

การตรวจรูปแบบการเสียของแผ่นบันทึกข้อมูลโดยใช้การประมวลผลภาพ  
และโครงข่ายประสาทเทียม

นายสิริศักดิ์ เลียงวงษ์สันต์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชาแมคคาทรอนิกส์  
มหาวิทยาลัยเทคโนโลยีสุรนารี  
ปีการศึกษา 2555

**DETECTION OF MEDIA DEFECT MODE BY USING  
IMAGE PROCESSING AND NEURAL NETWORK**

**Sirisak Liangwongsan**

**A Thesis Submitted in Partial Fulfillment of the Requirements for the**

**Degree of Master of Engineering in Mechatronics**

**Suranaree University of Technology**

**Academic Year 2012**

การตรวจรูปแบบการเสียของแผ่นบันทึกข้อมูลโดยใช้การประมวลผลภาพ  
และโครงข่ายประสาทเทียม

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้บัณฑิตวิทยาลัยฉบับนี้เป็นส่วนหนึ่งของการศึกษา  
ตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

\_\_\_\_\_

(ผศ. ดร.จิระพล ศรีเสวีผล)

ประธานกรรมการ

\_\_\_\_\_

(ผศ. ดร.อนันต์ อุ่นศิริไพลย์)

กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)

\_\_\_\_\_

(ผศ. ดร.เผด็จ เผ่าละออ)

กรรมการ

\_\_\_\_\_

(ศ. ดร.ชูกิจ ลิมปิจำนงค์)

รองอธิการบดีฝ่ายวิชาการ

\_\_\_\_\_

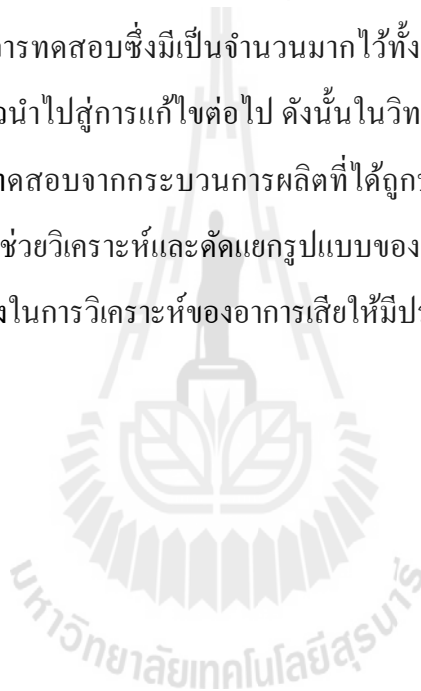
(รศ. ร.อ. ดร.กนต์ธร ชำนิประศาสน์)

คณบดีสำนักวิชาวิศวกรรมศาสตร์

ลิขสิทธิ์ เลียงวงษ์สันต์ : การตรวจรูปแบบการเสียบของแผ่นบันทึกข้อมูลโดยใช้การประมวลผลภาพและโครงข่ายประสาทเทียม (DETECTION OF MEDIA DEFECT MODE BY USING IMAGE PROCESSING AND NEURAL NETWORK)

อาจารย์ที่ปรึกษา : ผู้ช่วยศาสตราจารย์ ดร.อนันท์ อุ่นศิริลัย, 101 หน้า.

อุปกรณ์หนึ่งที่เป็นส่วนสำคัญของคอมพิวเตอร์คือ ฮาร์ดดิสก์ (Hard Disk) ซึ่งเป็นอุปกรณ์จัดเก็บข้อมูลหลักของคอมพิวเตอร์ โดยในกระบวนการผลิตฮาร์ดดิสก์แต่ละตัวนั้นจะต้องผ่านขั้นตอนการผลิตและกระบวนการทดสอบที่มีความยุ่งยากซับซ้อน โดยในแต่ละขั้นตอนจะมีการเก็บบันทึกข้อมูลที่ได้มาจากการทดสอบซึ่งมีเป็นจำนวนมากไว้ทั้งหมดเพื่อนำมาวิเคราะห์สาเหตุของการเสียบของฮาร์ดดิสก์แล้วนำไปสู่การแก้ไขต่อไป ดังนั้นในวิทยานิพนธ์นี้ได้นำเสนอแนวทางการนำข้อมูลที่ได้มาจากการทดสอบจากกระบวนการผลิตที่ได้ถูกบันทึกเอาไว้ด้วยการนำเทคนิคการประมวลผลภาพมาใช้เพื่อช่วยวิเคราะห์และคัดแยกรูปแบบของการเกิด defect บนแผ่นบันทึกข้อมูลเป็นการเพิ่มแนวทางในการวิเคราะห์ของอาการเสียบให้มีประสิทธิภาพมากยิ่งขึ้น



สาขาวิชา วิศวกรรมเครื่องกล

ปีการศึกษา 2555

ลายมือชื่อนักศึกษา \_\_\_\_\_

ลายมือชื่ออาจารย์ที่ปรึกษา \_\_\_\_\_

SIRISAK LIANGWONGSAN : DETECTION OF MEDIA DEFECT MODE  
BY USING IMAGE PROCESSING AND NEURAL NETWORK.

THESIS ADVISOR : ASST. PROF. ANANT OONSIVILAI, Ph.D., 101 PP.

DEFECT/FAILURE ANALYSIS

One importance component of computer is Hard disk drive. This component is main data storage in computer. In hard disk industry, the hard disk passed various complex process and tested system. In each step, it will generates a large data and recode them during test process. This data is useful for failure analysis, find root case of failure and solve this problem. In this paper we propose the Image processing for help to analysis and separate defect mode on media disks by use data form record of result after tested process this method will help to increase speed and accuracy in failure analysis task.

School of Mechanical Engineering

Academic Year 2012

Student's Signature \_\_\_\_\_

Advisor's Signature \_\_\_\_\_

## กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงตามวัตถุประสงค์ทุกประการ ผู้วิจัยขอขอบพระคุณบุคคลต่าง ๆ ที่ให้คำปรึกษา แนะนำ และช่วยเหลืออย่างดียิ่ง ทั้งด้านวิชาการและด้านการดำเนินงานวิจัย ดังนี้

ขอขอบคุณผู้ช่วยศาสตราจารย์ ดร.อนันท์ อุ้นศิริไทย์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้ความรู้และให้คำปรึกษาแนะนำดี ๆ และคอยผลักดันส่งเสริมการทำวิจัย รวมทั้งให้คำแนะนำในการดำเนินชีวิตด้วยความเมตตาโดยตลอดและสละเวลาตรวจสอบแก้ไขให้วิทยานิพนธ์มีความถูกต้องสมบูรณ์มากยิ่งขึ้น

ขอขอบคุณ พี่สุวัฒน์ สวัสดิกุล และพี่รุจิโรจน์ ทิพย์มนตรี ที่ให้โอกาสในการศึกษาและคำแนะนำดี ๆ ทั้งในด้านการงานและการดำเนินชีวิต

ขอขอบคุณพี่ๆ เพื่อนๆ บัณฑิตศึกษาทุกท่านที่ให้คำแนะนำในด้านต่าง ๆ และยังคงคอยให้กำลังใจมาโดยตลอด

สุดท้ายนี้ขอกราบขอบพระคุณอาจารย์ทุกท่านที่ประสิทธิ์ประสาทวิชาความรู้ให้แก่ผู้ทำวิจัย ทั้งในอดีตตลอดจนถึงปัจจุบัน และกราบขอบพระคุณบิดา มารดา รวมถึงครอบครัวทุกท่านที่ให้การอบรม เลี้ยงดู และส่งเสริมการศึกษาเป็นอย่างดีมาโดยตลอด จนทำให้ผู้วิจัยประสบความสำเร็จในชีวิตเรื่อยมา

สิริศักดิ์ เกียงวงษ์สันต์

# สารบัญ

หน้า

บทคัดย่อ (ภาษาไทย).....	ก
บทคัดย่อ (ภาษาอังกฤษ).....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ช
คำอธิบายสัญลักษณ์และคำย่อ.....	ญ
<b>บทที่</b>	
<b>1 บทนำ.....</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของงานวิจัย.....	4
1.3 ข้อตกลงเบื้องต้น.....	4
1.4 ขอบเขตของวิทยานิพนธ์.....	4
1.5 วิธีการดำเนินงาน.....	4
1.6 ประโยชน์ที่คาดว่าจะได้รับ.....	5
1.7 การจัดทำรูปเล่มวิทยานิพนธ์.....	5
<b>2 ปรัชญาวรรณกรรมและงานวิจัยที่เกี่ยวข้อง.....</b>	<b>6</b>
2.1 การประมวลผลภาพดิจิทัล.....	8
2.2 รูปร่างของภาพ(Image Shape).....	9
2.3 มาตรฐานของสี.....	10
2.4 การกรองข้อมูลภาพ.....	15
2.5 การกรองโดยใช้หน้าต่าง.....	18
2.6 การกรองโดยวิธีคอนโวลูชัน.....	20
2.7 การค้นหาเส้นตรงโดยใช้วิธีการของ Hough Transform.....	34

## สารบัญ (ต่อ)

หน้า

2.8	การค้นหาเส้นโค้งหรือวงกลมโดยใช้วิธีการของ Hough Transform .....	39
2.9	โครงข่ายประสาทเทียม .....	40
2.10	โครงข่ายประสาทเทียมหนึ่งหน่วยแบบหลายอินพุต.....	44
2.11	โครงข่ายประสาทเทียมแบบหลายชั้น .....	45
2.12	งานวิจัยที่เกี่ยวข้อง.....	47
<b>3</b>	<b>วิธีการดำเนินการวิจัย.....</b>	<b>49</b>
3.1	การวิเคราะห์ปัญหาารูปแบบของเสียที่เกิดขึ้น บนแผ่นบันทึกข้อมูลของฮาร์ดดิสก์.....	49
3.2	การเตรียมข้อมูล .....	52
3.3	การคัดแยกรูปแบบ Dent pattern .....	53
3.4	การคัดแยกรูปแบบ Full pattern และแบบ Circum pattern ร่วมกับโครงข่ายประสาทเทียม.....	54
3.5	การปรับสอนโครงข่ายประสาทเทียม.....	56
3.6	การคัดแยกรูปแบบ Radial line และแบบ Straight line.....	65
<b>4</b>	<b>การพัฒนาเครื่องมือและผลการทดลอง.....</b>	<b>72</b>
4.1	ขั้นตอนการคัดแยกรูปแบบของเสีย .....	72
4.2	การพัฒนาเครื่องมือเพื่อใช้ในการคัดแยกของเสีย บนแผ่นบันทึกข้อมูล .....	74
4.3	ผลการทดลอง.....	76
<b>5</b>	<b>สรุปและข้อเสนอแนะ .....</b>	<b>77</b>
5.1	สรุปผลงานวิจัย.....	77
5.2	ข้อเสนอแนะในงานวิจัยต่อไป.....	78
	รายการอ้างอิง.....	79
	ภาคผนวก	
	ภาคผนวก ก. ข้อมูลที่ใช้ตรวจสอบโปรแกรมที่สร้างขึ้น .....	81
	ภาคผนวก ข. บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา.....	93
	ประวัติผู้เขียน .....	101



## สารบัญตาราง

ตารางที่		หน้า
2.1	ค่าความสัมพันธ์ระหว่างค่า $\rho$ และ $\theta$ .....	37
3.1	ชุดข้อมูลเพื่อทำการสอนของเสียในรูปแบบเต็มพื้นที่.....	57
3.2	ชุดข้อมูลเพื่อทำการทดสอบโครงข่ายประสาทเทียม ของเสียในรูปแบบเต็มพื้นที่ .....	59
3.3	เปรียบเทียบค่าความผิดพลาด (MSE) โครงข่าย ประสาทเทียมของเสียในรูปแบบเต็มพื้นที่.....	60
3.4	ชุดข้อมูลเพื่อทำการสอนในรูปแบบวงแหวน.....	61
3.5	ชุดข้อมูลเพื่อทำการทดสอบโครงข่ายประสาทเทียม ของเสียในรูปแบบวงแหวน .....	63
3.6	เปรียบเทียบค่าความผิดพลาด (MSE) โครงข่าย ประสาทเทียมของเสียในรูปแบบวงแหวน .....	64

## สารบัญรูป

รูปที่		หน้า
2.1	แสดงการแบ่งแตรคและเซ็กเตอร์ .....	7
2.2	แสดงไซลินเดอร์ของฮาร์ดดิสก์ .....	7
2.3	แสดงระบบสี HSV .....	12
2.4	แสดงระบบสี HLS.....	13
2.5	แสดงตัวอย่างการกรองข้อมูลภาพโดยใช้ค่ามัธยฐาน.....	17
2.6	แสดงตัวอย่างการกรองข้อมูลภาพโดยใช้ค่าฐานนิยม .....	18
2.7	แสดงตัวอย่างการกรองโดยใช้หน้าต่าง.....	18
2.8	แสดงการกรองข้อมูลภาพโดยใช้หน้าต่าง.....	20
2.9	แสดงผลของการเติมค่าศูนย์รอบภาพเริ่มต้นก่อนทำการคอนโวลูชัน .....	21
2.10	แสดงการคอนโวลูชันที่มีการเติมค่าศูนย์ให้กับภาพเริ่มต้น .....	22
2.11	แสดงตัวอย่างการกรองภาพด้วยเทมเพลต กรองความถี่สูงและความถี่ต่ำ .....	24
2.12	แสดงตัวอย่างการหาความแรงของขอบภาพด้วยวิธีโซเบล .....	26
2.13	แสดงตัวอย่างการหาความแรงของขอบภาพ ด้วยวิธีโซเบลภาพสุดท้ายได้จาก.....	26
2.14	แสดงการหาขอบภาพโดยใช้อนุพันธ์อันดับหนึ่ง.....	30
2.15	แสดงการหาขอบภาพโดยใช้อนุพันธ์อันดับสอง .....	31
2.16	แสดงขั้นตอนการหาขอบภาพโดยวิธีของ Canny .....	32
2.17	แสดงการนับจำนวนเส้นตรงของการแปลง Hough Transform .....	34
2.18	แสดงการแปลงรูปแบบระหว่าง Image Space กับ Parameter Space .....	35
2.19	แสดงความสัมพันธ์ระหว่างค่าของเวกเตอร์ $\rho$ กับจุดของเส้นตรง .....	35
2.20	แสดงการหาเส้นตรงของภาพสองมิติโดยใช้ Hough Transform.....	36
2.21	แสดงตัวอย่างการทำ Hough Transform.....	38
2.22	แสดงการแปลงค่าของ Hough Space เมื่อ มีการเปลี่ยนแปลงจุดในเส้นตรง.....	38

## สารบัญรูป (ต่อ)

รูปที่	หน้า
2.23	แสดงโครงข่ายประสาทเทียมหนึ่งหน่วย แบบไม่มีไบแอส ..... 41
2.24	แสดงโครงข่ายประสาทเทียมหนึ่งหน่วย แบบมีไบแอส..... 42
2.25	แสดงฟังก์ชันการถ่ายโอนแบบลอกซิกมอยด์ และสัญลักษณ์ที่ใช้..... 43
2.26	แสดงฟังก์ชันการถ่ายโอนแบบเชิงเส้น และสัญลักษณ์ที่ใช้..... 43
2.27	แสดงโครงข่ายประสาทเทียมหนึ่งหน่วยแบบหลายอินพุต..... 44
2.28	แสดงโครงข่ายประสาทเทียมหนึ่งหน่วยแบบหลายอินพุตในรูปแบบย่อ ..... 45
2.29	แสดงโครงข่ายประสาทเทียมสามชั้น ..... 45
2.30	แสดงโครงข่ายประสาทเทียมหลายอินพุตหลายชั้นในรูปแบบย่อ ..... 46
3.1	แสดงรูปแบบของเสียที่เกิดจากรอยขูดตัว ของแผ่นบันทึกข้อมูล (Dent Pattern)..... 50
3.2	แสดงรูปแบบของเสียจากกระจายตัวแบบเต็มพื้นที่ (Full tern)..... 50
3.3	แสดงรูปแบบของเสียจากการกระจายตัวแบบ จับกลุ่มเป็นวงแหวน (Circum Pattern) ..... 51
3.4	แสดงรูปแบบของเสียจากกระจายตัวแบบ จับกลุ่มแนวรัศมี (Radial Pattern) ..... 51
3.5	แสดงรูปแบบของเสียจากการกระจายตัว แบบจับกลุ่มเป็นเส้น (line Pattern) ..... 52
3.6	แสดงการแปลง Text File ไปเป็น รูปภาพโดยใช้ Python Imaging Library (PIL)..... 66
3.7	แสดงภาพเมื่อผ่านขั้นตอนการหาขอบด้วย Canny Method ..... 67
3.8	แสดงตัวอย่างรูปต้นฉบับ Circum Pattern ก่อนเข้ากระบวนการ Circle Hough Transform ..... 68
3.9	แสดง (Circum Pattern) ในรูปแบบ Accumulator 2D หลังผ่าน กระบวนการ Circle Hough Transform..... 68
3.10	แสดง (Circum Pattern) ในรูปแบบ Accumulator 3D หลังผ่าน กระบวนการ Circle Hough Transform ..... 69

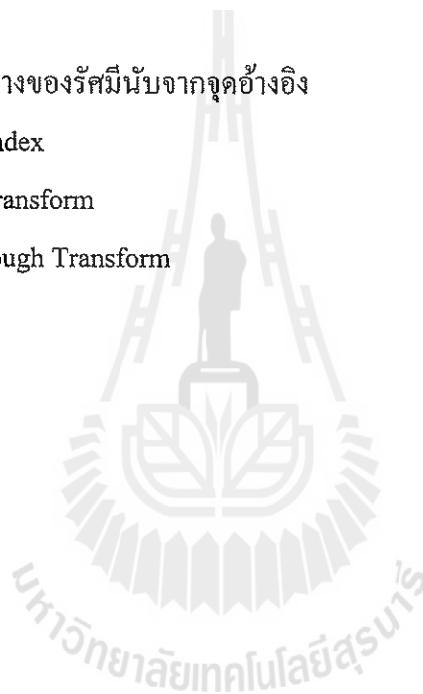
## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.11 แสดงตัวอย่างรูปต้นฉบับ Line Pattern ก่อนเข้ากระบวนการ Hough Transform .....	69
3.12 แสดง (Line Pattern ) ในรูปแบบ Accumulator 2D หลังผ่าน กระบวนการ Hough Transform.....	70
3.13 แสดง (Line Pattern) ในรูปแบบ Accumulator 3D หลังผ่าน กระบวนการHough Transform.....	70
4.1 แสดงผังกระบวนการคัดแยกรูปแบบของเส้นบนแผ่นบันทึกข้อมูล .....	73
4.2 แสดงหน้าแสดงผลผ่าน Web Application .....	75
4.3 แสดงการติดต่อกันระหว่าง Engine และ Web Application .....	75



## คำอธิบายสัญลักษณ์และคำย่อ

HDD	=	Hard Disk Drive
$\theta$	=	ค่ามุม
X	=	แนวตั้ง
Y	=	แนวระดับ
Z	=	แนวตั้ง
$\rho$	=	ค่าระยะทางของรัศมีนับจากจุดอ้างอิง
CI	=	Cluster Index
HT	=	Hough Transform
CHT	=	Circle Hough Transform



# บทที่ 1

## บทนำ

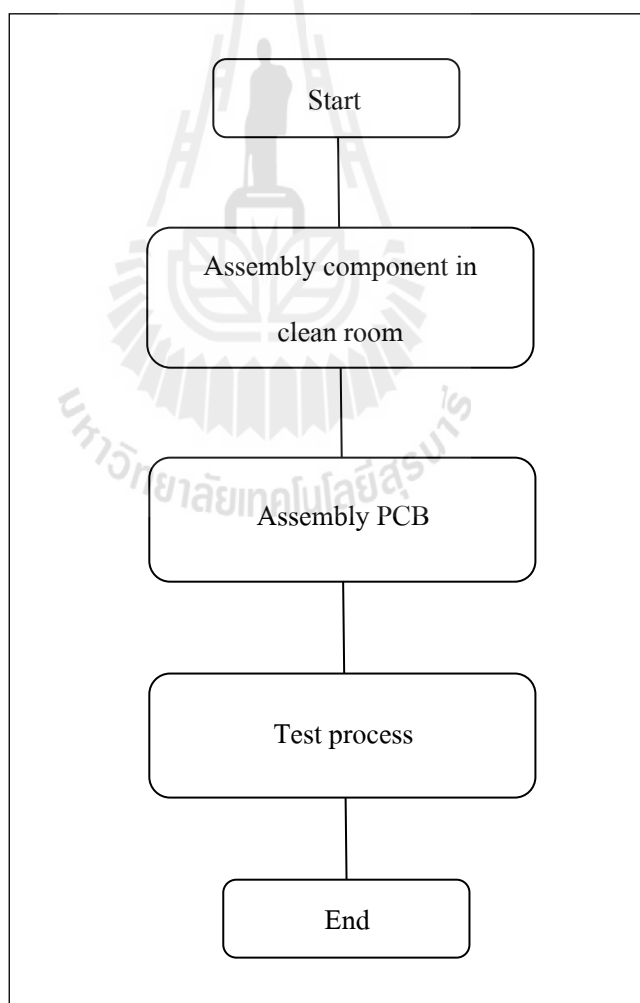
### 1.1 ความเป็นมาและความสำคัญของปัญหา

ฮาร์ดดิสก์ (Hard Disk Drive, HDD) เป็นอุปกรณ์บันทึกข้อมูลที่รู้จักกันดี ซึ่งในปัจจุบันความต้องการในการบันทึกข้อมูลดิจิทัลมีเพิ่มมากขึ้นดังนั้นจึงมีการนำฮาร์ดดิสก์ไปใช้งานหลายด้านและเพื่อเป็นการรองรับการใช้งานความจุของฮาร์ดดิสก์จึงต้องเพิ่มขึ้นตามไปด้วยซึ่งทำให้กระบวนการผลิตฮาร์ดดิสก์มีความซับซ้อนเพิ่มมากขึ้นและเป็นที่แน่นอนว่าในกระบวนการผลิตนั้นจะต้องมีของเสียเกิดขึ้นดังนั้นเพื่อเป็นการลดของเสียให้น้อยที่สุดจึงจำเป็นต้องมีการวิเคราะห์ความเสียหายของฮาร์ดดิสก์ว่ามีมาจากสาเหตุใดเพื่อดำเนินการแก้ไขต่อไป และข้อมูลที่สำคัญในการวิเคราะห์ของเสียคือข้อมูลของตำแหน่งที่ไม่สามารถอ่านหรือเขียนข้อมูลได้บนแผ่นบันทึกข้อมูลหรือ media โดยการวิเคราะห์การเสียของแผ่น media ในปัจจุบันทำได้โดยการนำข้อมูลมาจาก result file ที่ได้มาจากกระบวนการทดสอบมาวิเคราะห์ เพื่อระบุถึงสาเหตุของความเสียหายที่เกิดขึ้น

