework by Google for building natively compiled, multi-platform applications from a single codebase, based on Dart language. Figure 4.7 illustrates the high-level components and their relationship within the architecture of the developed software.

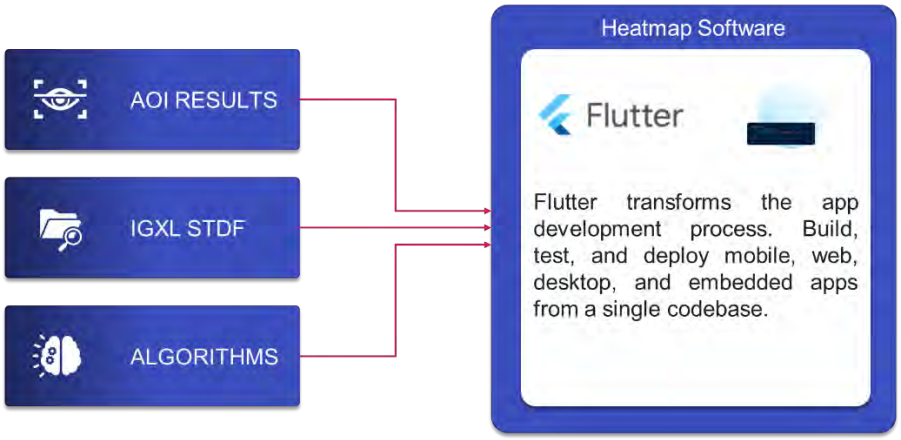


Figure 4.8: Basic Structural Representation of Developed Software

4.7.1 Pre-Processing in Developed Software

Figure 4.8 shows aligning confirmation developed within the software, for AOI and Physical Test Data together with rotation and translation along with offset. As can be seen from Figure 4.8, the wafer map generated by AOI and Physical Test Data does not match the same coordinates, even after rotation and translation. Therefore, it's been designed to have an extra offset to adjust the wafer maps together.

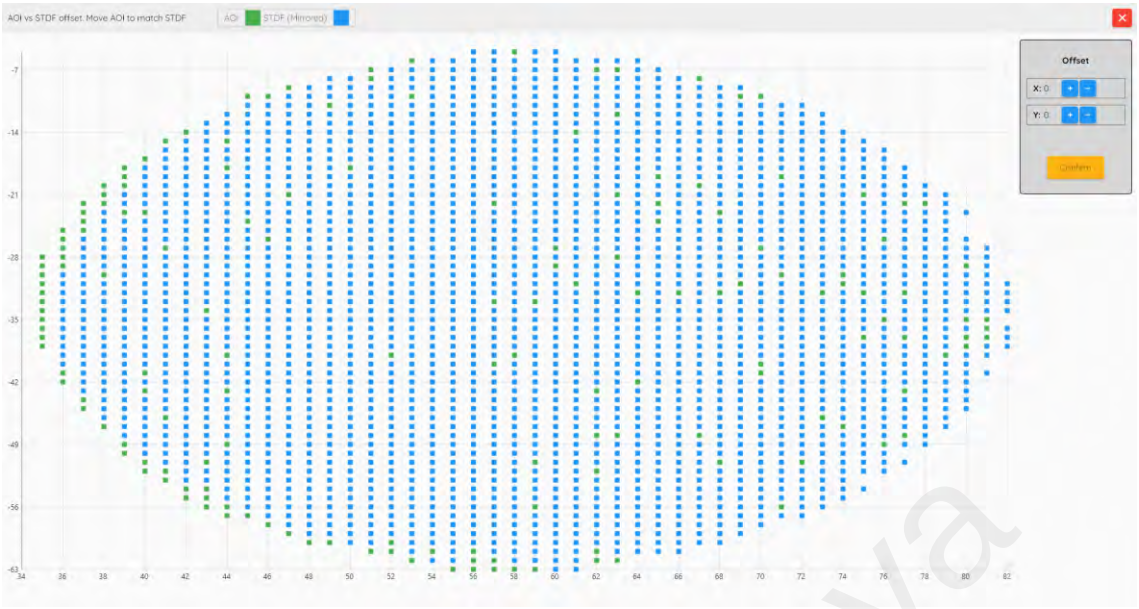


Figure 4.9: Alignment Confirmation Between AOI and Physical Test Data

4.7.2 Root-Causing and Hidden Defect Identification in Developed Software

In the developed software, engineers can view a predicted list of potential defects that are induced by the test process, alongside with root-causing of the faulty site-number. The only requirement for analyzing and visualizing is to load the AOI report from the server and select the respective Physical Test Data report.