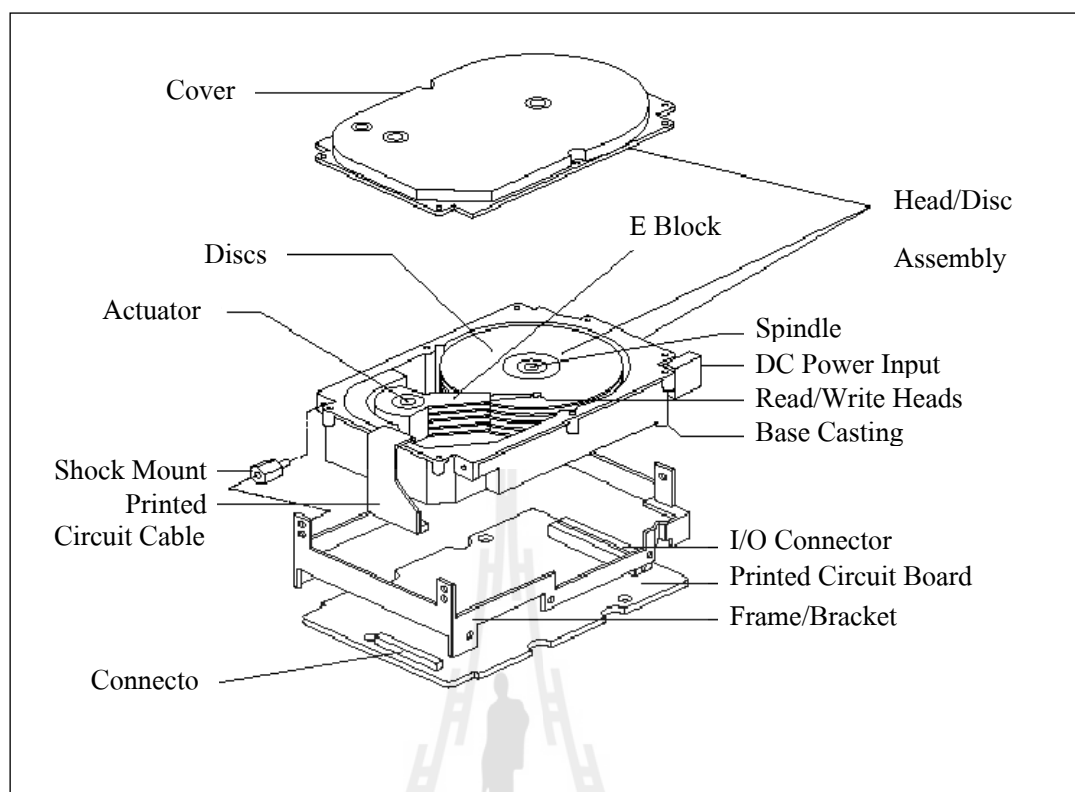
เดิมทีนั้นการวิเคราะห์อาการเสียของฮาร์ดดิสก์ที่เกิดขึ้นนั้นจำเป็นต้องอาศัยคนในการวิเคราะห์ข้อมูลที่ได้มาจากกระบวนการผลิตแต่ข้อจำกัดของวิธีนี้ก็คือจะไม่สามารถวิเคราะห์ข้อมูลที่มีจำนวนมากได้เพราะต้องใช้เวลาและทรัพยากรในการวิเคราะห์เป็นอย่างมากดังนั้นงานวิจัยนี้จึงนำเสนอการวิเคราะห์อาการเสียของฮาร์ดดิสก์จากรูปแบบของเสียที่เกิดขึ้นโดยใช้วิธีการประมวลผลภาพและโครงข่ายประสาทเทียมเพื่อให้สามารถดำเนินการกับข้อมูลที่มีจำนวนมากอย่างมีประสิทธิภาพได้

### 1.1.1 กระบวนการผลิตฮาร์ดดิสก์

ในกระบวนการผลิตของฮาร์ดดิสก์ โดยทั่วไปนั้นจะมีอยู่ 3 ส่วนหลัก ๆ คือการนำอุปกรณ์ต่างของฮาร์ดดิสก์มาประกอบโดยเครื่องจักรอัตโนมัติภายในห้องที่มีการควบคุมปริมาณของฝุ่นละอองในมาตรฐานที่กำหนดเหตุที่ต้องใช้เครื่องจักรในการประกอบก็เพราะในกระบวนการประกอบนั้นจะต้องใช้ความแม่นยำที่สูงในการประกอบหรือยึดอุปกรณ์แต่ละชิ้นเข้าด้วยกันจะต้องมีการวัดและควบคุมทุกขั้นตอนเพราะยิ่งฮาร์ดดิสก์มีความจุเพิ่มมากขึ้นเท่าไรความถูกต้องและความแม่นยำในการประกอบก็ต้องเพิ่มมากขึ้นไปด้วยเช่นกันและเมื่อประกอบเสร็จอยู่ในรูปของฮาร์ดดิสก์แล้วกระบวนการต่อไปก็คือการประกอบแผ่นวงจรควบคุมหรือแผ่น PBC เข้ากับฮาร์ดดิสก์ก่อนที่จะนำฮาร์ดดิสก์ที่ผ่านการประกอบแผ่นวงจรควบคุมมาเรียบร้อยแล้วไปทำการทดสอบด้วยเครื่องทดสอบ



รูปที่ 1.1 แสดงกระบวนการผลิตฮาร์ดดิสก์



รูปที่ 1.2 แสดงส่วนประกอบของ ของฮาร์ดดิสก์

ในการทดสอบนั้นก็จะมีการทดสอบที่เรียกว่า Initial Test และ Final Test (IO) โดยในส่วน ของ Initial Test ก็จะทำการทดสอบการติดต่อและการทำงานเบื้องต้นของระบบกลไกต่าง ๆ ของ ฮาร์ดดิสก์ไดรฟ์ เช่น การทำการทดสอบในการติดต่อและส่งคำสั่งการทำงานให้กับฮาร์ดดิสก์ไดรฟ์ (Serial Communication Test) หรือทดสอบการเคลื่อนที่ของกลไก (Mechanical Scan) เช่น การหมุน ของมอเตอร์, การเคลื่อนที่ของแขนหัวอ่านและก็การทำ servo tuning เป็นต้น ส่วนใน Final Test นั้น ก็จะทำการทดสอบฮาร์ดดิสก์ไดรฟ์เกี่ยวกับการรับ-ส่ง ข้อมูลผ่านทางอินพุตและเอาต์พุตพอร์ต และ ทำการทดสอบเกี่ยวกับการ อ่าน-เขียน ข้อมูลของฮาร์ดดิสก์

โดยแต่ละขั้นตอนในการทดสอบนั้นจะมีการเก็บบันทึกข้อมูลผลการทดสอบซึ่งเราจะเรา จะเรียกว่า Result file ซึ่งใน Result file นี้จะเก็บข้อมูลการทดสอบทุกอย่างให้อยู่ในรูปของตัวอักษร หรือ Text file ประโยชน์ของการเก็บข้อมูลการบันทึกไว้ทั้งหมดนั้นจะส่งผลดีต่อการวิเคราะห์เมื่อมี การผิดปกติหรือมีการเสียหายเกิดขึ้นในระหว่างการทดสอบและปัญหาที่เกิดขึ้นเป็นส่วนใหญ่ใน การประกอบฮาร์ดดิสก์ก็คือเกิดการเสียหายบนแผ่นบันทึกข้อมูลซึ่งในการวิเคราะห์สาเหตุของการ เสียหายบนแผ่นบันทึกข้อมูลนั้นนั้นจะทำได้จากการนำข้อมูลที่ได้จากกระบวนการทดสอบมา plot



ข้อมูลการเสียหายที่เกิดขึ้นบนแผ่นบันทึกข้อมูลเป็นรูปภาพแล้วจึงให้ผู้เชี่ยวชาญทำการวิเคราะห์รูปแบบของเสียที่เกิดขึ้นเพื่อจะหาสาเหตุที่ทำให้เกิดความเสียหายขึ้นและแก้ปัญหาที่ต้นเหตุได้อย่างทัน่วงที

## 1.2 วัตถุประสงค์ของงานวิจัย

- 1.2.1 เพื่อศึกษารูปแบบและจำแนก Defect แบบต่างๆบนแผ่นบันทึกข้อมูล
- 1.2.2 เพื่อศึกษาเทคนิคการประมวลผลภาพ
- 1.2.3 ประยุกต์ใช้เทคนิคการประมวลผลภาพในการจำแนก Defect

## 1.3 ข้อตกลงเบื้องต้น

- 1.3.1 ใช้ Software Python และ MATLAB
- 1.3.2 Media ของ HDD ที่ใช้ในการวิจัยเป็นแบบ 3.5 นิ้ว 7200 rpm

## 1.4 ขอบเขตของวิทยานิพนธ์

- 1.4.1 ศึกษากรูปร่างรูปแบบของการของ defect บนแผ่น media (defect pattern) ที่อยู่ในรูปแบบ Circum, Line, Radial
- 1.4.2 ศึกษาวิธีการทางการประมวลผลภาพและวิธีการที่เหมาะสมเพื่อวิเคราะห์ defect ใน media ที่เราสนใจว่าอยู่ใน pattern แบบใด

## 1.5 วิธีการดำเนินงาน

- 1.5.1 การดำเนินงาน
  1. สํารวจและศึกษาปริทัศน์และวรรณกรรมที่เกี่ยวข้อง
  2. ศึกษาและเก็บตัวอย่างข้อมูล
  3. วิเคราะห์ข้อมูลตัวอย่างเพื่อนำมากำหนดทำเป็น defect pattern
  4. ศึกษาวิธีการประมวลผลภาพและวิธีการตัดแยกรูปแบบ defect ที่เหมาะสม
  5. พัฒนา software เพื่อนำมาวิเคราะห์ defect บนแผ่น media
  6. ทดสอบผลที่ได้จาก software
  7. วิเคราะห์และสรุปผลที่ได้จากการทดลอง
  8. เรียบเรียงวิทยานิพนธ์ ตรวจสอบแก้ไขและเข้ารูปเล่มเพื่อนำเสนอคณะกรรมการ

ต่อไป

### 1.5.2 ระเบียบวิธีวิจัย

1. วิเคราะห์รูปแบบของเสียหรือ defect ที่เกิดขึ้นบนแผ่นบันทึกข้อมูล โดยใช้วิธีการประมวลผลภาพและวิธีการคัดแยกรูปแบบ defect ที่เหมาะสม

### 1.5.3 เครื่องมือที่ใช้ในการวิจัย

1. Laptop Intel Centino Core2 Duo 2.0 GHz
2. MATLAB 7.0
3. Software Python

### 1.5.4 การเก็บรวบรวมข้อมูล

1. เก็บข้อมูลรูปแบบของเสียที่เกิดขึ้นบนแผ่นบันทึกข้อมูลหรือ defect ที่ได้จากกระบวนการทดสอบ

## 1.6 ประโยชน์ที่คาดว่าจะได้รับ

1.6.1 เพื่อลดของเสียที่เกิดขึ้นเนื่องจาก media defect

1.6.2 เพื่อวิเคราะห์หาสาเหตุของการเสียหายที่เกิดขึ้นบนแผ่นบันทึกข้อมูลได้อย่างรวดเร็ว

## 1.7 การจัดทำรูปเล่มวิทยานิพนธ์

วิทยานิพนธ์นี้ประกอบด้วย 5 บท 1 ภาคผนวก ซึ่งมีรายละเอียดโดยย่อ ดังนี้

บทที่ 1 เป็นบทนำซึ่งจะกล่าวถึงความสำคัญของปัญหาวัตถุประสงค์และเป้าหมายของงานวิจัยวิทยานิพนธ์ตลอดจนขอบเขตและประโยชน์ที่คาดว่าจะได้รับจากงานวิจัยนี้

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่เกี่ยวกับการประมวลผลภาพรวมถึงปริศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

บทที่ 3 กล่าวถึงวิธีดำเนินการทดลอง

บทที่ 4 การพัฒนาเครื่องมือและผลการทดลอง

บทที่ 5 เป็นบทสรุปและข้อเสนอแนะ

## บทที่ 2

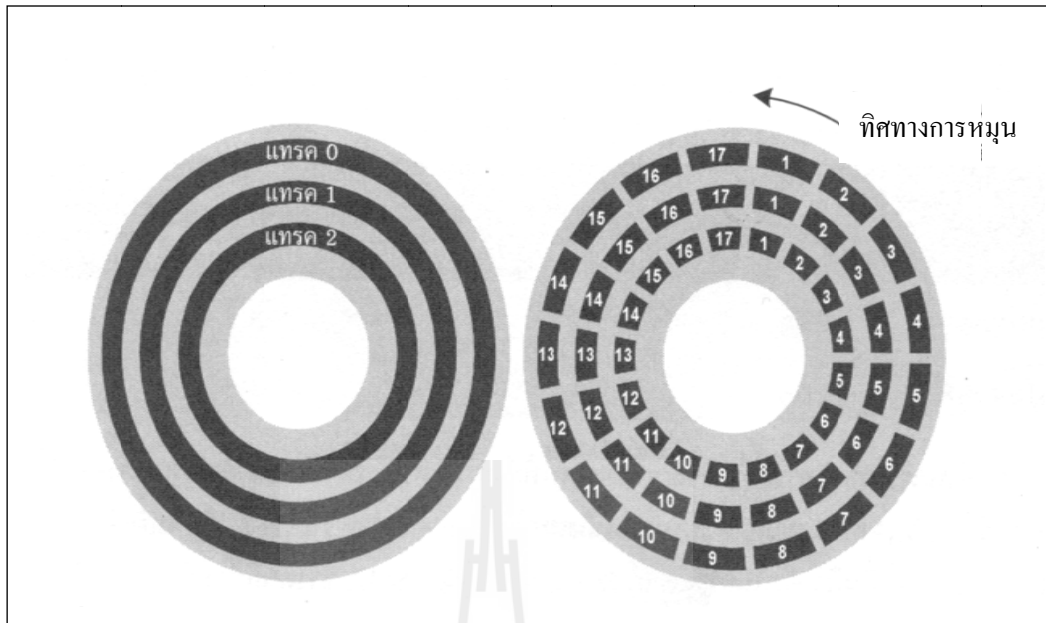
### ปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

โครงสร้างของการจัดเรียงและการเก็บข้อมูลของฮาร์ดดิสก์ข้อมูลที่เก็บบนแผ่นบันทึกข้อมูลทางกายภาพอาจจะอยู่กันอย่างกระจัดกระจายไม่เรียงลำดับทำให้ต้องมีการแบ่งพื้นที่ของฮาร์ดดิสก์ออกเป็น ส่วน ๆ ซึ่งแบ่งออกเป็น แทรค (Track) เซ็กเตอร์ (Sector) และ ไซลินเดอร์ (Cylinder) ก็เพื่อที่จะทำให้โปรแกรมใช้งานต่างๆสามารถอ้างตำแหน่งทางกายภาพบนฮาร์ดดิสก์ได้

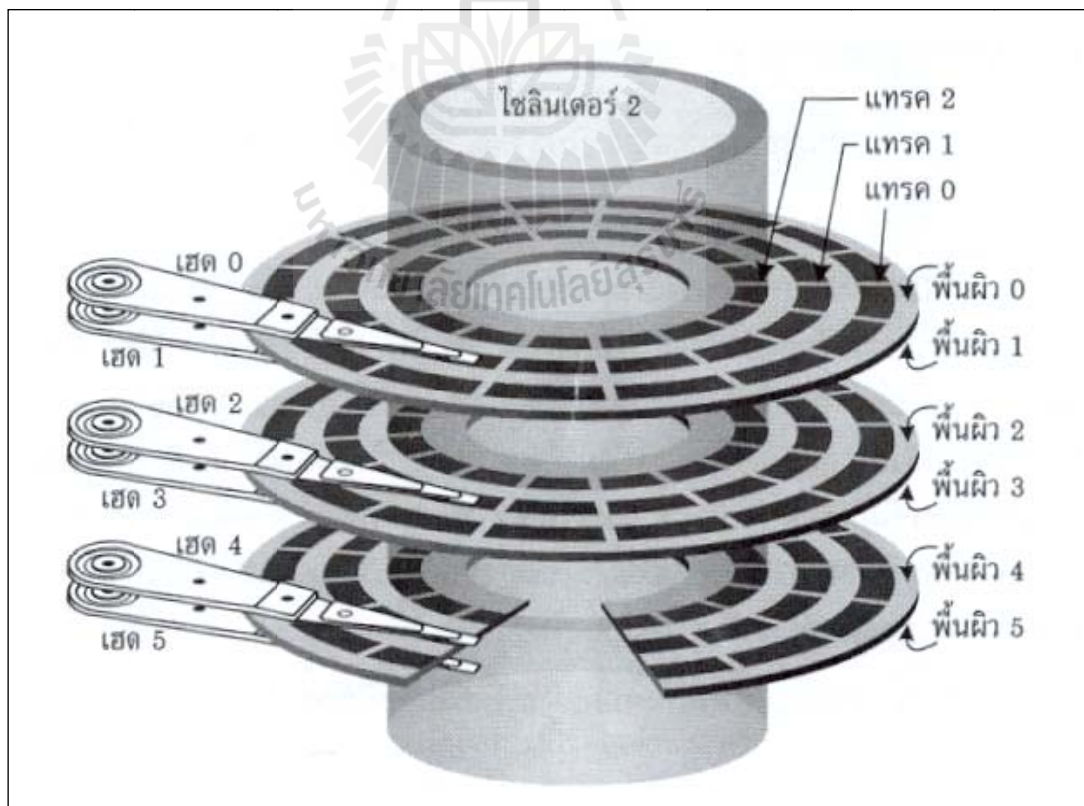
โครงสร้างการเก็บข้อมูลภายในฮาร์ดดิสก์ ในการจัดเรียงข้อมูลในฮาร์ดดิสก์นั้นสิ่งที่สำคัญก็คือ จะต้องมีการกำหนดโครงสร้างในการเก็บข้อมูลเพื่อที่จะให้เครื่องคอมพิวเตอร์รู้ว่าจะต้องนำข้อมูลไปเก็บไว้ที่ใด หรืออ่านข้อมูลมาจากตำแหน่งใด ก่อนการใช้งานฮาร์ดดิสก์จึงต้องมีการกำหนดโครงสร้างหรือที่เรียกว่า การฟอร์แมตระดับต่ำ (Low Level Format) จะมีผลในการกำหนดโครงสร้างภายในให้เป็นส่วนย่อย ๆ ที่เรียกว่า เซ็กเตอร์ขึ้นมา นอกจากนี้ยังมีการจัดกลุ่มของเซ็กเตอร์เป็นแทรค และกลุ่มของแทรคเรียกว่า ไซลินเดอร์ หากฮาร์ดดิสก์ยังไม่ได้ถูกฟอร์แมตเครื่องคอมพิวเตอร์จะไม่ว่าข้อมูลอยู่ตำแหน่งไหนบ้างเป็นต้น

ความหมายของแทรค คือ พื้นที่ผิวบนแผ่นบันทึกข้อมูลบนฮาร์ดดิสก์จะถูกแบ่งออกเป็นวงรอบ ซึ่งจะอ้างถึงในแต่ละแทรคได้โดยอาศัยลำดับเลขกำกับ โดยแทรค 0 จะอยู่วงนอกสุดส่วนแทรคที่ 1, 2, ... จะเป็นวงรอบถัดไปด้านในโดยลำดับ ดังนั้นแทรคที่มีลำดับเลขมากที่สุดจะอยู่วงในสุด โดยวงนอกจะเก็บข้อมูลได้มากกว่าวงใน ซึ่งฮาร์ดดิสก์โดยทั่วไปจะมีประมาณ 105,000 แทรคต่อนี้ว

ในแต่ละแทรคจะแบ่งออกเป็นกลุ่มย่อยของข้อมูลตามแนวรัศมีของวงกลม คล้ายการแบ่งขนมเค้ก เรียกว่าเซ็กเตอร์ ซึ่งแต่ละเซ็กเตอร์โดยปกติจะเก็บได้ 512 ไบต์ และแต่ละเซ็กเตอร์จะถูกกำกับด้วยตัวเลขอ้างอิง โดยปกติแล้วการที่แทรควงนอกมีพื้นที่มากกว่าแทรควงใน ในขณะที่เซ็กเตอร์ในแต่ละแทรคนั้นมีจำนวนเท่ากัน ทำให้เกิดการสิ้นเปลืองพื้นที่บนแผ่นบันทึกข้อมูล จึงได้มีการจัดเก็บข้อมูลแบบใหม่ที่สามารถทำให้จำนวนเซ็กเตอร์ของแทรควงนอกมีจำนวนมากกว่าแทรควงในได้ ซึ่งทำให้ฮาร์ดดิสก์สามารถจุข้อมูลได้เพิ่มขึ้นซึ่งเราเรียกการเก็บข้อมูลแบบนี้ว่าการเก็บข้อมูลแบบแบ่งโซน



รูปที่ 2.1 แสดงการแบ่งแทร็คและเซ็กเตอร์



รูปที่ 2.2 แสดงไซลินเดอร์ของฮาร์ดดิสก์

ไชลินเดอร์เป็นการจัดกลุ่มของแทรกหมายเลขเดียวกัน ของแผ่นบันทึกข้อมูลทุก ๆ แผ่น ดังนั้นไชลินเดอร์ จึงมีลักษณะเป็นทรงกระบอกเนื่องจากฮาร์ดดิสก์จะประกอบด้วยหลายหัวอ่าน จึงสามารถทำการอ้างอิงหรือการเข้าถึงได้โดยระบุหมายเลขของ ไชลินเดอร์ หัวอ่าน และเซ็กเตอร์ ซึ่งประโยชน์ ของการแบ่งข้อมูลเป็นไชลินเดอร์ คือ ไม่ต้องเสียเวลาในการเลื่อนหัวอ่าน หากข้อมูลอยู่ในไชลินเดอร์เดียวกัน

## 2.1 การประมวลผลภาพดิจิทัล

การมองเห็นของมนุษย์เป็นสิ่งที่สำคัญและเป็นกลไกการรับภาพที่ซับซ้อนอย่างหนึ่ง ซึ่งจะให้ข้อมูลที่มีความจำเป็นสำหรับใช้ในางานง่าย ๆ (ตัวอย่างเช่น การจดจำวัตถุ) และสำหรับงานที่มีความซับซ้อน (ได้แก่ การวางแผน การตัดสินใจ การค้นคว้าทางวิทยาศาสตร์ การพัฒนาทางด้านความคิด) ดังคำสุภาษิตของจีนกล่าวไว้ว่า "รูปภาพสามารถแทนคำได้เป็นพัน ๆ คำ" รูปภาพมีบทบาทมากสำหรับองค์กรต่าง ๆ เช่น หนังสือพิมพ์โทรทัศน์ภาพยนตร์ซึ่งได้ใช้ภาพ (ภาพนิ่ง ภาพเคลื่อนไหว) เป็นสื่อนำเสนอข้อมูลข่าวสารต่าง ๆ สิ่งที่น่าสนใจของข้อมูลที่เกี่ยวข้องกับการมองเห็นหรือข้อมูลภาพนั้นก็คือกระบวนการประมวลผลภาพ (Image Processing) โดยใช้ดิจิทัลคอมพิวเตอร์ความพยายามทางด้านการประมวลผลภาพได้เริ่มขึ้นในปี 1964 ณ ห้องแลป Jet Propulsion (Pasadena California) ซึ่งได้นำการบวนการประมวลผลภาพมาใช้ในการพิจารณาภาพถ่ายดาวเทียมของดวงจันทร์ต่อมาได้มีการตั้งสาขาทางวิทยาศาสตร์สาขาใหม่มีชื่อว่า Digital image processing หลังจากนั้นงานทางด้านการประมวลผลภาพก็พัฒนาขึ้นเรื่อย ๆ และใช้กันอย่างกว้างขวางสำหรับงานในหลาย ๆ ด้านตัวอย่างเช่นทางด้านสื่อสาร โทรคมนาคม การสื่อสารทางโทรทัศน์ทางการพิมพ์ทางด้านกราฟฟิกการแพทย์และการค้นคว้าทางวิทยาศาสตร์ Digital image processing จะเกี่ยวกับการแปลงข้อมูลภาพให้อยู่ในรูปแบบข้อมูลดิจิทัล (Digital format) ซึ่งสามารถที่จะนำเอาข้อมูลนี้จัดผ่านกระบวนการต่าง ๆ ด้วยดิจิทัลคอมพิวเตอร์ได้ ในระบบของดิจิทัล อินพุตและเอาพุตของระบบจะอยู่ในรูปแบบดิจิทัลเท่านั้น Digital image analysis จะเกี่ยวกับวิธีการอธิบายและการจดจำข้อมูลภาพดิจิทัล ซึ่งอินพุตของระบบจะเป็นข้อมูลภาพดิจิทัลและเอาพุตจะเป็นเครื่องหมายที่ใช้แทนข้อมูลภาพดิจิทัลเหล่านั้น ในการวิเคราะห์ภาพมีอยู่หลายวิธีด้วยกันที่ได้นำมาจากการทำงานของตามนุษย์ (human vision) นั่นก็คืองานทางด้าน Computer Vision เป็นลักษณะเดียวกับ Digital image analysis นั่นเอง การมองเห็นของมนุษย์นับว่าเป็นกระบวนการที่ซับซ้อนซึ่งลักษณะเทคนิคโดยทั่ว ๆ ไปในกระบวนการ Digital image analysis และ Computer Vision จะค่อนข้างซับซ้อน เช่นกัน

## 2.2 รูปร่างของภาพ (Image Shape)

วัตถุที่มีอยู่ตามธรรมชาติและที่มนุษย์สร้างขึ้นมีรูปร่างที่แตกต่างกันไป ทั้งที่เป็นรูปทรงเรขาคณิตและไม่เป็นรูปทรงเรขาคณิต ในศาสตร์ของการประมวลผลภาพนั้น การกำหนดขอบเขตของภาพทุกภาพให้อยู่ในรูปสี่เหลี่ยม (Rectangular image model) เป็นวิธีที่นิยมใช้กันมากที่สุด เนื่องจากทำให้การอ่านภาพ การจัดเก็บข้อมูลภาพในหน่วยความจำ และการแสดงผลภาพออกทางอุปกรณ์ต่าง ๆ เป็นไปได้ง่ายและมีประสิทธิภาพ

การเก็บข้อมูลภาพลงหน่วยความจำของคอมพิวเตอร์สามารถทำได้โดยการจองหน่วยความจำของเครื่องไว้ในรูปของตัวแปรอะเรย์ (array) โดยค่าในแต่ละช่องของอะเรย์แสดงถึงคุณสมบัติของจุดภาพ (pixel) และตำแหน่งของช่องอะเรย์เป็นตัวกำหนดตำแหน่งของจุดภาพสมมติให้ Image เป็นตัวแปรแบบอะเรย์ขนาด  $M \times N$  ( $M$  แถว และ  $N$  คอลัมน์) ที่ใช้เก็บภาพขนาด  $M \times N$  จุด ( $M$  จุดในแนวนอน และ  $N$  จุดในแนวตั้ง) ค่าสี (หรือความสว่าง ในกรณีที่เป็นภาพ grey level) ของจุดภาพในแถวที่ 5 คอลัมน์ที่ 4 จะตรงกับค่าของ Image (5, 4) จะเห็นว่าเราใช้ตำแหน่งของจุดภาพทั้งสองแกนเป็นตัวชี้ค่าข้อมูลในอะเรย์

จากการใช้หน่วยความจำเพื่อการเก็บภาพในลักษณะที่กล่าวมา เนื้อที่ในการเก็บภาพสามารถคำนวณได้จาก  $M \times N \times g$  เมื่อ  $g$  เป็นจำนวนเต็มที่แทนจำนวนบิตของข้อมูลในแต่ละจุดภาพ ตัวอย่างถ้า  $g$  มีค่าเท่ากับ 8 บิตเราจะสามารถเก็บความแตกต่างของระดับสีที่เป็นไปสูงสุด 256 ระดับ ค่า  $M$  และ  $N$  จะเป็นตัวบอกถึงความละเอียดของภาพ สำหรับคอมพิวเตอร์ทั่วไปในระบบ VGA (Video Graphic Array) จะมีขนาด 640 x 480, 800 x 600 และ 1024 x 768 จุด เป็นต้น การกำหนดความละเอียดจะขึ้นอยู่กับงานที่จะใช้ในงานบางอย่างใช้ความละเอียดแค่ 30 x 50 จุดก็พอแล้วแต่ในงานบางชนิด ใช้ความละเอียดถึง 1000 x 1000 จุด ก็ยังไม่พอ

ปกติแล้วในการเก็บข้อมูลภาพโดยเครื่องมือต่าง ๆ จะเก็บตามมาตรฐานของโทรทัศน์ซึ่งมีอัตราส่วน  $x$  ต่อ  $y$  เท่ากับ 4:3 สำหรับเครื่องมือเก็บข้อมูลภาพที่ไม่เป็นไปตามอัตราส่วน 4:3 เมื่อนำภาพนี้ไปแสดงในจอภาพมาตรฐานจะทำให้ภาพที่แสดงนั้นมีขนาดของจุดภาพไม่เป็นสี่เหลี่ยมจัตุรัสเช่นในบางระบบอาจใช้ความละเอียดในการแสดงเท่ากับ 640 x 512 ซึ่งจะทำให้ขนาดของจุดภาพที่ได้มีขนาดของด้านกว้างมีความยาวมากกว่าด้านสูง ซึ่งลักษณะดังกล่าวนี้เป็นหัวข้อที่ต้องสนใจสำหรับการเขียนโปรแกรมทางด้านกราฟฟิกและการจัดการข้อมูล

จำนวนสีสูงสุดที่เป็นไปได้ของแต่ละจุดภาพขึ้นอยู่กับจำนวนบิตที่ใช้ เมื่อมีการกำหนดให้ขนาดของบิตต่อจุด มากขึ้นจะทำให้จำนวนของสีมากขึ้นด้วย ตัวอย่างเช่น

$$1 \text{ บิต} = 2^1 = 2 \text{ สี}$$

$$2 \text{ บิต} = 2^2 = 4 \text{ สี}$$

$$4 \text{ บิต} = 2^4 = 16 \text{ สี}$$

$$8 \text{ บิต} = 2^8 = 256 \text{ สี}$$

$$16 \text{ บิต} = 2^{16} = 65536 \text{ สี}$$

สำหรับการแสดงข้อมูลภาพที่มีขนาด 1 บิตและ 8 บิตนั้นจะมีการทำงานที่จะใกล้เคียงกัน เนื่องจากหน่วยประมวลผลจะไม่สามารถจัดการกับข้อมูลที่เป็นบิตเดี่ยว ๆ ได้ดังนั้นในการแสดงข้อมูลออกทางจอภาพตัวโปรเซสเซอร์จะทำการก๊อปปี้ข้อมูลทั้ง 8 บิต (1 Byte) ส่งให้กับจอภาพซึ่งในกรณีที่ Pixel มีขนาด 1 บิต เมื่อโปรเซสเซอร์จะทำงานกับบิตแรกที่ต้องการแล้วก็จะทำการก๊อปปี้ข้อมูลชุดใหม่ทันทีโดยที่ไม่เกี่ยวกับข้อมูลอีก 7 บิตที่เหลือส่วนในกรณี Pixel ที่มีขนาด 8 บิต โปรเซสเซอร์จะทำการก๊อปปี้ข้อมูลชุดใหม่ก็ต่อเมื่อโปรเซสเซอร์ทำงานกับทุกบิตแล้ว

ตัวอย่างสำหรับระบบที่มีความละเอียดเท่ากับ 800 x 600 และมีขนาด 16 บิตต่อ Pixel จะสามารถแสดงสีได้ทั้งหมด 65536 ระดับและต้องใช้เนื้อที่ในการเก็บเท่ากับ 800 x 600 x 16 บิต

## 2.3 มาตรฐานของสี

มาตรฐานของสีที่ใช้อยู่ในปัจจุบันมีอยู่หลายระบบด้วยกัน ทั้งนี้จะขึ้นอยู่กับนำไปใช้ แต่โดยทั่วไปแล้วทุกมาตรฐานจะมีแนวคิดเดียวกันคือ การแทนจุดสีด้วยจุดที่อยู่ภายในสเปส 3 มิติ โดยจะมีแกนอ้างอิงสำหรับจุดสีนั้นในสเปสซึ่งแต่ละแกนจะมีความเป็นอิสระต่อกัน ตัวอย่างเช่นในระบบ RGB จะมีแกนสีคือ แแกนสีแดง เขียว และน้ำเงินในระบบ HLS จะมีแกนเป็น ค่าสี (hue) ความสว่าง (lightness) และความบริสุทธิ์ของสี (saturation)

ตัวอย่างระบบสีที่นิยมใช้กัน ได้แก่ ระบบ RGB HSV (Hue Saturation Value) และ HLS (Hue Lightness Saturation)

### 2.3.1 ระบบสี RGB

ระบบสี RGB เป็นระบบสีที่เกิดจากการรวมกันของแสงสีแดง เขียว และน้ำเงิน โดยมีการรวมกันแบบ Additive ซึ่งโดยปกติจะนำไปใช้ในจอภาพแบบ CRT (Cathode ray tube) ในการใช้งานระบบสี RGB ยังมีการสร้างมาตรฐานที่แตกต่างกันออกไปที่นิยมใช้งานได้แก่  $RGB_{CIE}$  และ  $RGB_{NTSC}$

ระบบสีแบบ RGB ของ CIE เป็นระบบสีที่พัฒนาขึ้นโดย CIE (Commission International 1 'Eclairage) ซึ่งอ้างอิงสีด้วยสีแดงที่ 700 nm สีเขียวเท่ากับ 546.1 nm และสีน้ำเงิน 435.8 nm

ระบบสีแบบ RGB ของ NTSC เป็นระบบที่พัฒนาโดย NTSC (National Television System Committee) เพื่อใช้สำหรับการแสดงภาพของจอภาพแบบ CRT เป็นมาตรฐานสำหรับผู้ผลิตแบบ CRT ให้มีลักษณะเดียวกัน

### 2.3.2 ระบบสี HSV

ระบบสี HSV (Hue Saturation Value) เป็นการพิจารณาสีโดยใช้ Hue Saturation และ Value ซึ่ง Hue คือ ค่าสีของสีหลัก (แดง เขียวและน้ำเงิน) ในทางปฏิบัติจะอยู่ระหว่าง 0 และ 255 ซึ่งถ้า Hue มีค่าเท่ากับ 0 จะแทนสีแดงและเมื่อ Hue มีค่าเพิ่มขึ้นเรื่อย ๆ สีก็จะเปลี่ยนแปลงไปตามสเปกตรัมของสีจนถึง 256 จึงจะกลับมาเป็นสีแดงอีกครั้งซึ่งสามารถแทนให้อยู่ในรูปขององศาได้ดังนี้คือ สีแดง = 0 องศา สีเขียวเท่ากับ 120 องศา สีน้ำเงินเท่ากับ 240 องศา Hue สามารถคำนวณได้จากระบบสี RGB ได้ดังนี้

*red red min (red, green, blue)*

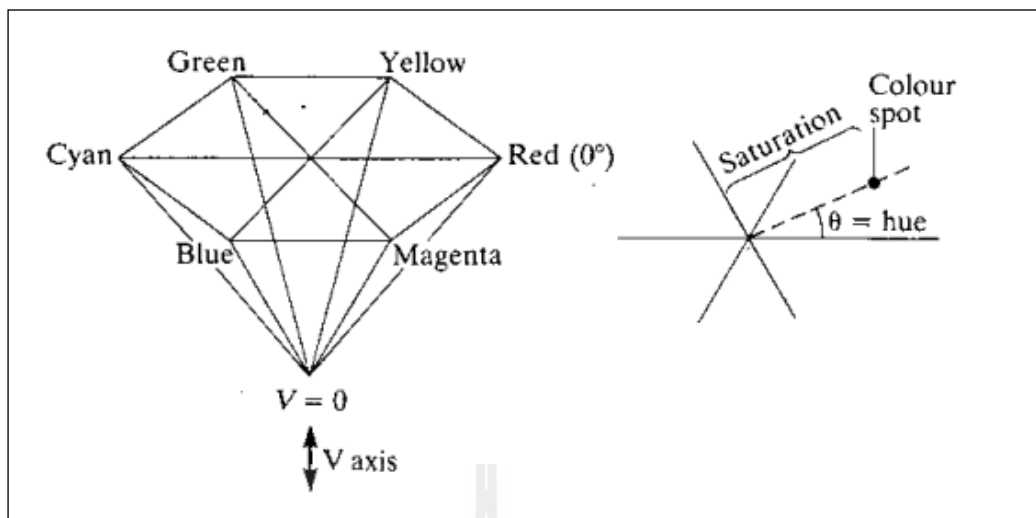
*green green min (red, green, blue)*

*blue blue min (red, green, blue)*

(2.1)







รูปที่ 2.3 แสดงระบบสี HSV

จากลักษณะโมเดลของระบบ Hue พบว่าจะมีค่าอย่างน้อยหนึ่งค่าที่จะเท่ากับ 0 แต่ถ้ามียี่สองค่าเท่ากับ 0 แล้ว hue จะเป็นมุมของสี (ค่าสี) มีค่าเป็นไปตามสีที่สามและถ้าทั้งสามสีมีค่าเท่ากับ 0 แล้วจะทำให้ไม่มีค่าของ Hue หรือสีที่ได้จะมีค่าเท่ากับสีขาวนั่นเอง ตัวอย่างเช่น จอภาพขาว-ดำ ถ้าเกิดมีสีใดสีหนึ่งมีค่าเท่ากับ 0 จะทำให้ค่าสีที่ได้เป็นไปตามสีที่เหลือการให้นำหนักในการพิจารณาเมื่อสีแดงมีค่าเท่ากับ 0

$$\frac{(240 \times blue_h) + (120 \times green_h)}{blue_h + green_h} \quad (2.2)$$

Saturation คือความบริสุทธิ์ของสีซึ่งถ้า Saturation มีค่าเท่ากับ 0 แล้วสีที่ได้จะไม่มี Hue ซึ่งจะเป็นสีขาวล้วนแต่ถ้า Saturation มีค่าเท่ากับ 255 แสดงว่าจะไม่มีแสงสีขาวผสมอยู่เลย

Saturation สามารถคำนวณได้ดังนี้

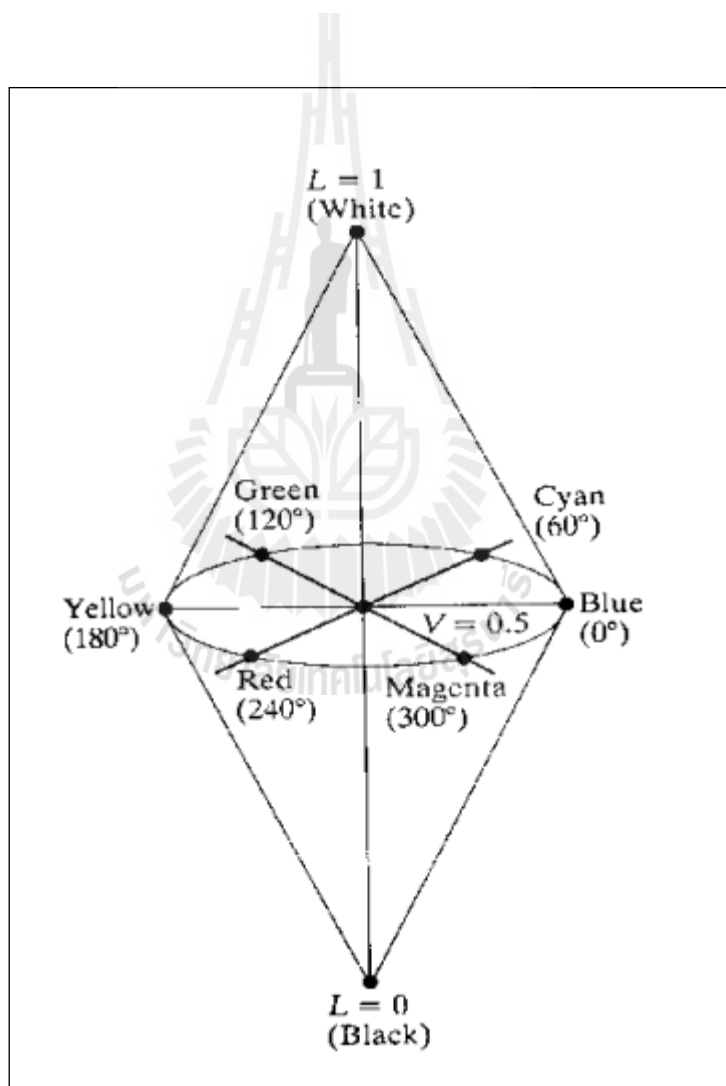
$$Saturation = \frac{\max(red, green, blue) - \min(red, green, blue)}{\max(red, green, blue)} \quad (2.3)$$

Value คือความสว่างของสี ซึ่งสามารถวัดได้โดยค่าความเข้มของความสว่างของแต่ละสีที่ประกอบกันสามารถคำนวณได้จาก

$$value = \max(\text{red}, \text{green}, \text{blue}) \quad (2.4)$$

### 2.3.3 ระบบสีแบบ HLS

ระบบสีแบบ HLS (Hue lightness saturation) พัฒนาโดย Teletromix Incorporated จะมีลักษณะคล้ายกับ HSV ดังนั้นชื่อของระบบจะขึ้นอยู่กับ Hue Lightness และ Saturation



รูปที่ 2.4 แสดงระบบสี HLS

Hue คือค่าของสีหลักซึ่งมีสีน้ำเงินอยู่ที่ 0 องศา สีเขียวอยู่ที่ 120 องศาและสีแดงอยู่ที่ 240 องศา Lightness คือค่าความสว่างซึ่งจะมีค่าเปลี่ยนแปลงตามแนวแกน L โดยที่  $L = 0$  จะเป็นสีดำ  $L = 1$  จะเป็นสีขาว สามารถคำนวณได้ดังนี้

$$lightness = \frac{\max(red, green, blue) + \min(red, green, blue)}{2} \quad (2.5)$$

CMY (Cyan Magenta Yellow) เป็นระบบสีที่พัฒนาขึ้นมาใช้สำหรับการพิมพ์ภาพสีโดยมีสีหลักคือสี Cyan Magenta และ Yellow ซึ่งเรียกว่า Subtractive primaries Color (สีแดง เขียวและน้ำเงิน เรียกว่า Additive primaries Color) ระบบสีแบบ CMY สามารถหาได้โดยการนำเอาสีในระบบ RGB ลบกับสีขาวดังนี้คือ

$$\begin{aligned} C &= 1 - R \\ M &= 1 - G \\ Y &= 1 - B \end{aligned} \quad (2.6)$$

ระบบสี CMY จะนำไปใช้สำหรับการพิมพ์ภาพสีแต่ยังไม่ดีเท่าที่ควรเนื่องจากไม่ยังไม่สามารถสร้างสีดำได้อย่างถูกต้อง ดังนั้นจึงมีการใช้ระบบ CMYK แทน โดย  $K = \min(C, M, Y)$  K เป็นสีที่ 4 แทนสีดำ

$$\begin{aligned} C &= C - K \\ M &= M - K \\ Y &= Y - K \end{aligned} \quad (2.7)$$

#### 2.3.4 ระบบสีแบบ YUV

ระบบสีแบบ YUV ใช้สำหรับโทรทัศน์แบบ PAL และ SECAM ซึ่งยังมีใช้อยู่ในหลาย ๆ ประเทศโดย Y คือค่าความสว่างของภาพ ส่วนสัญญาณ U และ V เป็นสัญญาณที่เก็บค่าสีของภาพ ต่อมาได้มีระบบ YIQ มาใช้แทนเนื่องจากพบว่าสัญญาณ I และ Q สามารถลด Bandwidth ได้มากกว่าสัญญาณ U และ V ในขณะที่ได้ภาพที่มีคุณภาพเท่ากัน

### 2.3.5 ระบบสีแบบ YIQ

เป็นระบบที่ใช้ใน TV Broadcasting สำหรับ NTSC ประโยชน์หลักก็เพื่อให้ใช้งาน  
ได้กับโทรทัศน์แบบขาว-ดำ โดยที่  $y$  คือความสว่างของภาพ ส่วน  $I$  และ  $Q$  จะเป็นสัญญาณที่  
เข้ารหัสสีของภาพไว้ดังนั้นสำหรับโทรทัศน์ขาว-ดำ นั้นสามารถใช้ค่า  $Y$  ค่าเดียวก็สามารถได้ภาพที่  
สมบูรณ์

### 2.3.6 ระบบสีแบบ XYZ

เป็นระบบสีที่ CIE ได้กำหนดให้มีขึ้นเป็นมาตรฐานเนื่องจากในระบบสี RGB  
ยังไม่สามารถสร้างสีที่เป็นไปได้ทั้งหมดดังนั้นจึงได้มีตั้งระบบสี XYZ ซึ่งเป็นระบบสีที่สมมุติขึ้น

## 2.4 การกรองข้อมูลภาพ

การกรองข้อมูลภาพ (Image Filtering) คือการนำภาพไปผ่านตัวกรองสัญญาณเพื่อให้ได้  
ภาพผลลัพธ์ออกมา ภาพผลลัพธ์ที่ได้จะมีคุณสมบัติแตกต่างจากภาพเริ่มต้น วัตถุประสงค์หลักของ  
การกรองข้อมูลภาพคือการเน้น (enhance) หรือลดทอน (attenuate) คุณสมบัติบางประการของภาพ  
เพื่อให้ได้ภาพที่มีคุณสมบัติตามต้องการ

การกรองข้อมูลภาพคือการประมวลผลภาพอย่างหนึ่งที่สำคัญมาก เนื่องจากการใช้งาน  
จริง ภาพที่ได้มามักมีสัญญาณรบกวน หรือสัญญาณไม่พึงประสงค์อื่น ๆ ปะปนอยู่ด้วยการกรอง  
ข้อมูลภาพสามารถปรับปรุงให้ภาพมีคุณสมบัติที่ดีขึ้น เหมาะแก่การประมวลผลในขั้นต่อไป

องค์ประกอบสำคัญของการกรองข้อมูลภาพคือตัวกรอง หากเปรียบเทียบเป็นสัญญาณไฟฟ้า  
ที่มีความถี่ต่าง ๆ ผสมกันอยู่ ตัวกรองก็คือวงจรไฟฟ้าที่ทำหน้าที่เลือกหรือกรองให้สัญญาณไฟฟ้าที่  
มีความถี่ในช่วงที่ต้องการผ่านออกไปได้ คุณสมบัติของตัวกรองคือตัวกำหนดคุณสมบัติของภาพ  
ผลลัพธ์

เราอาจมองข้อมูลของภาพ ๆ หนึ่งให้เป็นสัญญาณ ๆ หนึ่งได้ ด้วยการกำหนดให้ระดับ  
ความเข้มแสงของแต่ละจุดคือขนาด (amplitude) ของสัญญาณ ณ ตำแหน่งนั้น ๆ ข้อแตกต่าง  
ระหว่างสัญญาณไฟฟ้ากับภาพคือ

1. ขนาดของสัญญาณไฟฟ้าคือค่าแรงดันหรือกระแส แต่ขนาดของข้อมูลภาพคือระดับ  
ความเข้มแสงของจุดภาพ
2. การเปลี่ยนแปลงของสัญญาณไฟฟ้าเป็นการเปลี่ยนแปลงเทียบกับเวลาความถี่ของ  
สัญญาณไฟฟ้าถูกกำหนดโดยอัตราการเปลี่ยนแปลงของขนาดของตัวกรอง
3. สัญญาณไฟฟ้าเป็นสัญญาณมิติเดียว ( amplitude vs time) แต่ภาพเป็นสัญญาณ 2 มิติ  
(intensity vs X & Y)

ตัวกรองคือระบบ ๆ หนึ่งซึ่งรับสัญญาณเข้า (input) ประมวลผลสัญญาณ และส่งสัญญาณออก (output) โดยทั่วไปตัวกรองจะถูกสร้างให้เป็นระบบเชิงเส้น (linear system) เนื่องจากออกแบบได้ง่าย และมีประสิทธิภาพดี ปัจจุบันมีทฤษฎีและเทคนิคมากมายเกี่ยวกับการออกแบบตัวกรองสัญญาณแบบเชิงเส้น

ในการกรองข้อมูลภาพ เรามักพิจารณาว่าภาพคือสัญญาณ 2 มิติที่ประกอบขึ้นจากสัญญาณความถี่ต่าง ๆ ผสมกันอยู่ในสัดส่วนที่ต่างกัน การออกแบบตัวกรองจึงเป็นการกำหนดว่าเราต้องการกำจัดสัญญาณความถี่ใดออกไป (หรือต้องการเลือกสัญญาณความถี่ใดบ้าง) หากผู้อ่านมีความรู้พื้นฐานเกี่ยวกับการกรองสัญญาณไฟฟ้า ก็จะสามารถทำความเข้าใจเกี่ยวกับการกรองข้อมูลภาพได้ไม่ยาก เพราะการกรองข้อมูลภาพคือส่วนขยายของความรู้เดิมให้รองรับการประมวลผลสัญญาณ 2 มิติ