Figure 4.9 presents a screenshot of the developed software, where the engineers are enabled to have the quick look on the root-cause analysis on the left, and the identified coordinates of potential defects on the right, all within few seconds of loading AOI report.

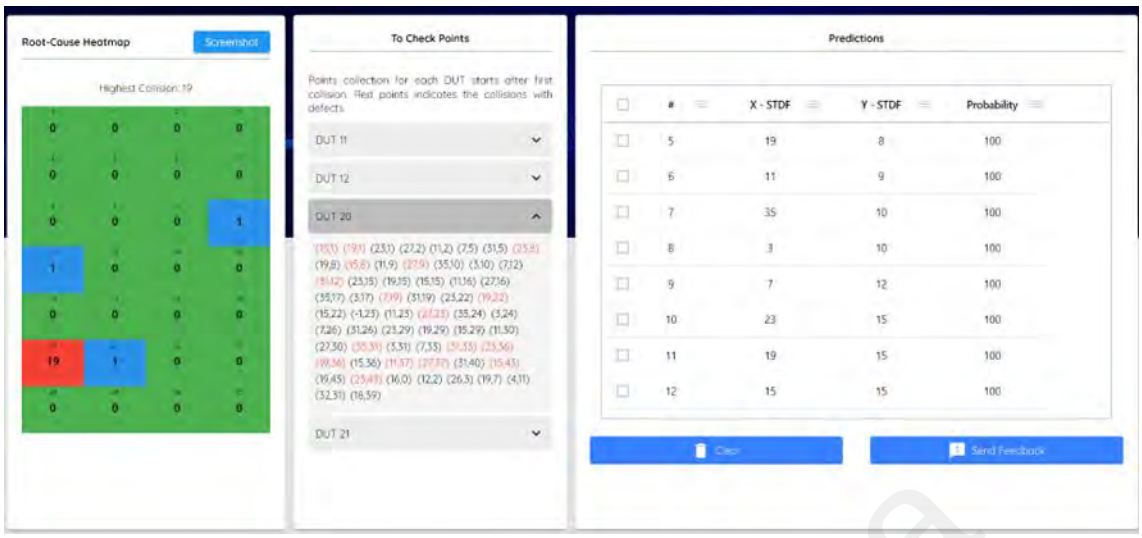


Figure 4.10: Screenshot of Developed Software for Root-Causing and Identification of Hidden Defects

4.8 Extending to Other Domains and Applications

The algorithm's fundamental approach to pattern recognition and defect detection shows promise for adaptation to various manufacturing and quality control scenarios. Future work should investigate the algorithm's applicability to different industries and inspection challenges, such as electronic component manufacturing, printed circuit board inspection, or precision engineering. This research direction would focus on enhancing the algorithm's flexibility to handle diverse defect types, varying spatial patterns, and different inspection requirements while maintaining its core advantages in semi-supervised learning and minimal labelling requirements.

CHAPTER 5: CONCLUSION

5.1 Conclusion

In conclusion, this study has successfully fulfilled its primary objectives in addressing the challenge of detecting test-induced hidden defects in semiconductor wafers. First, through a comprehensive investigation of the relationship between defect occurrence and test configuration, the research unveiled underlying patterns that contribute to the manifestation of hidden defects. This insight laid the foundation for the development of a novel graph-based semi-supervised learning (GSSL) algorithm.

Second, the machine learning-based GSSL method demonstrated an average accuracy exceeding 60% in predicting the precise location of hidden defects across multiple semiconductor wafer products. By leveraging wafer inspection data and incorporating domain-specific knowledge, the algorithm constructs a graph representation of the wafer, enabling the identification of hidden defects through an iterative label propagation process.

Finally, the study optimized the graph-based algorithm to improve accuracy and reduce time complexity in hidden defect localization. The algorithm's effectiveness was validated through extensive experimentation, showcasing its potential to enhance yield management and quality control processes in the semiconductor industry.

By successfully addressing these objectives, this innovative approach provides a valuable tool for semiconductor manufacturers to improve production efficiency, minimize yield losses, and deliver high-quality products to their customers. The developed GSSL algorithm represents a significant advancement in the field of

semiconductor defect detection and paves the way for future research and industrial applications.

5.2 Future Work

As the enhanced GSSL approach with domain-based graph construction and iterative weighted label propagation demonstrated promising results in detecting hidden test-induced defects, further work on this section can be explored. Future work involves exploring the recent advancements of neural networks in GSSL to detect test-induced hidden defects at higher levels. Neural network architectures have the capacity to learn representations of entire graphs, which could enable the extension of current work beyond individual wafer analysis. This approach has the potential to identify defects across multiple wafers or even entire production batches, thereby increasing manufacturing productivity and yield.