#### 2.4.1 ประเภทของตัวกรอง

ตัวกรองแบ่งออกได้เป็น 4 ประเภทตามลักษณะการเลือกความถี่คือ

2.4.1.1 ตัวกรองความถี่ต่ำผ่าน (Low-pass Filter)

2.4.1.2 ตัวกรองความถี่สูงผ่าน (High-pass Filter)

2.4.1.3 ตัวกรองแถบความถี่ผ่าน (Band-pass Filter)

2.4.1.4 ตัวกรองหยุดแถบความถี่ (Band-stop Filter)

ค่าพารามิเตอร์หลักในการกำหนดคุณสมบัติของตัวกรองคือ ค่าความถี่คัตออฟ (cut-off frequency) ความถี่คัตออฟคือความถี่ที่ระบุจุดตัดของสัญญาณว่าจะให้ผ่าน หรือไม่ผ่าน ตัวอย่างเช่น ตัวกรองความถี่ต่ำผ่านที่มีค่าความถี่คัตออฟเท่ากับ 1,000 เฮิรตซ์จะยอมให้สัญญาณที่มีความถี่ต่ำกว่า 1,000 เฮิรตซ์ผ่านไปได้ แต่จะไม่ยอมให้สัญญาณที่มีความถี่สูงกว่า 1,000 เฮิรตซ์ผ่าน

สำหรับตัวกรองความถี่สูงผ่านจะทำงานตรงข้ามกับตัวกรองความถี่ต่ำผ่าน คือ ไม่ยอมให้สัญญาณที่มีความถี่ต่ำกว่าความถี่คัตออฟผ่านไปได้ แต่จะยอมให้ความถี่ที่สูงกว่าความถี่คัตออฟผ่านได้

วงจรถ่ายแถบความถี่ผ่านยอมให้สัญญาณในช่วงความถี่หนึ่งผ่านไปได้ หากสัญญาณมีความถี่อยู่นอกช่วงจะถูกลดทอนหรือไม่ยอมให้ผ่านไป สำหรับวงจรถ่ายแถบความถี่จะมีลักษณะการทำงานที่ตรงข้ามกันคือจะลดทอนสัญญาณที่มีความถี่ในช่วงที่กำหนดลดลง และจะผ่านความถี่ที่อยู่นอกช่วง

ในการกรองสัญญาณใด ๆ เราจะต้องทราบความถี่ หรือช่วงความถี่ของสัญญาณที่เราต้องการและสัญญาณที่เราไม่ต้องการ จากนั้นเราจะเลือกตัวกรองที่เหมาะสมมาใช้เพื่อกำจัดสัญญาณที่ไม่ต้องการออก และหรือเน้นสัญญาณที่ต้องการให้เด่นชัดยิ่งขึ้น ตัวอย่างเช่นสัญญาณรบกวน

## 2.4.2 การกรองโดยการเฉลี่ยจากหลายภาพ

หากเรามีชุดของภาพคุณภาพต่ำหลาย ๆ ภาพซึ่งถ่ายจากมุมมองเดียวกัน เราสามารถสร้างภาพใหม่ที่มีคุณภาพสูงกว่าจากชุดภาพนั้นได้ หากสัญญาณรบกวนเกิดขึ้นแบบสุ่มภาพที่เก็บแต่ละครั้งย่อมมีลักษณะแตกต่างกัน หากความเข้มแสงของจุดในภาพหนึ่งถูกรบกวน เราสามารถนำข้อมูลความเข้มแสงของจุด จากภาพอื่น ณ ตำแหน่งเดียวกันมาแทน แต่ละจุดในภาพผลลัพธ์ที่ได้จะเกิดจากการเฉลี่ย (หรือเลือก) จากจุดที่ตรงกันของภาพต่าง ๆ ในชุดภาพ

### 2.4.2.1 การกรองข้อมูลภาพโดยใช้ค่าเฉลี่ยแบบคณิตศาสตร์ (mean filtering)

วิธีการนี้จะใช้ค่าเฉลี่ยแบบคณิตศาสตร์ของจุดทั้งหมด หากมีภาพขนาด  $N \times M$  ทั้งหมด  $K$  ภาพ เราสามารถคำนวณหาภาพใหม่ได้ดังนี้

$$\hat{I}(x, y) = \frac{1}{K} \sum_{j=1}^K I_j(x, y) \quad (2.8)$$

โดยที่  $\hat{I}(x, y)$  = ความเข้มแสงของจุด ณ ตำแหน่ง  $(x, y)$  ในภาพผลลัพธ์

### 2.4.2.2 การกรองข้อมูลภาพโดยใช้ค่ามัธยฐาน (median filtering)

วิธีการนี้จะนำเอาความเข้มแสงของจุดที่ตรงกันในภาพต่าง ๆ มาเรียงลำดับ (sort) จากน้อยไปหามาก จากนั้นจะเลือกค่าที่อยู่ตรงกลางไปใช้ หากจำนวนภาพทั้งหมดเป็นจำนวนคู่ค่าทั้งสองที่อยู่ตรงกลางจะนำมาหาค่าเฉลี่ย วิธีการนี้จะต้องใช้การเรียงลำดับซึ่งเป็นกระบวนการที่ใช้เวลาในการคำนวณสูง แต่ข้อดีคือไม่สูญเสียความคมชัด

ภาพที่ 1	ภาพที่ 2	ภาพที่ 3	ผลลัพธ์
1 2 1 3	2 3 4 3	3 2 1 4	2 2 1 3
4 2 2 1	5 3 4 1	2 1 4 0	4 2 4 1
0 1 1 3	3 2 4 2	1 4 2 0	1 2 2 2
2 2 1 1	1 3 1 2	2 4 0 2	2 3 1 2

รูปที่ 2.5 แสดงตัวอย่างการกรองข้อมูลภาพโดยใช้ค่ามัธยฐาน

### 2.4.2.3 การกรองข้อมูลภาพโดยใช้ค่าฐานนิยม (modal filtering)

วิธีการนี้คล้ายกับวิธีใช้ค่ามัธยฐาน แต่ไม่ใช้การเรียงลำดับข้อมูล ระดับความเข้มแสงที่ใช้บ่อยที่สุดจะถูกเลือกไปใช้ วิธีนี้เสมือนการ โหวตลงคะแนนเสียง ผู้ที่ได้คะแนนเสียงสูงที่สุดคือผู้ชนะ วิธีนี้เหมาะสำหรับการลดสัญญาณรบกวนที่เกิดขึ้นไม่บ่อย

ภาพที่ 1	ภาพที่ 2	ภาพที่ 3	ผลลัพธ์
1 2 1 3	2 3 4 3	3 2 1 4	2 2 1 3
4 2 2 1	5 3 4 1	2 1 4 0	4 2 4 1
0 1 1 3	3 2 4 2	1 4 2 0	1 2 2 2
2 2 1 1	1 3 1 2	2 4 0 2	2 3 1 2

รูปที่ 2.6 แสดงตัวอย่างการกรองข้อมูลภาพโดยใช้ค่าฐานนิยม

## 2.5 การกรองโดยใช้หน้าต่าง

การกรองข้อมูลภาพวิธีนี้จะใช้หน้าต่างในการกำหนดขอบเขตของการพิจารณาเพื่อหา ระดับความเข้มแสงของจุดต่าง ๆ ในภาพผลลัพธ์ ความเข้มแสงของจุดที่อยู่รอบ ๆ จุดกึ่งกลางของ หน้าต่างจะถูกนำมาหาค่าเฉลี่ย ค่าเฉลี่ยที่ได้คือค่าความเข้มแสงของจุดในภาพผลลัพธ์ หน้าต่างจะถูก เลื่อนไปยังตำแหน่งต่าง ๆ ในภาพจนครบทุกจุด

ภาพเริ่มต้น	ภาพผลลัพธ์
0 0 0 0 0 0	
0 1 2 1 2 0	(A) B C D
0 2 3 9 1 0	E F G H
0 1 3 2 1 0	I J K L
0 0 0 0 0 0	

รูปที่ 2.7 แสดงตัวอย่างการกรองโดยใช้หน้าต่าง

จากรูปที่ 2.7 จะเห็นว่า มีหน้าต่างขนาด  $3 \times 3$  ครอบอยู่ที่มุมบนด้านซ้ายของภาพเริ่มต้น ความเข้มแสง ณ จุดกึ่งกลางของหน้าต่างมีค่าเท่ากับ 1 ความเข้มแสงของจุดภาพในภาพผลลัพธ์ ณ ตำแหน่งที่ตรงกับกึ่งกลางของหน้าต่างที่ครอบอยู่บนภาพเริ่มต้น (จุด A) สามารถคำนวณได้จาก ค่าเฉลี่ยความเข้มแสงของทุกจุดในหน้าต่าง การหาค่าเฉลี่ยสามารถทำได้ 3 แบบคือ การหาค่าเฉลี่ยแบบคณิตศาสตร์ การหาค่าเฉลี่ยแบบมัธยฐาน และการหาค่าเฉลี่ยแบบฐานนิยม

การหาค่าเฉลี่ยแบบคณิตศาสตร์ทำได้โดยการหาผลรวมของค่าความเข้มแสงของจุดทุกจุดในหน้าต่าง แล้วหารด้วยจำนวนจุดทั้งหมดในหน้าต่าง จากตัวอย่างในรูปที่ 5.5 ความเข้มแสงที่จุด A มีค่าเท่ากับ  $(0+0+0+0+1+2+0+2+3) / 9 = 8/9$  ค่าความเข้มแสงที่จุดอื่น ๆ สามารถคำนวณได้โดยการเลื่อนหน้าต่างให้จุดกึ่งกลางตรงกับจุดที่ต้องการหาค่า

การหาค่าเฉลี่ยแบบมัธยฐานทำได้โดยการนำค่าทั้งหมดในตารางมาเรียงลำดับ (sort) จากน้อยไปหามาก (หรือจากมากไปหาน้อยก็ได้) จากนั้นจะเลือกค่าที่อยู่ตรงกลางของลำดับเป็นค่าความเข้มแสงของจุดในภาพผลลัพธ์ หากจำนวนจุดในหน้าต่างเป็นจำนวนคู่ ผลลัพธ์จะคำนวณได้จากการเฉลี่ยค่าระหว่างจุดกึ่งกลางทั้งสอง จากตัวอย่างในรูปที่ 5.5 เมื่อเรียงลำดับความเข้มแสงจะได้ลำดับดังนี้ (0 0 0 0 1 2 2 3) ค่าที่อยู่ตรงกลางคือ 0 ดังนั้นความเข้มแสงที่จุด A มีค่าเท่ากับ 0

การหาค่าเฉลี่ยแบบฐานนิยมทำได้โดยการเลือกระดับความเข้มแสงที่ใช้บ่อยที่สุดในหน้าต่างมาเป็นคำตอบ ปัญหาที่อาจเกิดจากการใช้วิธีนี้คือ มีระดับความเข้มแสงที่ใช้บ่อยที่สุดมากกว่า (มีหลายคำตอบ) วิธีการแก้ไขคือการหาค่าเฉลี่ย หรือเปลี่ยนไปใช้การหาค่าเฉลี่ยแบบมัธยฐาน จากตัวอย่างในรูป 2.7 ค่าความเข้มแสงที่ใช้บ่อยที่สุดคือ 0 ดังนั้นความเข้มแสงที่จุด A มีค่าเท่ากับ 0

นอกจากการหาค่าผลลัพธ์โดยวิธีหาค่าเฉลี่ยทั้ง 3 ดังที่ได้กล่าวมาแล้ว ยังมีวิธีการหาผลลัพธ์อีกวิธีหนึ่งคือการหาค่าเฉลี่ยจากจุด  $k$  จุดที่มีค่าความเข้มใกล้เคียงกับค่าความเข้มแสงของจุดกึ่งกลางของหน้าต่าง วิธีนี้เรียกว่า  $k$ -closest averaging การคำนวณหาผลลัพธ์เริ่มจากการนำค่าความเข้มแสงของทุกจุดในหน้าต่างมาเรียงลำดับจากน้อยไปหามาก จากนั้นค่าที่อยู่รอบ ๆ ค่าของจุดกึ่งกลางหน้าต่างจำนวน  $k$  ค่าจะถูกเลือกมาเพื่อหาค่าเฉลี่ย ค่าเฉลี่ยนี้คือความเข้มแสงของจุดในภาพผลลัพธ์ ในการหาค่าเฉลี่ย อาจนำค่าของความเข้มสีที่จุดกึ่งกลางมาคิดด้วยก็ได้ จากตัวอย่างในรูปที่ 2.7 เมื่อเรียงลำดับความเข้มแสงจะได้ลำดับดังนี้ (0 0 0 0 1 2 2 3) หากกำหนดให้  $k = 4$  และไม่นำค่าที่จุดกึ่งกลาง (1) มาคิด ค่าความเข้มแสงที่จุด A มีค่าเท่ากับ  $(0+0+2+2)/4 = 1$  หากนำค่าที่จุดกึ่งกลางมาคิด ค่าความเข้มแสงที่จุด A จะมีค่าเท่ากับ  $(0+0+1+2+2)/5 = 1$



1 2 2 1	0 1 1 0	0 0 0 0	1 2 1 3
1 2 2 2	1 2 2 1	0 2 2 1	2 2 3 1
1 2 2 1	0 2 1 0	0 0 0 0	1 4 2 1
(ก)	(ข)	(ค)	(ง)

รูปที่ 2.8 แสดงการกรองข้อมูลภาพโดยใช้หน้าต่าง (ก) ใช้การเฉลี่ยแบบคณิตศาสตร์ (แสดงผลลัพธ์หลังการปิดเศษ) (ข) ใช้มัธยฐาน (ค) ใช้ฐานนิยม (ใช้มัธยฐานแทนสำหรับจุดที่มีปัญหา) (ง) ใช้ k-closest averaging (แสดงผลลัพธ์หลังการปิดเศษ)

## 2.6 การกรองโดยวิธีคอนโวลูชัน

วิธีการกรองข้อมูลภาพที่กล่าวมาส่วนใหญ่อาศัยหลักของการหาค่าเฉลี่ย โดยอาจเป็นการหาค่าเฉลี่ยของจุดเดียวกันจากภาพหลาย ๆ ภาพ หรืออาจเป็นการหาค่าเฉลี่ยจากจุดต่าง ๆ ที่อยู่รอบ ๆ จุดที่เราสนใจ เนื่องจากการหาค่าเฉลี่ยเป็นการลดการเปลี่ยนแปลงของข้อมูลวิธีการที่ผ่านมายังใช้ได้ดีกับการกำจัดสัญญาณรบกวนที่เป็นสัญญาณความถี่สูง ดังที่ได้กล่าวมาแล้วว่า การกรองสัญญาณมีวัตถุประสงค์เพื่อนำคุณสมบัติบางอย่างที่ต้องการในภาพให้เด่นชัดขึ้น ในขณะที่ลดทอนคุณสมบัติที่ไม่ต้องการลง หากเราต้องการเน้นการเปลี่ยนแปลงของระดับความเข้มของจุดต่าง ๆ ภายในภาพให้เด่นชัดขึ้น ในที่นี้จะเสมือนกับการกรองสัญญาณความถี่สูงผ่าน เราจะไม่สามารถใช้วิธีการหาค่าเฉลี่ยได้ วิธีที่สามารถนำมาใช้ได้คือการคอนโวลูชัน (convolution)

### 2.6.1 การคอนโวลูชัน

ในการประมวลผลภาพ การคอนโวลูชันคือการกระทำกันระหว่างเทมเพลต (template) กับภาพ (image) เทมเพลตคือเมตริกซ์ขนาด  $n \times m$  ของชุดตัวเลขที่จะนำไปซ้อนทับภาพที่ตำแหน่งต่าง ๆ เพื่อหาผลลัพธ์ของการคอนโวลูชัน ถ้ากำหนดให้เทมเพลต  $T(x, y)$  เป็นเทมเพลตขนาด  $n \times m$  และภาพ  $I(X, Y)$  มีขนาด  $N \times M$  การคอนโวลูชันระหว่างเทมเพลตกับภาพสามารถแสดงได้ดังสมการต่อไปนี้

$$I'(X, Y) = T * I = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} T(i, j) \cdot I(x-i, Y-j) \quad (2.9)$$

โดย  $I'(X, Y)$  คือ ภาพผลลัพธ์จากการคอนโวลูชัน

จากสมการที่ 2.9 จะเห็นว่าระดับความเข้มแสง ณ จุด  $(X, Y)$  ในภาพผลลัพธ์ได้จากการหาผลรวมของผลคูณของระหว่างค่าในเทมเพลตกับค่าระดับความเข้มแสงของภาพในบริเวณที่เทมเพลตซ้อนทับอยู่ จากสมการ ตัวชี้ตำแหน่งจุดในภาพ  $(X-i, Y-j)$  แสดงให้เห็นว่ามีการพลิกเทมเพลตทางแกนนอน และแกนตั้ง สมการที่ 2.9 แสดงการคอนโวลูชันที่ไม่ต้องมีการพลิกเทมเพลต ซึ่งวิธีการนี้มีชื่อที่แท้จริงว่า cross-correlation และเป็นที่ยอมรับใช้ในการประมวลผลภาพ

$$I'(X, Y) = T * I = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} T(i, j) \cdot I(x+i, Y+j) \quad (2.10)$$

ขั้นตอนของการคอนโวลูชันประกอบด้วยการเลื่อน บวก และคูณ เราสามารถใช้การคอนโวลูชันในการประมวลผลภาพได้ในหลายลักษณะ เช่น กรองสัญญาณภาพการหาขอบภาพ (edge detection) หรือการหารูปทรงของวัตถุในภาพ เป็นต้น

โดยทั่วไป ในการคอนโวลูชัน เราจะไม่ยอมให้มีการเลื่อนเทมเพลตออกนอกขอบเขตของภาพ ดังนั้นถ้าเทมเพลตมีขนาดใหญ่กว่า  $1 \times 1$  ภาพผลลัพธ์จะมีขนาดเล็กกว่าภาพเริ่มต้นเสมอ ตัวอย่างเช่น การคอนโวลูชันระหว่างภาพขนาด  $4 \times 5$  กับ เทมเพลตขนาด  $2 \times 2$  ต่อไปนี้

เทมเพลต		ภาพเริ่มต้น		ภาพผลลัพธ์
		1 1 3 3 4		2 5 7 6 *
1 0		1 1 4 4 3		2 4 7 7 *
0 1	*	2 1 3 3 3	=	3 2 7 7 *
		1 1 1 4 4		* * * * *

รูปที่ 2.9 แสดงผลของการเติมค่าศูนย์รอบภาพเริ่มต้นก่อนทำการคอนโวลูชัน

จะให้ภาพผลลัพธ์ที่มีขนาด  $3 \times 4$  จากตัวอย่างข้างต้น ค่าความเข้มสี 3 ในภาพผลลัพธ์ได้จากการหาผลรวมของผลคูณระหว่างเทมเพลตกับภาพในบริเวณที่แรเงาซึ่งมีค่าเท่ากับ  $(1 \times 2) + (0 \times 1) + (0 \times 1) + (1 \times 1) = 3$

จากตัวอย่างพบว่าจุดมุมบนซ้ายของเทมเพลตคือจุดอ้างอิงในการกำหนดจุดในภาพผลลัพธ์ ในความเป็นจริงแล้ว เราสามารถเลือกจุดใด ๆ ในเทมเพลตให้เป็นจุดอ้างอิงก็ได้ การเลือกเอาจุดกึ่งกลางของเทมเพลตที่มีความกว้างและสูงเป็นจำนวนคี่ (เช่น เทมเพลตขนาด 3 x 3, 3 x 5, 5 x 5 และ 7 x 7 เป็นต้น) เป็นจุดอ้างอิงนับว่ามีความเหมาะสมยิ่ง อย่างไรก็ตาม ในแง่ของการเขียนโปรแกรมแล้ว การใช้จุดมุมเป็นจุดอ้างอิงจะลดความซับซ้อนในการเขียนโปรแกรมลง เนื่องจากไม่มีปัญหาสำหรับการคอนโวลูชันโดยใช้เทมเพลตที่มีขนาดไม่คงที่ ดังนั้นในหนังสือนี้จะใช้จุดมุมบนซ้ายเป็นจุดอ้างอิงในการทำคอนโวลูชันทุกครั้ง เว้นแต่มีการกำหนดให้เป็นอย่างอื่น

การคอนโวลูชันที่ไม่ยอมให้เทมเพลตเลื่อนออกนอกบริเวณขอบภาพเรียกว่าการคอนโวลูชันแบบไม่เป็นรายคาบ (aperiodic convolution) วิธีการนี้จะได้ภาพที่มีขนาดเล็กลง หากต้องการคงขนาดภาพไว้ จะต้องใช้การคอนโวลูชันแบบเป็นรายคาบ (periodic convolution) การคอนโวลูชันแบบนี้เปรียบเสมือนการม้วนภาพให้ขอบซ้ายมาชนกับขอบขวา และม้วนให้ขอบบนมาชนกับขอบล่าง เมื่อเทมเพลตเลื่อนตกขอบข้างใดข้างหนึ่ง ๆ ส่วนของเทมเพลตที่เลยขอบก็จะไปทับกับขอบภาพอีกด้านหนึ่ง

วิธีการอย่างง่ายที่ทำให้ภาพผลลัพธ์มีขนาดเท่ากับภาพเริ่มต้นคือการเติมค่าศูนย์บริเวณรอบ ๆ ภาพเริ่มต้น เพื่อให้ภาพเริ่มต้นมีขนาดใหญ่ขึ้น หลังจากการคอนโวลูชันจะได้ภาพผลลัพธ์ที่มีขนาดเท่ากับภาพเริ่มต้นก่อนมีการชดเชย รูปที่ 2.10 แสดงการคอนโวลูชันที่มีการเติมค่าศูนย์ให้กับภาพเริ่มต้น

ภาพเริ่มต้น	เติมศูนย์	เทมเพลต	ภาพผลลัพธ์
	0 0 0 0 0		
1 2 3	0 1 2 3 0	1 0 0	6 8 3
4 5 6	0 4 5 6 0	* 0 1 0	= 12 15 8
7 8 9	0 7 8 9 0	0 0 1	7 12 14
	0 0 0 0 0		

รูปที่ 2.10 แสดงการคอนโวลูชันที่มีการเติมค่าศูนย์ให้กับภาพเริ่มต้น

แม้เป็นกระบวนการประมวลผลภาพที่ง่าย แต่การคอนโวลูชันต้องใช้เวลาในการคำนวณสูง โดยเฉพาะอย่างยิ่งการคอนโวลูชันระหว่างภาพและเทมเพลตที่มีขนาดใหญ่ หากภาพมีขนาด  $M \times M$  และเทมเพลตมีขนาด  $n \times n$  จะต้องมีการคูณถึง  $M^2 n^2$  ครั้ง ถ้า  $M=512$  และ  $n=16$  จะต้องมีการคูณประมาณ 32 ล้านครั้ง การคำนวณที่มากขนาดนี้ทำให้ไม่สามารถประมวลผลภาพเคลื่อนไหวในเวลาจริง (real-time) ได้ เว้นแต่จะมีฮาร์ดแวร์ที่ออกแบบมาเฉพาะ สำหรับภาพและเทมเพลตที่มีขนาดใหญ่ ( $M \geq 512$  และ  $N \geq 32$ ) การแปลงภาพและเทมเพลตให้อยู่ในรูปของข้อมูลในโดเมนความถี่ (frequency domain) จะช่วยลดการคำนวณลงได้อย่างมาก จากตัวอย่างข้างต้น การคอนโวลูชันในโดเมนความถี่จะลดจำนวนครั้งของการคูณลงเหลือเพียง 256,000 ครั้ง เท่านั้น การคอนโวลูชันในโดเมนความถี่จะได้กล่าวถึงในหัวข้อถัดไป

### 2.6.2 เทมเพลตสำหรับกรองความถี่ต่ำผ่าน

เทมเพลตขนาด  $3 \times 3$  ต่อไปนี้สามารถใช้เพื่อลดการเปลี่ยนแปลงอย่างฉับพลันของค่าความเข้มแสงในภาพ  $[1 \ 1 \ 1, 1 \ 1 \ 1, 1 \ 1 \ 1]$

ผลของการคอนโวลูชันกับเทมเพลตนี้จะเหมือนกับการหาผลรวมของจุดภาพทั้ง 9 จุดที่เทมเพลตซ้อนทับอยู่ การเปลี่ยนแปลงใด ๆ ในบริเวณดังกล่าวจะถูกเฉลี่ยให้มีความราบเรียบ ผลที่ได้คือสัญญาณที่มีการเปลี่ยนแปลง (เช่นสัญญาณรบกวนความถี่สูง) จะถูกลดทอน ภาพที่ได้จะมีความคมลดลง คุณสมบัติเช่นนี้เปรียบได้กับการกรองสัญญาณความถี่ต่ำผ่านเทมเพลตการกรองความถี่ต่ำผ่านที่นิยมใช้อีกแบบหนึ่งคือ  $[1 \ 3 \ 1, 3 \ 16 \ 3, 1 \ 3 \ 1]$

เทมเพลตนี้จะเน้นความสำคัญของจุดที่อยู่ตรงกลางเทมเพลตเป็นพิเศษ โดยจะให้จุดกลางมีน้ำหนัก 50% ของทั้งหมด และให้น้ำหนักรวมของจุดทั้ง 4 ที่อยู่ด้านบน ด้านล่าง ด้านซ้าย และด้านขวาของจุดกลางมีค่าเท่ากับ 40% ส่วนจุดมุมทั้ง 4 มีน้ำหนักเพียง 10% โดยจุดที่อยู่ใกล้จุดศูนย์กลางจะมีน้ำหนักมากกว่าจุดที่อยู่ห่างออกไป

### 2.6.3 เทมเพลตสำหรับกรองความถี่สูงผ่าน

สัญญาณความถี่สูงคือสัญญาณที่มีการเปลี่ยนแปลงค่าไปมาอย่างรวดเร็ว ต่างกับสัญญาณความถี่ต่ำ ซึ่งมีการเปลี่ยนแปลงค่าอย่างช้า ๆ หรือไม่เปลี่ยนแปลงเลย การกรองความถี่สูงผ่าน (High pass filter) ก็คือการกรองสัญญาณที่เพิ่มความแรงของสัญญาณที่มีความถี่สูงและลดความแรงของสัญญาณที่มีความถี่ต่ำ เทมเพลตต่อไปนี้ใช้สำหรับการกรองความถี่สูงผ่าน  $[0-1 \ 0, -1 \ 4 \ -1, 0-1 \ 0]$

จะเห็นว่าผลรวมของทุกค่าในเทมเพลตมีค่าเท่ากับศูนย์ ซึ่งหมายความว่า ถ้าวางเทมเพลตนี้ลงบนบริเวณของภาพที่มีค่าความเข้มแสงคงที่ ผลลัพธ์ที่ได้จะมีค่าเป็นศูนย์ อย่างไรก็ตาม ถ้าค่าที่บริเวณตรงกลางแตกต่างกับค่ารอบ ๆ ผลลัพธ์ที่ได้จะแสดงค่าความแตกต่างยิ่งขึ้น

ภาพ	หลังจากการกรองความถี่สูงผ่าน	หลังจากการกรองความถี่ต่ำผ่าน
0 0 0 0 0		
0 1 1 1 0	2 1 2	4 6 4
0 1 1 1 0	1 0 1	6 9 6
0 1 1 1 0	1 0 1	6 9 6
0 1 1 1 0	1 -5 1	11 14 11
0 1 6 1 0	-4 20 -4	11 14 11
0 1 1 1 0	2 -4 2	9 11 9
0 0 0 0 0		

รูปที่ 2.11 แสดงตัวอย่างการกรองภาพด้วยเทมเพลตกรองความถี่สูง และความถี่ต่ำ

จากรูปที่ 2.11 จะเห็นว่าหลังการกรองความถี่สูงผ่านขอบภาพจะเด่นชัด ส่วนที่เป็นค่าคงที่จะกลายเป็นศูนย์ และส่วนที่มีการเปลี่ยนแปลงจาก 1 เป็น 6 ถูกขยายเป็นจาก -4 ไป 20 สำหรับการกรองความถี่ต่ำผ่าน การเปลี่ยนถูกลดทอนลง ภาพผลลัพธ์มีความราบเรียบขึ้น

## 2.6 การหาขอบภาพ (Edge Detection)

การหาขอบภาพ (Edge Detection) เป็นการหาเส้นรอบวัตถุที่อยู่ในภาพ เมื่อทราบเส้นรอบวัตถุ เราจะสามารถคำนวณหาพื้นที่ (ขนาด) หรือรู้จำชนิดของวัตถุนั้นได้ อย่างไรก็ตาม การหาขอบภาพที่ถูกต้องสมบูรณ์ไม่ใช่เป็นเรื่องที่ง่าย โดยเฉพาะอย่างยิ่งการหาขอบของภาพที่มีคุณภาพต่ำ มีความแตกต่างระหว่างพื้นหน้ากับพื้นหลังน้อย หรือมีความสว่างไม่สม่ำเสมอทั่วภาพ

ขอบภาพเกิดจากความแตกต่างของความเข้มแสงจากจุดหนึ่งไปยังอีกจุดหนึ่ง หากต่างนี้มีค่ามาก ขอบภาพก็จะเห็นได้ชัด ถ้าความแตกต่างมีค่าน้อย ขอบภาพก็จะไม่ชัดเจน

### 2.6.1 เทมเพลตสำหรับหาขอบภาพ

ถ้าต้องการหาขอบภาพในแนวนอนอย่างง่าย วิธีการก็คือหาผลต่างระหว่างจุดหนึ่งกับจุดที่อยู่ข้างล่าง (หรือข้างบน) ของจุดนั้น ดังนี้

$$Y_{diff}(x, y) = I(x, y) - I(x, y+1) \quad (2.11)$$

โดยที่  $Y_{diff}$  คือค่าความแตกต่างในแนวแกนตั้ง และ  $I(x,y)$  คือค่าความเข้มแสงของจุดภาพที่ตำแหน่ง  $(x, y)$

ผลของการใช้สมการ (2.11) มีค่าเทียบเท่ากับการคอนโวลูชันภาพด้วยเทมเพลต [1-1] การหาขอบภาพในแนวตั้งก็สามารถหาได้เช่นเดียวกันคือ

$$Y_{diff}(x, y) = I(x, y) - I(x, y) \quad (2.12)$$

โดยที่  $X_{diff}$  คือค่าความแตกต่างในแนวนอน และสมการที่ (2.12) มีให้ผลเทียบเท่ากับการคอนโวลูชันด้วยเทมเพลต [-1 1]

บางครั้งเราต้องการรวมผลต่างของค่าความแตกต่างในแนวแกนนอน และแกนตั้งเข้าด้วยกัน เพื่อที่จะได้มีตัววัดความแรงของขอบภาพ (Gradient Magnitude) เพียงตัวเดียว เนื่องจากค่าความแตกต่างอาจมีค่าเป็นบวกหรือลบ ดังนั้น การบวกค่าความแตกต่างของทั้งสองแกน อาจทำให้ขอบภาพเกิดการหักล้างกันเอง ในทางปฏิบัติ เราจะต้องนำค่าสัมบูรณ์ (Absolute Value) หรือค่ากำลังสอง (Squared Value) ของค่าความแตกต่างของทั้งสองแกนมาบวกกันแทน

นอกจากหาความแรงของขอบภาพแล้ว การหาทิศทางของขอบภาพ (Gradient Direction) ก็มีประโยชน์เช่นกัน การหาทิศทางของขอบภาพสามสามารถทำได้โดยการใช้สมการต่อไปนี้

$$GD(x, y) = \tan^{-1} \left\{ \frac{Y_{diff}(x, y)}{X_{diff}(x, y)} \right\} \quad (2.13)$$

โดยที่  $GD(x,y)$  ทิศทางของขอบภาพที่ตำแหน่ง  $(x,y)$

### 2.6.2 การหาขอบภาพโดยวิธี โซเบล

การหาขอบภาพโดยวิธี โซเบล (Sobel Edge Detection) เป็นการหาขอบภาพโดยใช้เทมเพลตขนาด 3x3 สองเทมเพลต โดยเทมเพลตแรกจะใช้หาค่าความแตกต่างในแนวนอน ( $X_{diff}$ ) และค่าความแตกต่างในแนวตั้ง ( $Y_{diff}$ )

$$\begin{array}{ccc}
 -1 & 0 & 1 \\
 X_{diff} = -2 & 0 & 2 \\
 -1 & 0 & 1
 \end{array}
 \qquad
 \begin{array}{ccc}
 1 & 2 & 3 \\
 Y_{diff} = 0 & 0 & 0 \\
 -1 & -2 & -1
 \end{array}$$

รูปที่ 2.12 แสดงตัวอย่างการหาความแรงของขอบภาพด้วยวิธีโซเบล

ภาพ	$ X_{diff}  +  Y_{diff} $	เทรลโฮลด์ด้วยค่า 12
0 0 0 0 0 0 2 0 3 3		
0 0 0 1 0 0 0 2 4 2	4 6 4 10 14 12 14 4	0 0 0 0 1 1 1 0
0 0 2 0 2 4 3 3 2 3	6 8 10 20 16 12 6 0	0 0 0 1 1 1 0 0
0 0 1 3 3 4 3 3 3 3	4 10 14 10 2 4 2 4	0 0 1 0 0 0 0 0
0 1 0 4 3 3 2 4 3 2	2 12 12 2 2 4 6 8	0 1 1 0 0 0 0 0
0 0 1 2 3 3 4 4 4 3		

รูปที่ 2.13 แสดงตัวอย่างการหาความแรงของขอบภาพด้วยวิธีโซเบล ภาพสุดท้ายได้จาก

การทำเทรลโฮลด์ด้วยค่า 12 แล้วปรับความเข้มแสงให้อยู่ในช่วง 0-1 เราสามารถสร้างเทมเพลตโซเบลที่มีขนาดใหญ่กว่า 3 x 3 เพื่อที่จะให้มีการครอบคลุมพื้นที่มากขึ้นได้

2.6.3 การหาขอบภาพเทมเพลตคี่กรีสอง

หากภาพที่ต้องการหาขอบมีการไล่ระดับความเข้มแสงแบบเป็นเชิงเส้น การใช้เทมเพลตโซเบลในการหาขอบภาพจะไม่สามารถทำได้ ตัวอย่างเช่น ภาพต่อไปนี้ [1 2 3 4 5 6 7 8 9, 1 2 3 4 5 6 7 8 9, 1 2 3 4 5 6 7 8 9, 1 2 3 4 5 6 7 8 9, 1 2 3 4 5 6 7 8 9]

เมื่อหาขอบภาพตามแกนตั้งด้วยเทมเพลต [-1 0 1, -2 0 2, -1 0 1] จะได้ผลลัพธ์ดังนี้ [8 8 8 8 8 8, 8 8 8 8 8 8, 8 8 8 8 8 8] และถ้าคอนโวลูชันด้วยเทมเพลตเดิมอีกครั้งจะได้ [0 0 0 0 0]

การกระทำดังกล่าวจะคล้ายกับการดิฟเฟอเรนทิเอทสมการเส้นตรงแล้วได้ค่าคงที่ และเมื่อดิฟเฟอเรนทิเอทค่าคงที่ก็จะได้ศูนย์

การใช้เทมเพลตคี่กรีสองเช่นเทมเพลตลาปลาเซียนในการหาขอบของภาพที่มีคุณสมบัติดังกล่าวจะสามารถแก้ปัญหาได้ ต่อไปนี้คือตัวอย่างของเทมเพลตลาปลาเซียน  $[0 -1 0, -1 4 -1, 0 -1 0]$  หรือ  $[-1 -1 -1, -1 8 -1, -1 -1 -1]$

เทมเพลตลาปลาเซียนเป็นที่นิยมใช้กันมาก เพราะสามารถกำจัดการไล่โทนสีหรือโทนความเข้มแสงได้ มันสามารถเน้นความเปลี่ยนแปลงได้ดีกว่าวิธีโซเบล แต่มันไม่ให้ข้อมูลเกี่ยวกับทิศทางของขอบ

#### 2.6.4 การหาขอบภาพแบบเป็นลำดับขั้น

ปัญหาในการหาขอบภาพโดยทั่วไปส่วนหนึ่งมักเกิดจากเน้นขอบภาพที่ไม่ชัดเจนและขอบภาพสั้น ๆ ที่ไม่ปะติดปะต่อกัน ขอบภาพเหล่านี้คือสิ่งที่ไม่ต้องการ แต่การใช้วิธีการหาขอบภาพธรรมดาจะไม่สามารถกำจัดขอบภาพลักษณะนั้นได้ วิธีแก้ไขคือใช้การหาขอบภาพแบบเป็นลำดับขั้นซึ่งมีขั้นตอนปฏิบัติดังนี้

1. สร้างภาพใหม่จากภาพเดิมและมีขนาด  $\frac{1}{4}$  ของภาพเดิมโดยค่าความเข้มของแต่ละจุดในภาพเล็กคำนวณได้จากค่าเฉลี่ยของจุดทั้งสี่ที่ตรงกันในภาพใหญ่
2. สร้างภาพที่เล็กลงไปอีกด้วยวิธีเดียวกัน จนกระทั่งภาพที่ได้แสดงเฉพาะขอบภาพที่ต้องการเท่านั้น
3. ทำการหาขอบภาพอาจใช้วิธีโซเบลหรือวิธีอื่น ๆ แล้วทำเทรซโฮลด์
4. ณ ตำแหน่งที่เกิดขอบภาพขึ้นให้ทำการหาขอบภาพบริเวณจุดทั้งสี่ของจุดที่ตรงกันในภาพที่ใหญ่กว่าในอันดับถัดไป
5. ทำซ้ำจนถึงภาพสุดท้ายซึ่งเป็นภาพเริ่มต้นก็จะได้ขอบภาพที่ต้องการ

#### 2.6.5 การตามขอบภาพ (Edge Following)

หากเราทราบจุดใดจุดหนึ่งบนขอบภาพเราสามารถหาจุดข้างเคียงที่เป็นขอบภาพและสามารถวนไปตามขอบภาพจนกลับมายังจุดเริ่มต้นได้ ในบทนี้จะแนะนำวิธีการตามขอบภาพอย่างง่ายดังต่อไปนี้

1. สมมุติให้จุด  $(x, y)$  เป็นจุดใดจุดหนึ่งบนขอบภาพ
2. ตั้งค่าแฟล็กให้จุด  $(x, y)$  ว่าเคยผ่านมาแล้ว
3. กำหนดหาค่าความแรงของขอบของจุดทั้งแปดที่อยู่รอบจุด  $(x, y)$
4. เลือกจุด 3 จุดที่มีค่าความแรงสูงสุด แล้วใส่ไว้ในอะเรย์แบบ 3 คอลัมน์ โดยเรียงตามลำดับความแรงของขอบ



5. เลือกจุดที่มีค่าความแรงสูงสุด แล้วพิจารณาว่าตำแหน่งของจุดอยู่ในทิศทางใดเมื่อเทียบกับจุด  $(x, y)$  โดยกำหนดให้ทิศต่าง ๆ มีค่าดังนี้  $[0, 1, 2, 7, *, 3, 6, 5, 4]$  โดยที่ \* แทนตำแหน่งของจุด  $(x, y)$  หากจุดที่มีค่าความแรงของขอบอยู่ด้านบนเมื่อเทียบกับจุด  $(x, y)$  ดังนั้นทิศทางที่กำหนดให้คือ 1

6. กำหนดให้ค่าทิศทางคือ  $d$

7. ทำซ้ำในข้อ 3 แต่พิจารณาเฉพาะจุดที่อยู่ใน 3 ทิศทาง คือ ทิศทาง  $d$  ทิศทาง  $(d+1) \bmod 8$  และทิศทาง  $(d-1) \bmod 8$

8. หากไม่พบจุดใดเลยที่มีค่าความแรงของขอบสูงพอในทิศทางที่เคลื่อนที่ไปให้ลบจุดนั้นออกจากอะเรย์ และเลือกจุดที่มีความแรงของขอบน้อยกว่าในอันดับถัดไป แล้วทำซ้ำในข้อที่ 3 หากทั้ง 3 จุดในแถวถูกลบออกหมด ให้ถอยกลับไปใช้จุดที่เหลือในแถวถัดไป

9. หยุดเมื่อวนกลับมายังจุดเดิม หรือเมื่อมีการใช้เวลาตามขอบภาพนานเกินไป

#### 2.6.6 การค้นหาขอบภาพโดยใช้ออนุพันธ์อันดับหนึ่ง

วิธีการค้นหาขอบภาพโดยใช้ออนุพันธ์อันดับหนึ่ง (Gradient Operator:  $\nabla$ ) มีหลักการคือ บริเวณขอบของวัตถุในภาพเมื่อ  $P(x, y)$  พังก์ชันของภาพ (เมื่อ  $P(x, y)$  พังก์ชันของภาพ) จะมีค่าเกรเดียนต์ที่สูง การพิจารณาขนาดของเกรเดียนต์ (Gradient Magnitude ( $|\nabla P|$ )) เปรียบเทียบกับค่าอ้างอิง (Threshold(T)) ที่กำหนดขึ้นเมื่อค่าของ  $|\nabla P|$  มีค่ามากกว่าค่าอ้างอิงแสดงว่าจุดดังกล่าวคือขอบของวัตถุที่ปรากฏในภาพที่จุด  $P(x, y)$  การค้นหาขอบของวัตถุโดยใช้ออนุพันธ์อันดับหนึ่งเป็นวิธีแยกส่วนประกอบของภาพและเมื่อความไม่ต่อเนื่องของค่าพิกเซลบริเวณรอยต่อระหว่างวัตถุกับพื้นหลังและค่าอนุพันธ์ย่อยที่ไม่ต่อเนื่องตามทิศทางของเกรเดียนต์ของแนวแกน  $x$  และแกน  $y$  กำหนดค่าได้ตามสมการ

$$\nabla_x P(x, y) = P(x, y) - P(x-1, y) \quad (2.14)$$

$$\nabla_y P(x, y) = P(x, y) - P(x, y-1) \quad (2.15)$$

ขนาดของเกรเดียนต์ของ  $P(x, y)$  กำหนดค่าได้จาก

$$|\nabla P(x, y)| = \sqrt{(\nabla_x P(x, y))^2 + (\nabla_y P(x, y))^2} \quad (2.16)$$

เพื่อให้ง่ายต่อการคำนวณประมาณค่าขนาดของเกรเดียนต์ได้ว่า

$$|\nabla P(x, y)| = |\nabla_x P(x, y)| + |\nabla_y P(x, y)| \quad (2.17)$$

การค้นหามุมองศาที่มียอดค่าของเส้นตรงเช่นวิธี Canny Edge Detection เป็นวิธีการแบ่งออกเป็นช่วงย่อยในแต่ละพิกเซล กำหนดรูปแบบการเปลี่ยนแปลงภาพให้มีความเรียบร้อยด้วยตัวกรองเกาเซียนก่อนคำนวณหาขนาดและทิศทางของเกรเดียนต์ขึ้นมาและค่าของ Mask กำหนดค่าได้ดังสมการที่ 2.18 และ 2.19

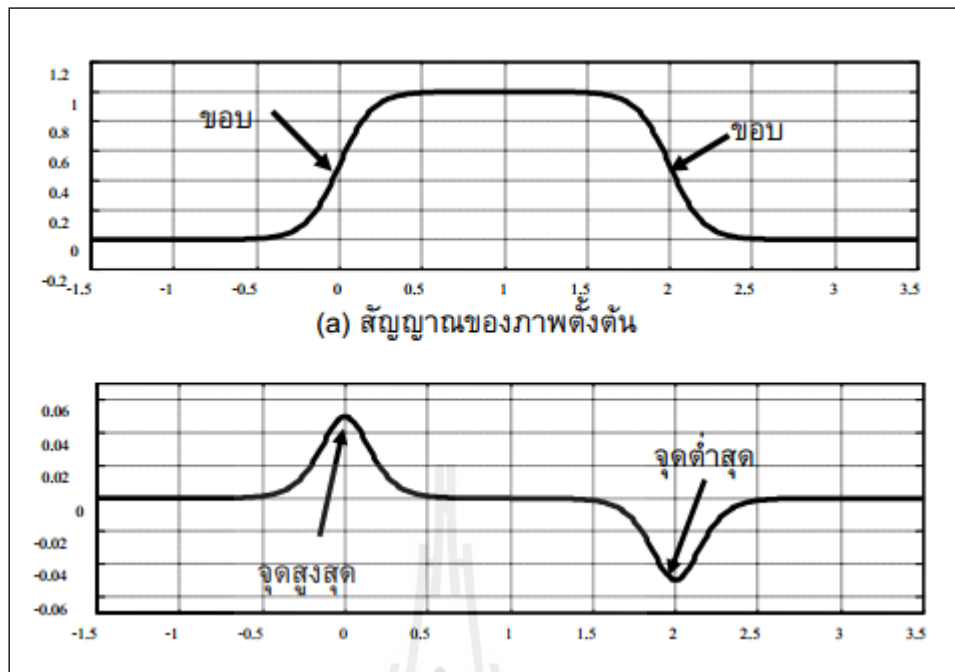
$$Mask(E_x) = \begin{bmatrix} Z_{x1} & Z_{x2} & Z_{x3} \\ Z_{x4} & Z_{x5} & Z_{x6} \\ Z_{x7} & Z_{x8} & Z_{x9} \end{bmatrix} = E_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (2.18)$$

$$Mask(E_y) = \begin{bmatrix} Z_{y1} & Z_{y2} & Z_{y3} \\ Z_{y4} & Z_{y5} & Z_{y6} \\ Z_{y7} & Z_{y8} & Z_{y9} \end{bmatrix} = E_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (2.19)$$

กำหนดให้ขนาดของ Mask เท่ากับ 3x3 และมีค่าเท่ากับ  $E_x$  และ  $E_y$  หากค่าอนุพันธ์อันดับหนึ่งของส่วน  $\frac{\partial P}{\partial x}$  และของส่วน  $\frac{\partial P}{\partial y}$  ขนาดและทิศทางของเกรเดียนต์หาได้จากสมการ 2.20 และ 2.21

$$M_{(i,j)} = \sqrt{E_x^2(i, j) + E_y^2(i, j)} \quad (2.20)$$

$$\theta(i, j) = \tan^{-1}(E_x^2(i, j) + E_y^2(i, j)) \quad (2.21)$$



รูปที่ 2.14 แสดงการหาขอบภาพโดยใช้อนุพันธ์อันดับหนึ่ง

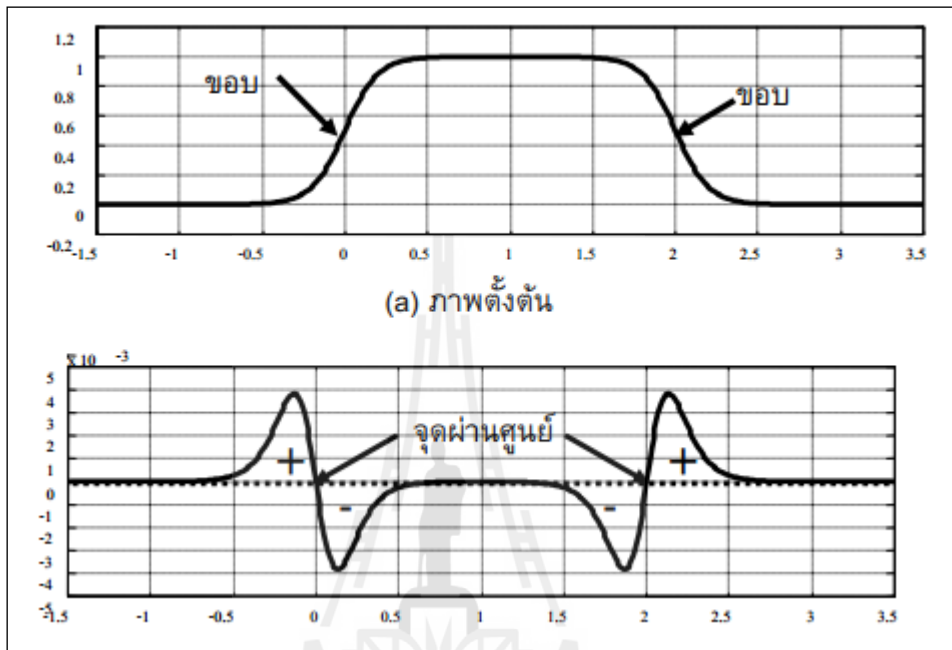
การกำหนดค่า Mask ของแนวแกน x และแกน y ทำให้ขนาดและทิศทางของเกรเดียนต์เปลี่ยนแปลงและลดขนาดของขอบในแต่ละพิกเซลของขอบภาพได้ พิกเซลรอบข้างของทิศทางของเกรเดียนต์ใดที่ไม่ใช่ค่าสูงสุดไม่ถือว่าเป็นขอบ ดังนั้นจะมีเพียงพิกเซลหนึ่งเดียวที่เป็นขอบ