Additionally, the enhanced GSSL approach proposed in this work could be extended to other semiconductor manufacturing applications, such as fabrication defect classification and wafer map pattern recognition. Applying GSSL with optimized graph construction to these related tasks may yield performance improvements by effectively utilizing both labelled and unlabelled wafer data.

REFERENCES

Ankerst, M., Breunig, M. M., Kriegel, H.-P., & Sander, J. (1999). *OPTICS: ordering points to identify the clustering structure* Proceedings of the 1999 ACM SIGMOD international conference on Management of data, Philadelphia, Pennsylvania, USA. https://doi.org/10.1145/304182.304187

Baly, R., & Hajj, H. (2012). Wafer Classification Using Support Vector Machines. *IEEE Transactions on Semiconductor Manufacturing*, *25*(3), 373-383. https://doi.org/10.1109/TSM.2012.2196058

Berton, L., de Paulo Faleiros, T., Valejo, A., Valverde-Rebaza, J., & de Andrade Lopes, A. (2017). RGCLI: Robust Graph that Considers Labeled Instances for Semi-Supervised Learning. *Neurocomputing*, *226*, 238-248. https://doi.org/https://doi.org/10.1016/j.neucom.2016.11.053

Berton, L., & Lopes, A. D. A. (2014, 24-28 Aug. 2014). Graph Construction Based on Labeled Instances for Semi-supervised Learning. 2014 22nd International Conference on Pattern Recognition,

Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., & Vandergheynst, P. (2017). Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, *34*(4), 18-42. https://doi.org/10.1109/MSP.2017.2693418

Chen, L. L. Y., Li, K. S.-M., Jiang, X. H., Wang, S. J., Huang, A. Y. A., Chen, J. E., Liang, H. C., & Hsu, C. L. (2021, 10-15 Oct. 2021). Semi-Supervised Framework for Wafer Defect Pattern Recognition with Enhanced Labeling. 2021 IEEE International Test Conference (ITC),

Chen, S., Zhang, Y., Yi, M., Ma, J., & Hou, X. (2021, 4-6 June 2021). Wafer maps defect recognition based on transfer learning of handwritten pre-training network. 2021 International Symposium on Computer Technology and Information Science (ISCTIS),

Cheng, K. C. C., Chen, L. L. Y., Li, J. W., Li, K. S. M., Tsai, N. C. Y., Wang, S. J., Huang, A. Y. A., Chou, L., Lee, C. S., Chen, J. E., Liang, H. C., & Hsu, C. L. (2021). Machine Learning-Based Detection Method for Wafer Test Induced Defects. *IEEE Transactions on Semiconductor Manufacturing*, *34*(2), 161-167. https://doi.org/10.1109/TSM.2021.3065405

Cheng, K. C. C., Li, K. S.-M., Huang, A. Y. A., Li, J. W., Chen, L. L. Y., Tsai, N. C.-Y., Wang, S. J., Lee, C. S., Chou, L., Liao, P. Y. Y., Liang, H. C., & Chen, J. E. (2020, 9-13 March 2020). Wafer-Level Test Path Pattern Recognition and Test Characteristics for Test-Induced Defect Diagnosis. 2020 Design, Automation & Test in Europe Conference & Exhibition (DATE),

Chong, Y., Ding, Y., Yan, Q., & Pan, S. (2020). Graph-based semi-supervised learning: A review. *Neurocomputing*, *408*, 216-230. https://doi.org/https://doi.org/10.1016/j.neucom.2019.12.130

Cognex. (2024). *Semiconductor wafer defect inspection*. Retrieved 1/7/2024 from https://www.cognex.com/industries/electronics/semiconductors/wafer-defect-inspection

Djouzi, K., & Beghdad-Bey, K. (2019, 26-27 June 2019). A Review of Clustering Algorithms for Big Data. 2019 International Conference on Networking and Advanced Systems (ICNAS),

Ebayyeh, A. A. R. M. A., & Mousavi, A. (2020). A Review and Analysis of Automatic Optical Inspection and Quality Monitoring Methods in Electronics Industry. *IEEE Access*, *8*, 183192-183271. https://doi.org/10.1109/ACCESS.2020.3029127

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Knowledge Discovery and Data Mining,

Ezugwu, A. E., Ikotun, A. M., Oyelade, O. O., Abualigah, L., Agushaka, J. O., Eke, C. I., & Akinyelu, A. A. (2022). A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. *Engineering Applications of Artificial Intelligence*, *110*, 104743. https://doi.org/https://doi.org/10.1016/j.engappai.2022.104743