#### 2.6.7 การค้นหาขอบภาพโดยใช้อนุพันธ์อันดับสอง

การค้นหาขอบโดยใช้อนุพันธ์อันดับสอง เปรียบได้กับการหาอนุพันธ์อันดับสองของภาพเพื่อให้ได้ขอบของภาพ โดยภาพที่ผ่านการอนุพันธ์อันดับสอง (Laplacian Operator ( $\nabla^2 P$ )) บริเวณที่เป็นส่วนขอบจะเด่นชัดขึ้น การประมาณค่าของ  $\nabla^2 P$  โดยใช้ Mask ของสมการที่ 2.18 และ 2.19 ทำให้ตำแหน่งของบริเวณขอบของวัตถุในภาพคือค่าจุดผ่านศูนย์ (Zero Crossing) ของ  $\nabla^2 P$  การหาอนุพันธ์อันดับสอง สามารถหาได้จากสมการที่ (2.22)

$$\nabla^2 P = \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \quad (2.22)$$

การค้นหาลักษณะขอบภาพด้วยอนุพันธ์อันดับสอง เมื่อทำ  $\nabla^2 P$  บริเวณขอบจะมีตำแหน่งเดียวกับค่าจุดผ่านศูนย์ของค่า  $\nabla^2 P$  พิจารณาได้จากตำแหน่งที่พิกเซลเปลี่ยนแปลงจากค่าที่เป็นบวกเป็นค่าที่เป็นลบหรือจากค่าที่เป็นลบไปเป็นค่าที่เป็นบวกดังภาพที่ 2.14

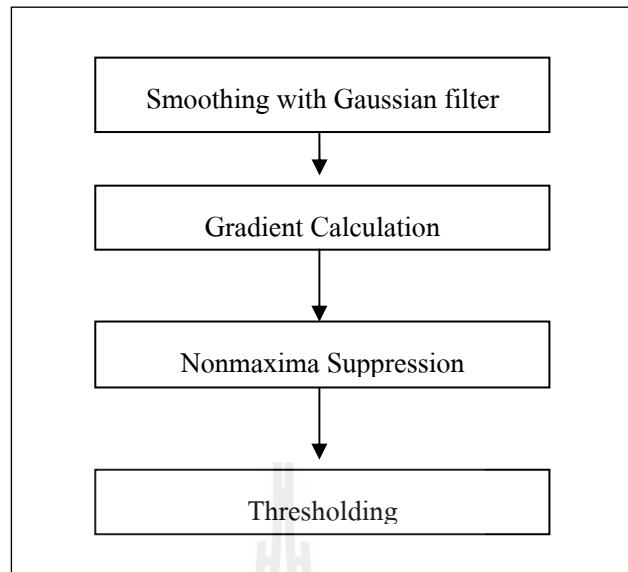


ภาพที่ 2.15 แสดงการหาขอบภาพโดยใช้อนุพันธ์อันดับสอง

การหาขอบโดยใช้อนุพันธ์อันดับสอง ไม่สนใจทิศทางของภาพในแนวแกน x และ y กำหนดจุดที่ค่า y เป็นจุดผ่านศูนย์ วิธีนี้เวลาในการคำนวณมากกว่าการค้นหาลักษณะขอบโดยใช้อนุพันธ์อันดับหนึ่งกล่าวคือเราสามารถตรวจจับความไม่ต่อเนื่องของพิกเซลในโดเมนรูปภาพได้โดยอนุพันธ์ของภาพนั่นเอง

#### 2.6.8 การค้นหาลักษณะขอบด้วย Canny

ขั้นตอนการค้นหาลักษณะขอบโดยวิธีของ Canny ประกอบด้วย 4 ขั้นตอน เริ่มต้นจากการปรับภาพให้เรียบ (Smoothing) ด้วยตัวกรองเกาส์เซียน เพื่อกำจัดสัญญาณรบกวน หลังจากนั้นหาอนุพันธ์อันดับหนึ่ง กำหนดค่าขนาดและทิศทางของเกรเดียนต์ นำค่าที่ได้มาคำนวณค่าของ Non-maxima Suppression กับค่าขนาดของเกรเดียนต์เพื่อทำให้ได้ขอบที่บางลงและในขั้นตอนสุดท้ายใช้การกำหนดจุดอ้างอิงสองระดับ (Double Thresholding) เพื่อระบุค่าของพิกเซลที่เป็นขอบและช่วยเชื่อมต่อบริเวณขอบโดยในแต่ละขั้นตอนมีรายละเอียดดังต่อไปนี้



รูปที่ 2.16 แสดงขั้นตอนการหาขอบภาพโดยวิธีของ Canny

#### 2.6.8.1 การปรับภาพให้เรียบ (Smoothing)

ขั้นตอนแรกการค้นหาขอบโดยอัลกอริทึมของ Canny ต้องกำจัดสัญญาณรบกวน (Noise) ออกก่อนด้วยวิธีการใช้ตัวกรองเกาส์เซียน กำหนดกรอบ (Masks) เป็นเมตริกซ์ขนาด  $3 \times 3$  หรือมีผลขนาดเท่ากับ 9 พิกเซล การกำหนดขนาดของตัวกรองเกาส์เซียน หากมีขนาดกว้างมาก จะมีผลทำให้ลดสัญญาณรบกวนได้มาก ถ้าขนาดกรอบกว้างมากเกินไปมีผลทำให้ขอบย่อย ๆ ที่เป็นส่วนรายละเอียดหายไปผลของภาพที่ผ่านการปรับภาพให้เรียบด้วยตัวกรองเกาส์เซียนหาได้จากสมการ 2.23

$$S_{(i,j)} = G_{(i,j,\sigma)} \cdot I_{(i,j)} \quad (2.23)$$

โดยที่  $I_{(i,j)}$  คือภาพที่ต้องการหาขอบ  
 $G_{(i,j,\sigma)}$  คือ Gaussian Smoothing Filter  
 $\sigma$  คือความคุมระดับของการ Smoothing

### 2.6.8.2 การคำนวณค่าของเกรเดียนต์ (Gradient Calculation)

ขั้นแรกปรับภาพ  $I_{(i,j)}$  ให้มีความเรียบผลลัพธ์ที่ได้คือค่าของภาพในฟังก์ชัน  $S_{(i,j)}$  ขั้นตอนที่สอง การหาค่าของเกรเดียนต์ในทิศทางของแกน x และแกน y และกำหนดขนาดของอนุพันธ์อันดับหนึ่งของ  $Px_{(i,j)}$  และ  $Qy_{(i,j)}$  ตามลำดับดังสมการที่ 2.24 และ 2.25

$$Px_{(i,j)} \approx (S_{(i,j+1)} + S_{(i,j)} + S_{(i+1,j+1)} - S_{(i+1,j)}) / 2 \quad (2.24)$$

$$Qy_{(i,j)} \approx (S_{(i,j)} - S_{(i+1,j)} + S_{(i,j+1)} - S_{(i+1,j+1)}) / 2 \quad (2.25)$$

นำค่า  $Px_{(i,j)}$  และ  $Qy_{(i,j)}$  ที่ผ่านการหาอนุพันธ์อันดับหนึ่งเมื่อคำนวณการแปลงรูปจากระนาบของระบบพิกัดฉาก (Rectangular Form) ไปเป็นระนาบพิกัดเชิงขั้ว (Polar Form) เพื่อหาขนาดและทิศทางของเกรเดียนต์แทนค่าตามสมการที่ 2.24 และ 2.25 ได้ค่าขนาดของเกรเดียนต์ดังนี้คือ  $M(i,j) = \sqrt{Px^2(i,j) + Qy^2(i,j)}$  และทิศทางของเกรเดียนต์ (Gradient Orientation) เท่ากับ  $\theta(i,j) = \tan^{-1}(Px(i,j) + Qy(i,j))$  และสามารถหาค่ามุม  $\theta$  ออกมาได้เมื่อแทนค่าตัวแปรในฟังก์ชัน  $\theta = \tan^{-1}(x, y)$

### 2.6.8.3 Non-maxima Suppression

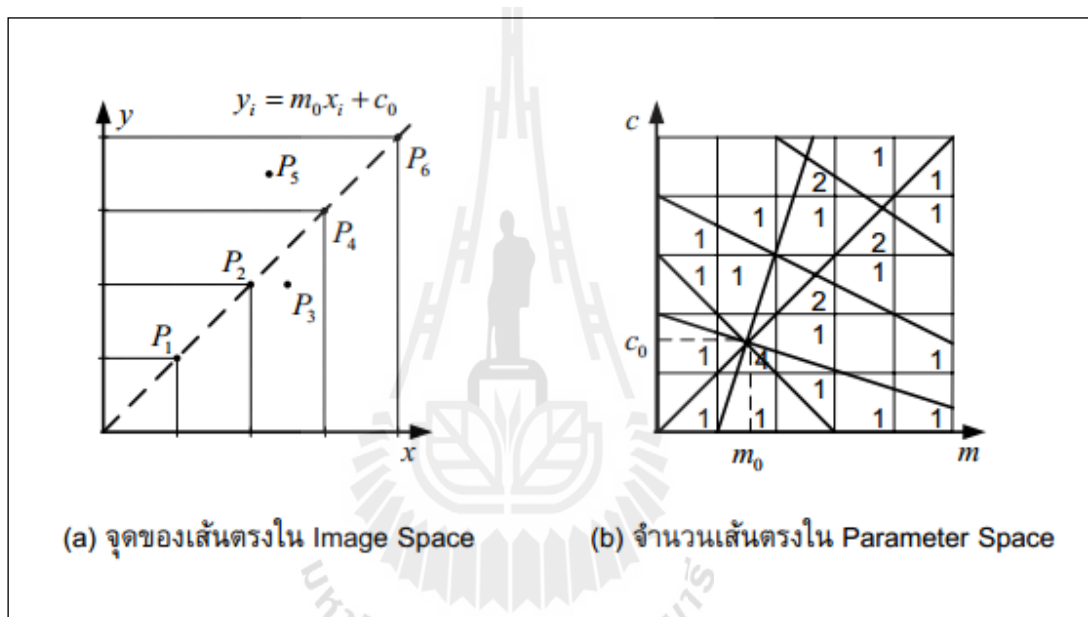
การค้นหาขอบภาพด้วยวิธีการของ Canny จุดที่ถือเป็นเส้นขอบของภาพได้นั้นต้องเป็นจุดที่ให้ค่าสูงสุดเฉพาะที่และเป็นทิศทางเดียวกันกับเกรเดียนต์ การค้นหาขอบภาพโดยใช้อนุพันธ์อันดับหนึ่งที่ได้ขอบที่บางเพียง 1 พิกเซล ภาพที่ได้หลังการทำ Non-maxima Suppression จะให้ค่าเป็นศูนย์ในทุกจุดที่เป็น local Maxima Point ซึ่งยังต้องคงค่าเดิมไว้

### 2.6.8.4 Thresholding

แม้ว่าภาพจะผ่านการ Smoothing ในขั้นตอนแรกแล้วก็ตาม ภาพที่ได้อาจยังมีเส้นขอบที่ไม่ใช่ขอบที่แท้จริงปรากฏอยู่ อันเนื่องจากสัญญาณรบกวนหรือลักษณะของวัตถุในภาพเป็นพื้นผิวที่มีลวดลายหรือมีรายละเอียดภายในมาก ดังนั้นเพื่อลดปัญหาดังกล่าวจึงได้มีการกำหนดค่า Threshold ขึ้นมา 2 ค่า คือ High Threshold ( $T_1$ ) และ Low Threshold ( $T_2$ ) โดยพิกเซลที่มีค่ามากกว่า  $T_1$  จะถูกปรับเป็น “1” เป็นพิกเซลที่เป็นขอบ แต่ถ้าค่าน้อยกว่า  $T_2$  จะถูกปรับเป็น “0” ส่วนค่าที่อยู่ระหว่างค่า Threshold ทั้งสองการปรับค่าเป็นค่า “0” หรือ “1” นั้นขึ้นอยู่กับพิกเซลที่อยู่รอบข้าง หากพบว่าพิกเซลที่อยู่รอบข้างของพิกเซลที่เป็นขอบ(ขอบค่า  $> T_1$ ) มีค่ามากกว่า  $T_2$  แล้วจะปรับค่าพิกเซลดังกล่าวให้มีค่าเป็น “1” และถือว่าเป็นสมาชิกหนึ่งในภาพขอบด้วยเช่นเดียวกัน ดังนั้นการทำ Threshold จะทำให้ภาพที่มีขอบหนาหรือบางนั่นเอง

## 2.7 การค้นหาเส้นตรงโดยใช้วิธีการของ Hough Transform

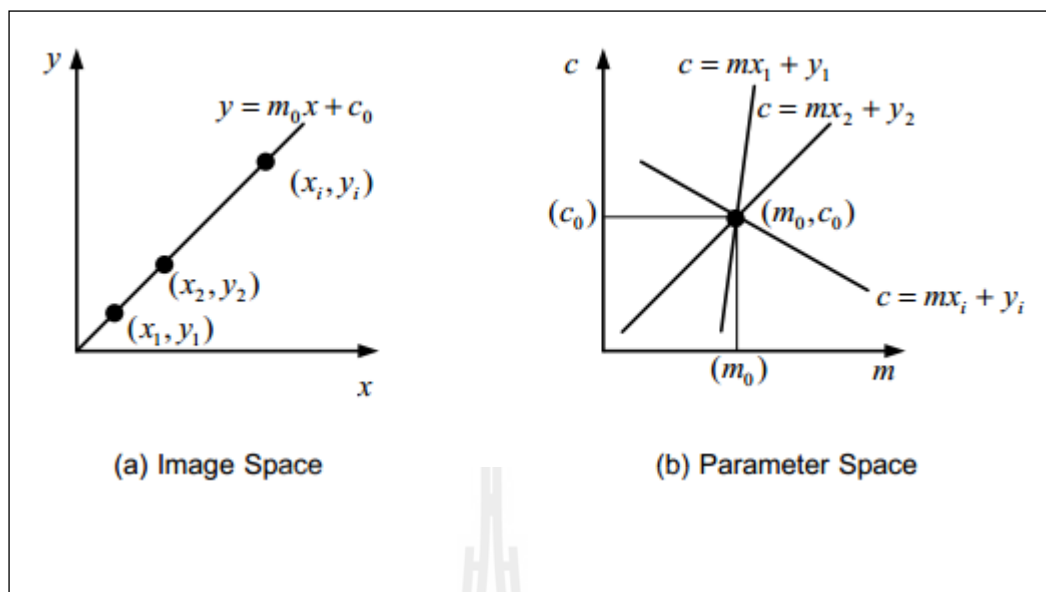
วิธีการของ Hough Transform คือการค้นหาเส้นตรงและวงกลมจากจุดต่าง ๆ โดยแต่ละจุดจะโหวตว่าจุดนั้น ๆ อยู่บนเส้นใดบ้าง เมื่อทุกจุดโหวตแล้ว สมการที่ถูกโหวตมากที่สุดจะเป็นเส้นที่ผ่านจุดดังกล่าว เมื่อพิจารณาเส้นตรง  $y = mx_0 + c$  เส้นตรงที่ผ่านจุด  $(x, y)$  มีค่าพารามิเตอร์คงที่  $(m, c)$  ซึ่งค่า  $c_0 = y - m_0x$  ดังนั้นจุดหนึ่งๆ จะโหวตให้กับสมการเส้นตรงที่มี Parameter ต่าง ๆ กันได้หลายสมการ เมื่อจุดทุกจุดได้โหวตไว้เรียบร้อยแล้ว สมการเส้นตรงที่ถูกโหวตมากที่สุดจะเป็นเส้นที่ผ่านจุดที่กำหนดมากที่สุดให้มากที่สุด



รูปที่ 2.17 แสดงการนับจำนวนเส้นตรงของการแปลง Hough Transform

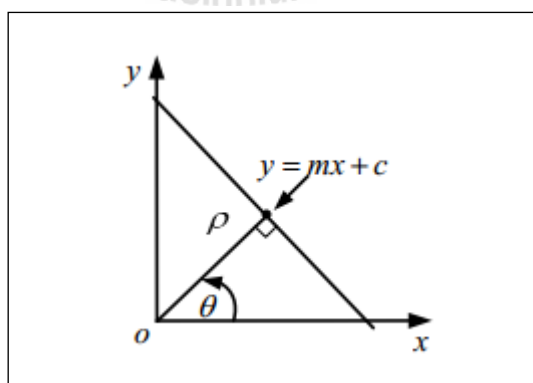
### 2.7.1 การแปลงรูปแบบจาก Image Space ไปสู่ Parameter Space

จากสมการของเส้นตรง  $y_i = mx_i + c$  เมื่อจุด  $(x_i, y_i)$  ดังแสดงในภาพที่ 2.17 (a) ค่าความชันและจุดแกน  $y$  ของสมการเป็นค่าคงที่  $(m_0, c_0)$  หรือเรียกว่าส่วนของ Parameter Space ดังนั้นการเปลี่ยนจาก Image Space ไปสู่ Parameter Space สมการของการเปลี่ยนแปลงคือที่จุด  $(m_0, c_0)$  จะเท่ากับ  $c_i = y - m_i x$  ดังแสดงในภาพที่ 2.17 (b)



รูปที่ 2.18 แสดงการแปลงรูปแบบระหว่าง Image Space กับ Parameter Space

ภาพที่ 2.18 (a) มีจำนวนของจุดทั้งสามจุดที่เส้นตรงที่มีค่าความชันและจุดตัดแกน  $y$  ที่ตำแหน่ง  $(m_0, c_0)$  ดังนั้นเมื่อพิจารณาในรูปแบบของ Parameter Space ที่จุด  $(m_0, c_0)$  ก็จะมีเส้นตรงที่เกิดขึ้นจากสมการเส้นตรงได้ทั้งหมดสามสมการที่ลากผ่านจุดดังกล่าว ดังนั้นการเมื่อกำหนดจุด  $(x, y)$  บนระนาบของ Image Space และทำการเปลี่ยนเป็นระนาบ Parameter Space หรือเรียกว่า Hough Space ดังแสดงในภาพที่ 2.19



รูปที่ 2.19 แสดงความสัมพันธ์ระหว่างค่าของเวกเตอร์  $\rho$  กับจุดของเส้นตรง



ความสัมพันธ์ระหว่างการเปลี่ยนแปลงรูปแบบคือที่จุด  $(\rho, \theta)$  บน Parameter Space จะเป็นจุดที่อยู่บนสมการเส้นตรงที่ลากผ่านจุด  $(x, y)$  เมื่อพิจารณาค่าของเวกเตอร์  $\rho$  ที่ตั้งฉากกับจุดดังกล่าวและทำมุมกับแกน  $x$  เท่ากับมุม  $\theta$  ดังนั้นการแปลงรูปจาก Image Space ไปสู่ Parameter Space สามารถหาค่าของเวกเตอร์  $\rho$  ได้จากสูตร

$$\rho = x \cos \theta + y \sin \theta \quad (2.26)$$

โดยที่  $\rho$  คือ ระยะที่วัดจากจุดกำเนิดไปตั้งฉากกับเส้นตรง

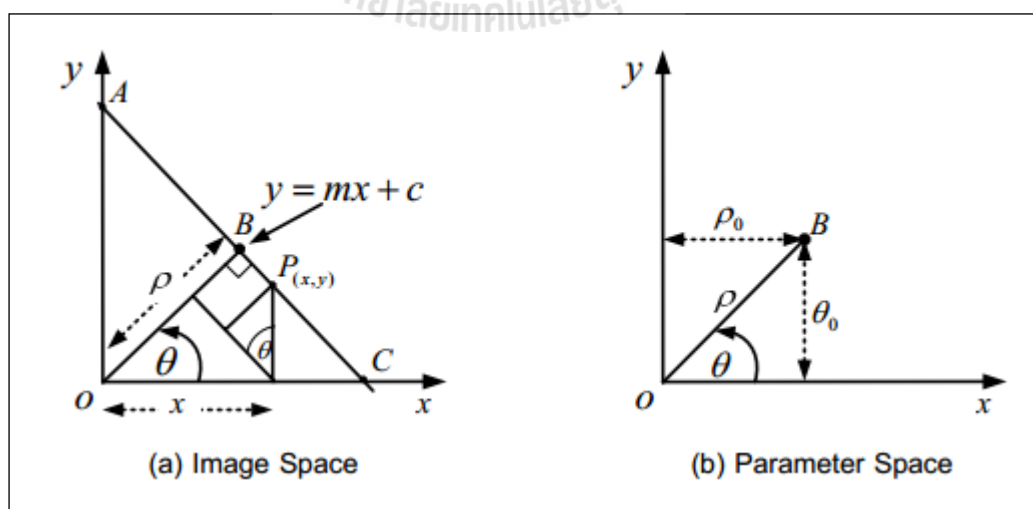
$\theta$  คือ ค่าของมุมระหว่างเวกเตอร์  $\rho$  กับแกน  $x$

### 2.7.2 การค้นหาเส้นตรงใช้ Hough Transform ในภาพสองมิติ

หลักการของ Hough Transform เพื่อการค้นหาองค์ประกอบของภาพสองมิติที่มีส่วนเส้นตรงปรากฏในภาพนั้น เป็นการหาเส้นตรงของภาพจากฟังก์ชันของสมการที่ 2.27

$$f(x, y, \rho_0, \theta_0) = x \cos \theta_0 + y \sin \theta_0 \quad (2.27)$$

และกำหนดให้จุดของภาพสองมิติมีค่าเท่ากับ  $(\rho_0, \theta_0)$  เมื่อ  $\rho_0$  ระยะที่วัดจากจุดกำเนิดไปตั้งฉากกับเส้นตรงและจุด  $\theta_0$  เป็นค่าของมุมระหว่างเวกเตอร์  $\rho$  กับแกน  $x$



รูปที่ 2.20 แสดงการค้นหาเส้นตรงของภาพสองมิติโดยใช้ Hough Transform

วิธีการ Hough Transform คือการหาค่า จากสมการที่ 2.26 โดยมีการกำหนดจุด  $(x_0, y_0)$  ของ Image Space ดังนั้นค่าของ  $\rho$  จะเท่ากับสมการ 2.28

$$\rho = \sqrt{x_0^2 + y_0^2} \left( \frac{x_0}{\sqrt{x_0^2 + y_0^2}} \cos \theta + \frac{y_0}{\sqrt{x_0^2 + y_0^2}} \sin \theta \right) \quad (2.28)$$

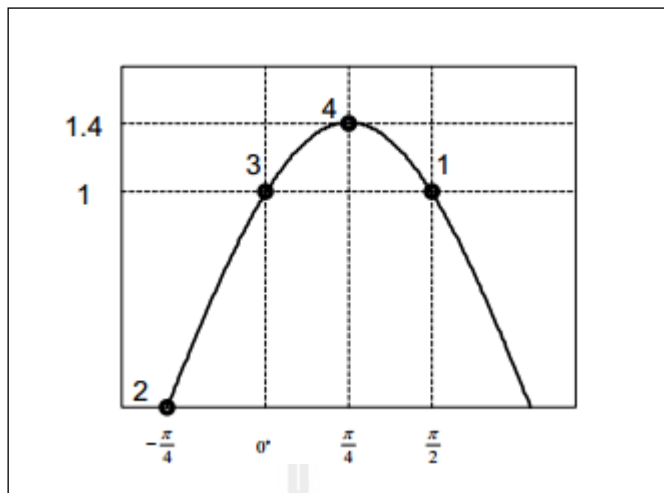
กำหนดให้  $r_0 \equiv \sqrt{x_0^2 + y_0^2}$  และ  $\alpha_0 \equiv \tan^{-1} \left( \frac{y_0}{x_0} \right)$  และเมื่อแทนค่าของสมการ 2.27 จะได้ค่าของ  $\rho$  ตามสมการ 2.29

$$\rho = r_0 (\cos \alpha \cos \theta + \sin \alpha_0 \sin \theta) = r_0 (\cos \alpha - \theta) \quad (2.29)$$

การแปลงรูปแบบของ HT พบว่า  $\rho = x \cos \theta + y \sin \theta$  ของจุด  $(x_0, y_0)$  ใน Image Space เป็นการแปลงรูปแบบไปสู่เส้นโค้งรูปไซน์ (Sinusoidal Curve) ใน Parameter Space ทิศทางทวนเข็มนาฬิกาและจุด  $(x_0, y_0)$  ของรูปเส้นโค้งรูปไซน์นี้ แสดงให้เห็นแทนเส้นตรงที่ลากผ่านจุด  $(x_0, y_0)$  ใน Image Space เมื่อทดลองแทนค่า  $\theta$  และ  $\rho$  ตามตารางที่ 2.1 ดังนั้นผลที่ได้เป็นการทำ Hough Transform ระหว่างค่าของเส้นตรงใน Image Space ไปสู่ค่าของจุดใน Hough Space ของดังแสดงในภาพที่ 2.20

ตารางที่ 2.1 ค่าความสัมพันธ์ระหว่างค่าของ  $\rho$  และ  $\theta$  เมื่อแทนค่าในสูตร 2.26

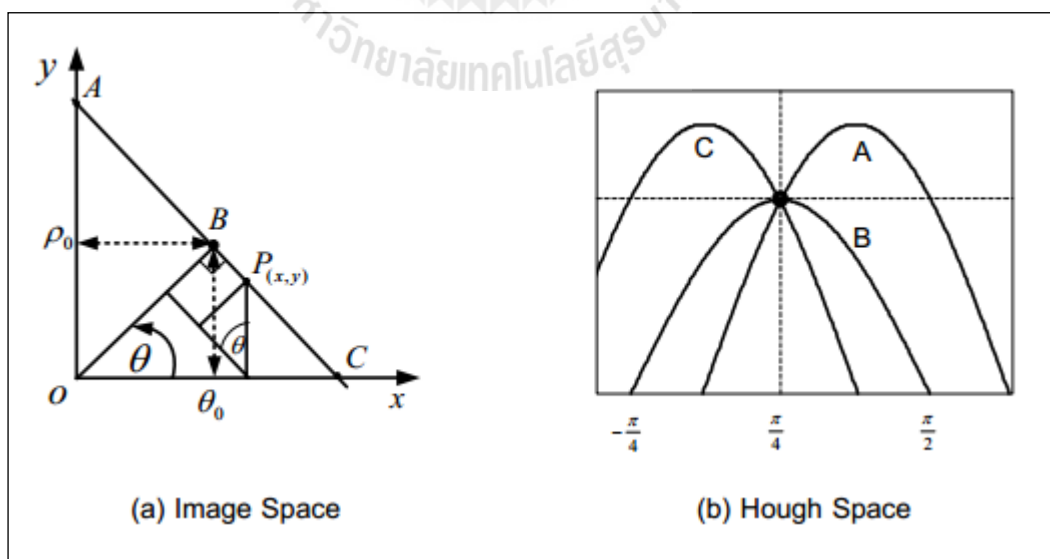
	จุดที่ 1	จุดที่ 2	จุดที่ 3	จุดที่ 4
$\theta$	$\pi/2 = 1.57$	$-\pi/4 = -0.785$	0	$\pi/4 = 0.785$
$\rho$	1	0	1	1.414



รูปที่ 2.21 แสดงตัวอย่างการทำ Hough Transform

เมื่อทั้ง 4 จุดมีที่เกิดขึ้นใน Hough Space ค่าของจุด 2 3 4 และ 1 ในภาพที่ 2.21(b) จะมีค่าตรงกันในแต่ละเส้นของในภาพที่ 2.21(a) Image Space ลักษณะของภาพแบบการแปลงบทกลับการแปลงรูปจาก Spatial Domain ไปสู่การแทนค่าของเส้นตรงและอธิบายได้ด้วยสมการ 2.30

$$\rho = x \cos \theta + y \sin \theta - \rho_0 = 0 \tag{2.30}$$



รูปที่ 2.22 แสดงการแปลงค่าของ Hough Space เมื่อมีการเปลี่ยนแปลงจุดในเส้นตรง

จากภาพที่ 2.21 ตัวอย่างของ 3 จุดของตำแหน่ง  $A(0,2)$ ,  $B(2,0)$  และ  $C(2,0)$  ของทุกจุดบนเส้นตรงจุดดังกล่าวนี้เป็นจุดที่เส้นตรงใน Hough Space ในความเป็นจริงแล้วจุดเป็นค่าหนึ่งของสมการเส้นตรงเส้นที่สองและสามมีค่าสอดคล้องกับเส้นโค้งที่รวมอยู่ในจุดของค่าเวกเตอร์  $\rho$  เท่ากับ 1.414 และค่าของมุม  $\theta$  เท่ากับ 0.79 เมื่อนำค่าของ  $(\rho, \theta)$  แทนค่าในสมการที่ 2.26 ได้ดังนี้

$$1.414 = x\cos(0.79)+y\sin(0.79) \quad (2.30)$$

โดยที่  $x+y = 2$

เมื่อเราทำ Hough Transform แล้วจุดที่  $P(x, y)$  ที่ปรากฏอยู่ในภาพจะมีเส้นตรงจำนวนมากมายที่ลากผ่าน ได้ดังนั้นวิธีการของ Hough Transform คือการนับค่าว่าจุด  $P(x, y)$  ดังกล่าวมีจำนวนเส้นของเส้นตรงลากผ่านจุดนี้จำนวนเท่าไร และถ้าพิจารณาใน Hough Space ก็จะมองเห็นเป็นจุดที่มีค่าของเส้นโค้งรูปซายด์ตัดผ่าน แสดงว่าจุดนั้นคือจุดเด่นของเส้นตรงที่ผ่านจุด  $P(x, y)$  มีค่ามากที่สุด จุดดังกล่าวมีค่าระยะห่างจากจุดกำเนิดมากที่สุดก็ต่อเมื่อเวกเตอร์  $\rho$  ตั้งฉากกับเส้นตรงที่ผ่านจุด  $P(x, y)$  ดังกล่าวนั่นเอง

## 2.8 การค้นหาเส้นโค้งหรือวงกลมโดยใช้วิธีการของ Hough Circle Transform

การสร้าง accumulator cell ในกรณีวงกลมจะเป็น  $A(i,j)$  3 มิติซึ่งพารามิเตอร์จะประกอบด้วย  $C_x$ ,  $C_y$  และ  $r$  สำหรับวิธีการคำนวณหาพิกัด  $x, y$  ที่อยู่บนวงกลมหรือส่วนโค้งอันเดียวกันจะใช้วิธีการเช่นเดียวกันกับกรณีของเส้นตรง ในทำนองเดียวกันการหาส่วนโค้งและวงกลมด้วย Hough Circle Transform จะใช้สมการ

$$(x - C_x)^2 + (y - C_y)^2 = r^2 \quad (2.31)$$

โดยที่  $C_x, C_y$  คือ จุดศูนย์กลางของวงกลมในแนวแกน  $x$  และ  $y$

$r$  คือ รัศมีของวงกลมหรือส่วนโค้ง

การสร้าง accumulator cell ในกรณีวงกลมจะเป็น  $A(i,j)$  3 มิติซึ่งพารามิเตอร์จะประกอบด้วย  $C_x, C_y$  และ  $r$  สำหรับวิธีการคำนวณหาพิกัด  $x, y$  ที่อยู่บนวงกลมหรือส่วนโค้งอันเดียวกันจะใช้วิธีการเช่นเดียวกันกับกรณีของเส้นตรง

## 2.9 โครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมเป็นการจำลองการทำงานของสมองมนุษย์ ด้วยโปรแกรมคอมพิวเตอร์ เป็นแนวความคิดที่ต้องการให้คอมพิวเตอร์มีความชาญฉลาดในการเรียนรู้เหมือนที่มนุษย์มีการเรียนรู้ สามารถฝึกฝนได้ และสามารถนำความรู้และทักษะไปแก้ปัญหาดifferent ๆ มีนักวิจัยจำนวนมากได้คิดค้นรูปแบบโครงข่ายประสาทเทียมแบบต่าง ๆ เพื่อนำมาประยุกต์ใช้อย่างกว้างขวาง การประยุกต์ใช้งานโครงข่ายประสาทเทียมมีตั้งแต่การใช้เพื่อตัดสินใจง่ายไปจนถึงงานที่มีความยุ่งยากซับซ้อน ตัวอย่างการประยุกต์ใช้งานบางส่วน ได้แก่ งานด้านการควบคุม งานด้านการควบคุม งานด้านการบิน ด้านยานยนต์ ด้านการบริหารจัดการ ด้านการธนาคาร ด้านการทหาร ด้านการบันเทิง และอื่น ๆ อีกมากมาย

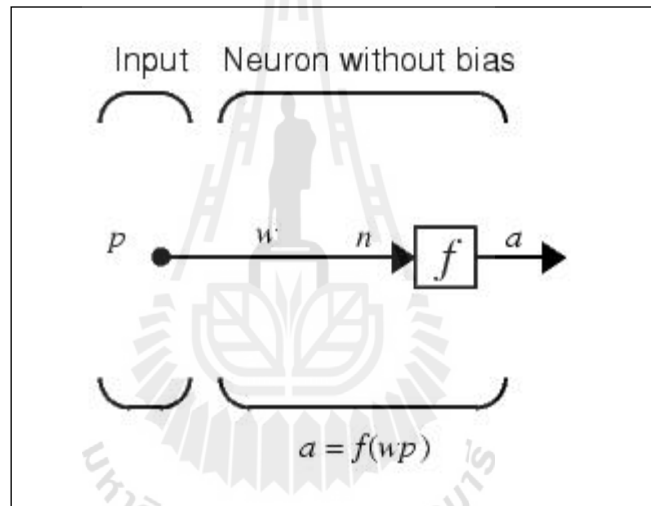
โครงข่ายประสาทเทียม (artificial neural network: ANN) เป็นการจำลองการทำงานของโครงข่ายประสาทของมนุษย์ (Biological Neurons) ซึ่งประกอบด้วยส่วนของการประมวลผลที่เรียกว่านิวรอน (Neuron) ทุก ๆ นิวรอนสามารถมีอินพุตได้หลายอินพุตแต่มีเอาต์พุตเพียงเอาต์พุตเดียว และทุก ๆ เอาต์พุตจะแยกไปยังอินพุตของนิวรอนอื่น ๆ ภายในโครงข่าย การติดต่อกันภายในระหว่างนิวรอนไม่ใช่ลักษณะการต่อแบบธรรมดาทุก ๆ อินพุตจะมีน้ำหนักเป็นตัวกำหนดกำลังของการติดต่อภายในและช่วยในการตัดสินใจ การทำงานของนิวรอนในบางโครงข่ายจะถูกกำหนดไว้ตายตัว แต่บางโครงข่ายสามารถที่จะปรับแต่งได้ซึ่งอาจจะเป็นการปรับแต่งจากภายนอกโครงข่ายหรือนิวรอนสามารถปรับได้ด้วยตัวของมันเอง ในจุดนี้แสดงถึงความสามารถในการเรียนรู้และจดจำของโครงข่ายประสาทเทียม

สมองประกอบด้วยประสาทจำนวนมหาศาล (ประมาณ  $10^{11}$ ) และมีจุดต่อจำนวนโครงข่ายประสาทประกอบขึ้นด้วยตัวส่วนสำคัญ 3 ส่วน คือ โยประสาท (nerve fiber หรือ dendrites) ตัวเซลล์ (cell body หรือ soma) และแกนประสาทนำออก (axon) ในแต่ละโครงข่ายประสาทจะเชื่อมต่อกันโดยจุดประสานประสาท (synapse) ซึ่งสามารถเปลี่ยนค่าความต้านทานได้ตามสัญญาณที่ส่งระหว่างกันของเซลล์ประสาท การส่งสัญญาณระหว่างเซลล์ประสาทได้โดยการถ่ายทอดสารประกอบไอออนและโพแทสเซียม

จากลักษณะและการทำงานของเซลล์ประสาทหรือนิวรอน ดังที่กล่าวมาข้างต้นได้ถูกนำมาสร้างทฤษฎีทางคณิตศาสตร์และจำลองการทำงานในรูปแบบพื้นฐานโดยใช้ชื่อว่าโครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมแบบง่ายจะมีค่าอินพุตเป็นสเกลาร์หนึ่งอินพุต โดยไม่มีค่าเอนเอียง หรือไบแอส (bias) โดยค่าอินพุตสเกลาร์  $p$  ถูกป้อนเข้าผ่านจุดต่อและคูณกับค่าความแข็งแรง (strength) ซึ่งเป็นค่าน้ำหนักสเกลาร์ (scalar weight:  $w$ ) และได้ผลคูณเป็นค่าสเกลาร์  $wp$  กลายเป็นค่าอินพุตที่ถูกจัดน้ำหนัก (weighted input:  $n$ ) ส่งต่อไปยังฟังก์ชันถ่ายโอน (transfer function:  $f$ ) ซึ่งเกิดเป็นค่าเอาต์พุตสเกลาร์ (scalar output:  $a$ ) ดังแสดงภาพที่ 2.22 ซึ่งสามารถคำนวณค่าเอาต์พุต  $a$  ได้จากสมการ

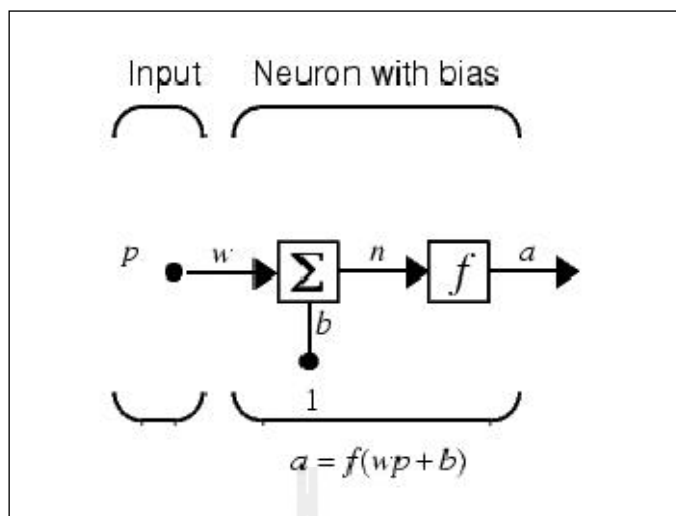
$$\begin{aligned} n &= wp \\ a &= f(n) = f(wp) \end{aligned} \tag{2.32}$$



รูปที่ 2.23 แสดงโครงข่ายประสาทเทียมหนึ่งหน่วย แบบไม่มีไบแอส

สำหรับภาพที่ 2.23 แสดงโครงข่ายประสาทเทียมแบบง่ายที่มีค่าอินพุตเป็นสเกลาร์หนึ่งอินพุต และมีค่าเอนเอียง  $b$  ซึ่งมีอินพุตเป็น 1 โดยค่าสเกลาร์เอาต์พุตสามารถคำนวณได้จากสมการ

$$\begin{aligned} n &= wp + b \\ a &= f(n) = f(wp + b) \end{aligned} \tag{2.33}$$



รูปที่ 2.24 แสดงโครงข่ายประสาทเทียมหนึ่งหน่วย แบบมีไบแอส

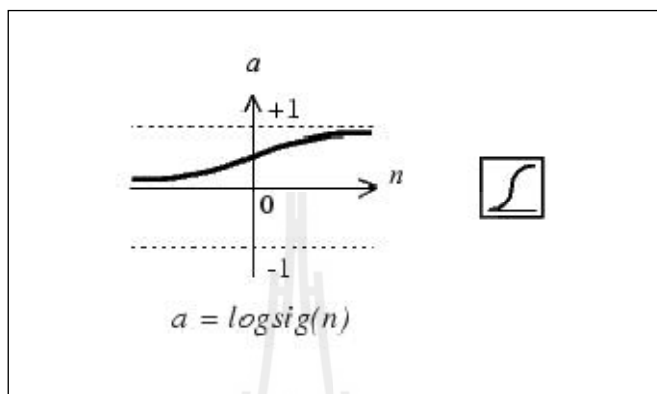
ในที่นี้  $f$  เป็นฟังก์ชันถ่ายโอน ตัวอย่างเช่น ฟังก์ชันซันบันได และฟังก์ชันซิกมอยด์ เป็นต้น ทำหน้าที่รับค่าอินพุต  $n$  ( $n = wp$  หรือ  $n = wp + b$ ) เพื่อเปลี่ยนเป็นค่าเอาต์พุต  $a$  ( $a = f(n)$ ) สำหรับค่าน้ำหนัก  $w$  และค่าไบแอส  $b$  เป็นค่าพารามิเตอร์ของโครงข่ายประสาทเทียมที่สามารถปรับได้จุดศูนย์กลางแนวคิดเกี่ยวกับโครงข่ายประสาทเทียมก็คือค่าพารามิเตอร์  $w$  และ  $b$  สามารถปรับเปลี่ยนได้เพื่อให้โครงข่ายประสาทเทียมแสดงพฤติกรรมตามที่เราต้องการ ยกตัวอย่าง เช่น การจดจำสิ่งที่เคยเรียนรู้มาก่อน ดังนั้นเราสามารถสอนโครงข่ายประสาทเทียมเพื่อให้ทำงานที่เราต้องการโดยการปรับแต่งค่าพารามิเตอร์น้ำหนัก และไบแอสของโครงข่ายประสาทเทียม หรือ บางทีโครงข่ายประสาทเทียมจะปรับพารามิเตอร์เพื่อให้ได้สิ่งคาดหวังด้วยตัวเองอย่างอัตโนมัติ

### 2.9.1 ฟังก์ชันการถ่ายโอน (Transfer Function)

ฟังก์ชันถ่ายโอนมีอยู่มากมายหลาย ๆ ชนิดด้วยกัน เช่น ฟังก์ชันถ่ายโอนแบบลอกซิกมอยด์ (Logarithmic sigmoid) ฟังก์ชันถ่ายโอนแบบเชิงเส้น (linear) ฟังก์ชันถ่ายโอนแบบจำกัดแฉ่ง (hardlimit) ฟังก์ชันถ่ายโอนแบบจำกัดแฉ่งสมมาตร (Symmetric hard-limit) ฟังก์ชันถ่ายโอนแบบ ไฮเพอร์บอลิกแทนเจนต์ซิกมอยด์ หรือแทนซิกมอยด์ (Hyperbolic tangent sigmoid) และอื่น ๆ สำหรับงานวิจัยนี้ใช้ฟังก์ชันการถ่ายโอนที่ใช้มี 2 แบบ คือ แบบลอกซิกมอยด์ และฟังก์ชันถ่ายโอนแบบเชิงเส้น

ฟังก์ชันถ่ายโอนแบบเชิงเส้นฟังก์ชันถ่ายโอนแบบลอจิสติกมอยด์

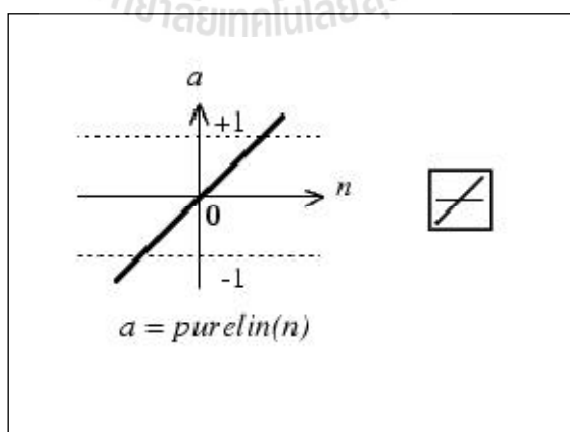
$$f(n) = \frac{1}{1 + \exp(-n)} \quad (2.34)$$



รูปที่ 2.25 แสดงฟังก์ชันการถ่ายโอนแบบลอจิสติกมอยด์ และสัญลักษณ์ที่ใช้

2.9.2 ฟังก์ชันถ่ายโอนแบบเชิงเส้น

$$f(n) = n \quad (2.35)$$



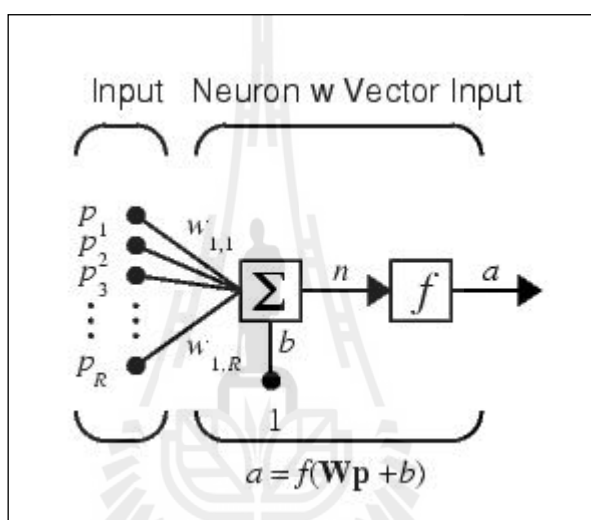
รูปที่ 2.26 แสดงฟังก์ชันการถ่ายโอนแบบเชิงเส้น และสัญลักษณ์ที่ใช้



## 2.10 โครงข่ายประสาทเทียมหนึ่งหน่วยแบบหลายอินพุต

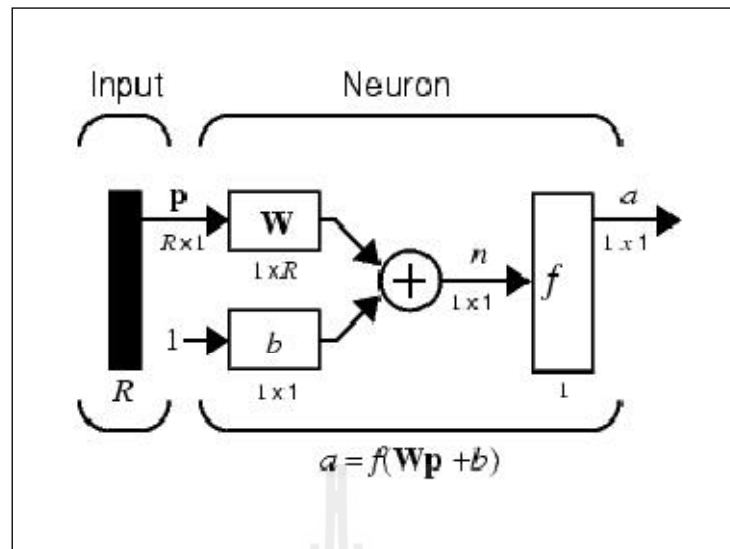
โครงข่ายประสาทเทียมแบบหลายอินพุตในรูปเวกเตอร์  $p = [p_1 \ p_2 \ \dots \ p_R]^T$  มีอินพุต  $R$  ค่า อินพุตแต่ละตัวถูกคูณด้วยเวกเตอร์น้ำหนัก  $W = [w_{11} \ w_{12} \ \dots \ w_{1R}]$  แล้วป้อนให้กับฟังก์ชันถ่ายโอน  $f$  เป็นเอาต์พุต  $a$  ดังสมการ

$$\begin{aligned} n &= wp + b = w_{11}p_1 + w_{12}p_2 + \dots + w_{1R}p_R + b \\ a &= f(n) = f(wp + b) \end{aligned} \quad (2.36)$$