Geng, H., Yang, F., Zeng, X., & Yu, B. (2021, 1-4 Nov. 2021). When Wafer Failure Pattern Classification Meets Few-shot Learning and Self-Supervised Learning. 2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD),

Hara, Y., Doi, H., Karasaki, K., & Iida, T. (1988). A system for PCB automated inspection using fluorescent light. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *10*(1), 69-78. https://doi.org/10.1109/34.3868

Huang, C.-J. (2007). Clustered defect detection of high quality chips using self-supervised multilayer perceptron. *Expert Syst. Appl.*, *33*, 996-1003. https://doi.org/10.1016/j.eswa.2006.07.011

Huang, S.-H., & Pan, Y.-C. (2015). Automated visual inspection in the semiconductor industry: A survey. *Computers in Industry*, *66*, 1-10. https://doi.org/https://doi.org/10.1016/j.compind.2014.10.006

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for Clustering Data*. Prentice Hall. https://books.google.com.my/books?id=7eBQAAAAMAAJ

Jebara, T., Wang, J., & Chang, S. (2009). Graph construction and b-matching for semi-supervised learning. *Proceedings of the 26th International Conference On Machine Learning, ICML 2009*. https://doi.org/10.1145/1553374.1553432

Jin, C. H., Na, H. J., Piao, M., Pok, G., & Ryu, K. H. (2019). A Novel DBSCAN-Based Defect Pattern Detection and Classification Framework for Wafer Bin Map. *IEEE Transactions on Semiconductor Manufacturing*, *32*(3), 286-292. https://doi.org/10.1109/TSM.2019.2916835

Kim, Y., Cho, D., & Lee, J.-H. (2021). Wafer defect pattern classification with detecting out-of-distribution. *Microelectronics Reliability*, *122*, 114157. https://doi.org/https://doi.org/10.1016/j.microrel.2021.114157

Li, K. S. M., Liao, P. Y. Y., Cheng, K. C. C., Chen, L. L. Y., Wang, S. J., Huang, A. Y. A., Chou, L., Han, G. C. H., Chen, J. E., Liang, H. C., & Hsu, C. L. (2021). Hidden Wafer Scratch Defects Projection for Diagnosis and Quality Enhancement. *IEEE Transactions on Semiconductor Manufacturing*, *34*(1), 9-16. https://doi.org/10.1109/TSM.2020.3040998

Li, R., & Kang, Z. (2023). *Deep Learning for Wafer Map Defect Detection: A Review*. https://doi.org/10.1109/PHM-Hangzhou58797.2023.10482800

Li, Y., & Wang, J. (2021). *A Defect Detection Method Based on Improved Mask R-CNN for Wafer Maps*. https://doi.org/10.1109/ICCNEA53019.2021.00038

Ma, J., Zhang, T., Yang, C., Cao, Y., Xie, L., Tian, H., & Li, X. (2023). Review of Wafer Surface Defect Detection Methods. *Electronics*, *12*(8).

Mahdi, M. A., Hosny, K. M., & Elhenawy, I. (2021). Scalable Clustering Algorithms for Big Data: A Review. *IEEE Access*, *9*, 80015-80027. https://doi.org/10.1109/ACCESS.2021.3084057

Naeem, S., Ali, A., Anam, S., & Ahmed, M. (2023). An Unsupervised Machine Learning Algorithms: Comprehensive Review. *IJCDS Journal*, *13*, 911-921. https://doi.org/10.12785/ijcds/130172

Nakazawa, T., & Kulkarni, D. V. (2018). Wafer Map Defect Pattern Classification and Image Retrieval Using Convolutional Neural Network. *IEEE Transactions on Semiconductor Manufacturing*, *31*(2), 309-314. https://doi.org/10.1109/TSM.2018.2795466

Nie, F., Xiang, S., Liu, Y., & Zhang, C. (2010). A general graph-based semi-supervised learning with novel class discovery. *Neural Computing and Applications*, *19*(4), 549-555. https://doi.org/10.1007/s00521-009-0305-8

Quirk, M., & Serda, J. (2001). *Semiconductor Manufacturing Technology*. Prentice Hall. https://books.google.com.my/books?id=O2eGQgAACAAJ

Saqlain, M., Abbas, Q., & Lee, J. Y. (2020). A Deep Convolutional Neural Network for Wafer Defect Identification on an Imbalanced Dataset in Semiconductor Manufacturing Processes. *IEEE Transactions on Semiconductor Manufacturing*, *33*(3), 436-444. https://doi.org/10.1109/TSM.2020.2994357

Saqlain, M., Jargalsaikhan, B., & Lee, J. Y. (2019). A Voting Ensemble Classifier for Wafer Map Defect Patterns Identification in Semiconductor Manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, *32*(2), 171-182. https://doi.org/10.1109/TSM.2019.2904306

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2009). The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, *20*(1), 61-80. https://doi.org/10.1109/TNN.2008.2005605

Shankar, N. G., & Zhong, Z. W. (2005). Defect detection on semiconductor wafer surfaces. *Microelectronic Engineering*, *77*(3), 337-346. https://doi.org/https://doi.org/10.1016/j.mee.2004.12.003

Shinde, P. P., Pai, P. P., & Adiga, S. P. (2022). Wafer Defect Localization and Classification Using Deep Learning Techniques. *IEEE Access*, *10*, 39969-39974. https://doi.org/10.1109/ACCESS.2022.3166512

Song, Z., Yang, X., Xu, Z., & King, I. (2023). Graph-Based Semi-Supervised Learning: A Comprehensive Review. *IEEE Transactions on Neural Networks and Learning Systems*, *34*(11), 8174-8194. https://doi.org/10.1109/TNNLS.2022.3155478

Srivastava, S. (2021). Study of gas sensor detection for NOX Gas: A review. *Materials Today: Proceedings*, *37*, 3709-3712. https://doi.org/https://doi.org/10.1016/j.matpr.2020.10.161

Theodosiou, T., Rapti, A., Papageorgiou, K., Tziolas, T., Papageorgiou, E., Dimitriou, N., Margetis, G., & Tzovaras, D. (2023). A Review Study on ML-based Methods for Defect-Pattern Recognition in Wafer Maps. *Procedia Computer Science*, *217*, 570-583. https://doi.org/https://doi.org/10.1016/j.procs.2022.12.253

Tsai, T. H., & Lee, Y. C. (2020). A Light-Weight Neural Network for Wafer Map Classification Based on Data Augmentation. *IEEE Transactions on Semiconductor Manufacturing*, *33*(4), 663-672. https://doi.org/10.1109/TSM.2020.3013004

van Engelen, J. E., & Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine Learning*, *109*(2), 373-440. https://doi.org/10.1007/s10994-019-05855-6

Wang, F., & Zhang, C. (2008). Label Propagation through Linear Neighborhoods. *IEEE Transactions on Knowledge and Data Engineering*, *20*(1), 55-67. https://doi.org/10.1109/TKDE.2007.190672

Wang, J., Xu, C., Yang, Z., Zhang, J., & Li, X. (2020). Deformable Convolutional Networks for Efficient Mixed-Type Wafer Defect Pattern Recognition. *IEEE Transactions on Semiconductor Manufacturing*, *33*(4), 587-596. https://doi.org/10.1109/TSM.2020.3020985

Wu, M. J., Jang, J. S. R., & Chen, J. L. (2015). Wafer Map Failure Pattern Recognition and Similarity Ranking for Large-Scale Data Sets. *IEEE Transactions on Semiconductor Manufacturing*, *28*(1), 1-12. https://doi.org/10.1109/TSM.2014.2364237

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2021). A Comprehensive Survey on Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(1), 4-24. https://doi.org/10.1109/TNNLS.2020.2978386

Xie, L., Huang, R., Gu, N., & Cao, Z. (2014). A novel defect detection and identification method in optical inspection. *Neural Computing and Applications*, *24*(7), 1953-1962. https://doi.org/10.1007/s00521-013-1442-7

Y C a, P., Pulabaigari, V., & B, E. (2018). Semi-supervised learning: a brief review. *International Journal of Engineering & Technology*, *7*, 81. https://doi.org/10.14419/ijet.v7i1.8.9977

Yu, N., Chen, H., Xu, Q., & Hasan, M. M. (2021, 22-24 Oct. 2021). Wafer map defect recognition with few shot learning based on hybrid self-attention mechanism and prototype network. 2021 China Automation Congress (CAC),

Zhou, D., Bousquet, O., Lal, T., Weston, J., & Olkopf, B. (2004). Learning with Local and Global Consistency. *Advances in Neural Information Processing Systems 16*, *16*.

Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., & Sun, M. (2018). *Graph Neural Networks: A Review of Methods and Applications*. https://doi.org/10.48550/arXiv.1812.08434

Zhu, X., Ghahramani, Z., & Lafferty, J. (2003). *Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions* (Vol. 3).