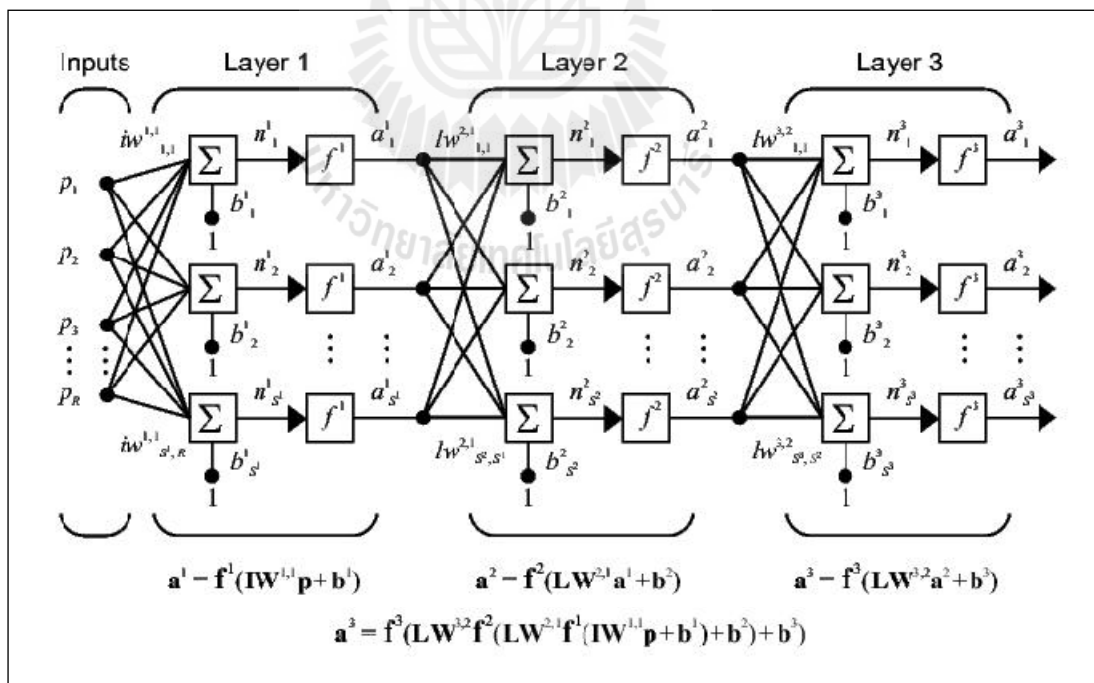
รูปที่ 2.27 แสดงโครงข่ายประสาทเทียมหนึ่งหน่วยแบบหลายอินพุต

เพื่อไม่ให้เปลืองพื้นที่มาก สามารถเขียนโครงข่ายประสาทเทียมหนึ่งหน่วยแบบหลายอินพุตในรูปย่อ โดยใช้สัญกรณ์เวกเตอร์ดังแสดงในภาพที่ 2.27



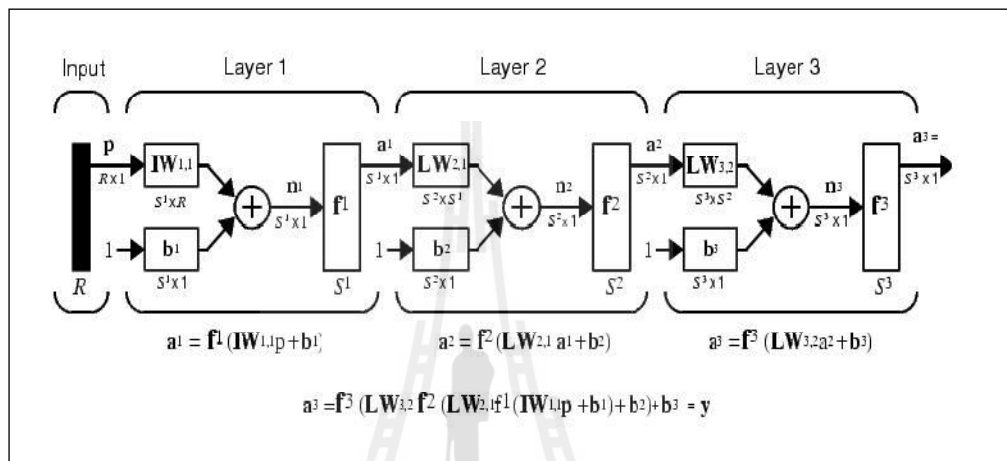
รูปที่ 2.28 แสดงโครงข่ายประสาทเทียมหนึ่งหน่วยแบบหลายอินพุตในรูปแบบย่อ

## 2.11 โครงข่ายประสาทเทียมแบบหลายชั้น (Multi Layered Perceptron)



รูปที่ 2.29 แสดงโครงข่ายประสาทเทียมสามชั้น

โครงข่ายประสาทเทียมแบบหลายชั้น (Multi-Layered Perceptron: MLP) เป็นโครงข่ายประสาทเทียมที่นิยมใช้มากที่สุด เป็นโครงข่ายที่สามารถทำงานที่มีความซับซ้อนมาก ๆ ได้ ซึ่งอาจกล่าวได้ว่า สามารถประยุกต์ใช้ได้กับงานเกือบทุกประเภท โดยมีข้อแม้ว่าต้องมีจำนวนชั้นและจำนวนนิวรอนที่เหมาะสม ภาพที่ 2.29 แสดงโครงข่ายประสาทเทียมแบบสามชั้น ภาพที่ 2.30 แสดงโครงข่ายประสาทเทียมแบบสามชั้นในรูปแบบเมทริกซ์



รูปที่ 2.30 แสดงโครงข่ายประสาทเทียมหลายอินพุตหลายชั้นในรูปแบบย่อ

โครงข่ายประสาทเทียมประกอบด้วยตัวประมวลผลเรียกว่านิวรอน ซึ่งเชื่อมต่อถึงกัน และนิวรอนแต่ละตัวจะส่งผ่านสัญญาณจากตัวมันไปยังนิวรอนตัวอื่น ๆ ต่างชั้น (layer) กัน ผ่านทางจุดเชื่อมต่อที่เรียกว่า Weight โดยที่ตัวนิวรอนที่อยู่ในเลเยอร์เดียวกันจะมีหน้าที่เหมือนกันซึ่งแต่ละนิวรอนจะรับค่าสัญญาณอินพุตที่เชื่อมต่อกับตัวมันทั้งหมดมาประมวลผล และจะให้สัญญาณออกมาที่เอาต์พุตเพียงหนึ่งค่าเท่านั้น โดยที่สัญญาณเอาต์พุตที่ได้จากแต่ละชั้นจะถูกส่งไปยังนิวรอนในชั้นปลายทางต่อไป (พยุง, 2548)

## 2.12 งานวิจัยที่เกี่ยวข้อง

เดิมทีในการวิเคราะห์ของเสียบนแผ่นเวเฟอร์ นั้นจะพิจารณาจำนวนของเสียที่เกิดขึ้นบนแผ่นเวเฟอร์ โดยแนวโน้มของของเสียจะมากขึ้นเมื่อขนาดของแผ่นเวเฟอร์มีขนาดใหญ่ขึ้น และเมื่อตัวชิปที่บรรจุอยู่ในแผ่นเวเฟอร์มีขนาดเล็กลง แต่อย่างไรก็ตามสำหรับกรณีที่จำนวนของเสียเท่ากัน การพิจารณาในขั้นต่อมาคือ การวิเคราะห์รูปแบบของเสีย (DeFect Patten) เนื่องจากจำนวนของเสียถึงแม้ว่าจะเท่ากันแต่รูปแบบของเสีย ที่เกิดขึ้นนั้นอาจแตกต่างกันได้ และส่งผลต่อลักษณะของการคาดการณ์ หรือการประเมินลักษณะอาการและสาเหตุในการเสีย ที่เกิดขึ้นบนแผ่นเวเฟอร์ที่แตกต่างกันออกไป โดยปัจจัยที่ส่งผลดังกล่าวนี้ คือ รูปแบบการกระจายตัวทางพื้นที่ของเสียที่เกิดขึ้นในบริเวณต่าง ๆ นั่นเอง ดังนั้นวิธีการสำหรับการวิเคราะห์รูปแบบของเสียต่าง ๆ ที่เกิดขึ้นบนแผ่นเวเฟอร์ที่จะได้อธิบายต่อไปนี้ จึงได้มุ่งเน้นที่การหารูปแบบของเสียที่เกิดขึ้นดังนี้

วิธีหาคัดชนีของการจับกลุ่มของของเสีย (Cluster Index: CI) เป็นวิธีสำหรับการประเมินการจับกลุ่มของเสีย โดยการสร้างแบบจำลองทางคณิตศาสตร์สำหรับคำนวณค่าของดัชนี ในการระบุถึงลักษณะของการจับกลุ่ม ซึ่งวิธีการนี้ได้มีการพัฒนาอย่างต่อเนื่อง ซึ่งเห็นได้จากการที่งานวิจัยหนึ่งที่ได้ทำการรวบรวมวิธีการจากงานวิจัยต่าง ๆ ที่ได้นำเสนอตามหลักการของการหาคัดชนีของการจับกลุ่ม และได้ทำการพัฒนาวิธีการใหม่ เพื่อแก้ปัญหาที่เกิดขึ้นจากวิธีการต่าง ๆ ที่ได้รวบรวมไว้ เพื่อให้ได้การหาคัดชนีการจับกลุ่มที่มีความยืดหยุ่น สามารถนำไปใช้พิจารณาในงานที่เกี่ยวข้องกับของเสียในรูปแบบต่าง ๆ ได้อย่างมีประสิทธิภาพและสามารถนำไปใช้ได้จริง

โดยวิธีการต่าง ๆ ที่งานวิจัยดังกล่าวนี้ ได้รวบรวมมาเริ่มจากวิธีการการกำหนดค่าของดัชนีอัลฟา (a Index) ที่ได้มาจากแบบจำลองไบโนเมียลเชิงลบ (Negative Binomial Mode) ซึ่งเป็นส่วนหนึ่งของแบบจำลองพัชอง (Poisson Model) โดยวิธีการนี้มีจุดอ่อนคือ ค่าของอัลฟา ที่ได้มาจากการคำนวณของเสีย นั้น มีลักษณะที่ไม่คงที่แน่นอน โดยเฉพาะอย่างยิ่งในกรณีที่มีการจับกลุ่มของเสียมีจำนวนน้อย ทำให้ได้ค่าของอัลฟาออกมาเป็นค่าลบ ดังนั้นจึงส่งผลต่อการประเมินลักษณะของรูปแบบของเสีย ซึ่งทำให้ได้ลักษณะที่ไม่แน่นอน ไม่คงที่ทั้ง ๆ ที่การกระจายตัวของของเสียนั้นเป็นแบบเดียวกัน

วิธีการต่อมาคือ การหาคัดชนีการจับกลุ่มโดยอาศัยการวิเคราะห์สถิติเชิงพื้นที่ [20] ซึ่งเป็นการใช้หลักการคำนวณบนพื้นที่สี่เหลี่ยม (Quadrant Methoe) ในการแบ่งพื้นที่ผิวของแผ่นเวเฟอร์ ออกเป็นหลาย ๆ ช่องที่มีขนาดเท่ากัน จากนั้นจึงทำการประเมินลักษณะการกระจายในแต่ละช่อง ดังนั้นประสิทธิภาพของดัชนีการจับกลุ่มด้วยวิธีการนี้ จึงขึ้นอยู่กับขนาดของช่อง และรูปร่างของช่องที่ให้มา ซึ่งหลักการที่นำมาใช้ในการกำหนดขนาดและจำนวนของช่องนั้น ขึ้นอยู่กับลักษณะของข้อมูลที่จะนำมาทำการทดสอบในแต่ละแบบ ทำให้การวัดการกระจายตัวของของเสียมี

ลักษณะเฉพาะ จึงส่งผลให้การประเมินการจับกลุ่มของของเสียที่ได้มานั้น ขาดความสอดคล้องกัน แม้ว่าในกรณีที่ลักษณะของการกระจายตัวของของเสียจะเหมือนกันก็ตาม

นอกจากนี้ยังมีการนำเสนอวิธีหาค่าดัชนีของการจับกลุ่ม ในลักษณะที่ไม่อาศัยการวิเคราะห์ข้อมูลทางสถิติ [21] ซึ่งเป็นการพัฒนาดัชนีการจับกลุ่ม โดยไม่อาศัยวิธีทางสถิติ แต่เป็นการประเมินระดับของการจับกลุ่มบนแผ่นเวเฟอร์ โดยการพิจารณาพิคคของตำแหน่งของเสียที่เกิดขึ้น และทำการจัดระดับของจำนวนของเสียที่อยู่ระหว่างจุดสองจุดที่พิจารณา ซึ่งค่าที่ได้จากวิธีการนี้บ่งชี้ว่า สำหรับการจับกลุ่มที่มีระดับสูงจะได้ค่าดัชนีมากกว่าหนึ่งขึ้นไป แต่อย่างไรก็ตามวิธีการนี้มีจุดอ่อนตรงที่ บางกรณีที่ลักษณะของการกระจายมีความแตกต่างกัน แต่ให้ค่าของดัชนีที่เท่ากัน ดังนั้นจึงส่งผลให้ไม่สามารถแยกแยะรูปแบบที่แท้จริงได้ ซึ่งสาเหตุของปัญหานี้มาจากการที่วิธีการนี้ทำการวิเคราะห์โดยไม่นำเอาวิธีการทางสถิติมาใช้ร่วมด้วย แต่เป็นการสร้างแบบจำลองทางคณิตศาสตร์ในแบบพื้นฐานที่มีความสามารถในการวิเคราะห์และคัดแยกรูปแบบการเสียได้อย่างรวดเร็ว แต่ไม่เหมาะสมสำหรับการนำไปใช้ในงานคัดแยกที่รูปแบบของเสียที่บนแผ่นเวเฟอร์นั้นมีความยุ่งยาก สลับซับซ้อนนั่นเอง

จากปัญหาที่เกิดขึ้นในวิธีที่ไม่ใช้การวิเคราะห์ทางสถิติ ดังนั้นจึงมีงานวิจัยที่ได้ออกแบบวิธีการที่อาศัยการหมุนเชิงหนึ่งมิติ (One Dimension Rotation) ซึ่งเป็นการคำนวณตามพิคคจากการฉาย (Projecting) ลงบนแกนใหม่โดยการหมุนแกน  $x$  ในทิศทางทวนเข็มนาฬิกา แทนการใช้เฉพาะการฉายบนแกน  $x$  เพียงอย่างเดียว ทำให้วิธีการนี้มีความสามารถเพิ่มจากที่มีอยู่ในวิธีที่ไม่ใช้การวิเคราะห์ทางสถิติ และสามารถแก้ปัญหาคัดแยกการจับกลุ่มที่แตกต่างกันได้แต่อย่างไรก็ตามวิธีการนี้ยังไม่สามารถทำการตรวจจับรูปแบบของเสียที่เกิดขึ้นที่บริเวณขอบได้ (Edge Defect Pattern)

จากข้อจำกัดในวิธีการข้างต้นทำให้ยังไม่สามารถนำไปใช้ในทางปฏิบัติได้ ดังนั้นจึงมีงานวิจัยที่ได้นำเสนอ วิธีการหาค่าดัชนีการจับกลุ่มที่อยู่อาศัยระยะห่างยูคลิดีเนียน (Euclidean Distance) [18] โดยเริ่มจากการคำนวณมุมระหว่างพิคคบนส่วนของวงกลมส่วนแรก จากนั้นจึงทำการคำนวณระยะห่างระหว่างมุมที่ได้อย่างเรียงลำดับไปแล้ว ซึ่งคำนวณระยะห่างยูคลิดีเนียนระหว่างพิคคของของเสียเทียบกับจุดเริ่มต้น และทำการจัดลำดับระยะห่างที่ได้และทำการคำนวณระยะห่างระหว่างตำแหน่ง ซึ่งค่าที่ได้จากการหาระยะนี้จะสามารถนำไปใช้ในการกำหนดดัชนีการจับกลุ่มของของเสียได้ และสามารถแยกแยะรูปแบบของเสียได้อย่างแม่นยำมากขึ้นกว่าวิธีการต่างๆ ที่ได้กล่าวมาแล้ว และยังคงข้อดีของงานวิจัยของการไม่ใช้การวิเคราะห์ทางสถิติ และการคำนวณตามพิคคการฉาย ไว้ได้อีกด้วย ดังนั้นในงานวิจัยนี้จึงได้นำเอาวิธีการนี้มาใช้ในส่วนของวิธีการหาค่าดัชนีการจับกลุ่มของเสีย ซึ่งเป็นขั้นตอนเพียงส่วนหนึ่งเท่านั้นของงานวิจัยนี้ โดยมีรายละเอียดของการดำเนินงานอธิบายไว้ในบทที่ 3

## บทที่ 3

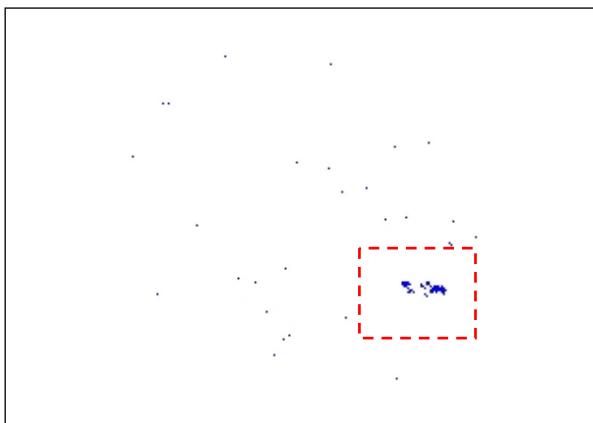
### วิธีการดำเนินการวิจัย

วัตถุประสงค์ในการศึกษาวิจัยในบทนี้ จะเป็นการวิเคราะห์ปัญหาที่อาจเกิดขึ้นในโรงงานผลิตฮาร์ดดิสก์ซึ่งมีกำลังการผลิตเป็นจำนวนมากปัญหาหนึ่งที่มีมักจะเกิดขึ้นในกระบวนการทดสอบฮาร์ดดิสก์ก็คือ ปัญหาของเสียที่เกิดขึ้นบนแผ่นบันทึกข้อมูล โดยบริเวณจุดที่ของเสียเกิดขึ้นนี้เราจะไม่ใช่ในการบันทึกข้อมูลเพราะว่าเป็นบริเวณที่มีปัญหาซึ่งทำให้ไม่สามารถทำการอ่านเขียนข้อมูลได้โดยสาเหตุที่ทำให้เกิดของเสียนี้อาจมาจากหลายสาเหตุลำพังของเสียที่มีเพียงจำนวนเล็กน้อยนั้นเราไม่สามารถที่จะนำไปวิเคราะห์อาการเสียที่เกิดขึ้นได้ว่ามาจากสาเหตุใดแต่ถ้าจำนวนของของเสียต่าง ๆ ที่เกิดขึ้นเหล่านั้นมีเป็นจำนวนมาก เราสามารถนำเทคนิคการประมวลผลภาพและการวิเคราะห์จากดัชนีการจับกลุ่มการจับกลุ่มของเสียที่เกิดขึ้นบนแผ่นบันทึกข้อมูลมาใช้ในการวิเคราะห์ถึงรูปแบบการกระจายตัวของจำนวนและตำแหน่งของเสียที่เกิดขึ้นซึ่งช่วยให้สามารถทำความเข้าใจกับสาเหตุของปัญหาและทำการแก้ไขได้อย่างรวดเร็วและมีความถูกต้องมากยิ่งขึ้น โดยในงานวิจัยฉบับนี้จะทำการวิเคราะห์อาการเสียของฮาร์ดดิสก์โดยนำข้อมูลที่ได้จากของของเสียต่าง ๆ ที่เกิดขึ้นบนแผ่นบันทึกข้อมูลมาทำการคัดแยกรูปแบบของความเสียหายโดยด้วยวิธีการที่กล่าวมาแล้วข้างต้นเพื่อช่วยหาสาเหตุของอาการเสียที่เกิดขึ้น ในกระบวนการผลิตได้

#### 3.1 การวิเคราะห์ปัญหารูปแบบของเสียที่เกิดขึ้นบนแผ่นบันทึกข้อมูลของฮาร์ดดิสก์

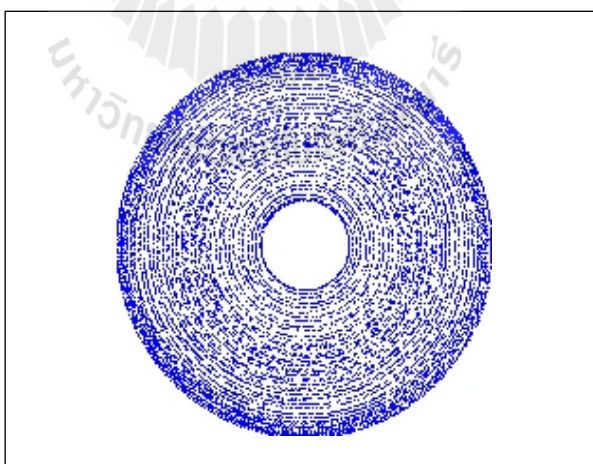
จากการศึกษารูปแบบของเสีย (Defect Pattern) ของที่เกิดขึ้นบนแผ่นบันทึกข้อมูลของฮาร์ดดิสก์ โดยพิจารณาจากการกระจายตัวของของเสียที่อยู่บนแผ่นบันทึกข้อมูล พบว่ามีรูปแบบการกระจายตัวของของเสียที่แบ่งได้เป็น กลุ่มใหญ่ ดังนี้

3.1.1 รูปแบบของเสียที่เกิดจากรอยยุบตัวของแผ่นบันทึกข้อมูล (Dent Pattern) รูปแบบของเสียที่เกิดขึ้นในรูปแบบนี้จะเกิดจากผิวของแผ่นบันทึกข้อมูลมีการยุบตัวลงอันเนื่องมาการผลิตแผ่นบันทึกข้อมูลเองหรือเกิดจากกระบวนการประกอบซึ่งอาจจะมีชิ้นส่วนของเครื่องจักรไปกระแทกหรือจิกให้เป็นร่องรอยลึกหรืออาจจะเกิดจากหัวอ่านไปกระแทกกับแผ่นบันทึกข้อมูลก็จะทำให้เกิดอาการในรูปแบบนี้ได้ทั้งสิ้น



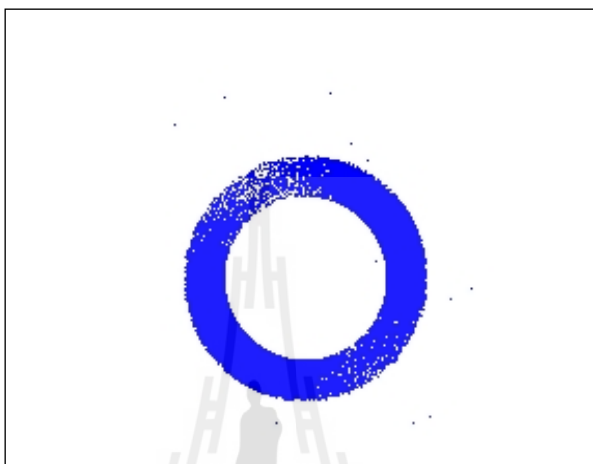
รูปที่ 3.1 แสดงรูปแบบของเสียที่เกิดจากรอยยุบตัวของแผ่นบันทึกข้อมูล (Dent Pattern)

3.1.2 รูปแบบของเสียจากการกระจายตัวแบบเต็มพื้นที่ (Full Pattern) การกระจายตัวของเสียแบบเต็มพื้นที่นี้ ส่วนใหญ่เป็นมีรูปแบบความเสียหายที่เกิดขึ้นจากการที่หัวอ่านไม่มีคุณภาพ ส่งผลให้การอ่านและเขียนข้อมูลมีปัญหา และเมื่อมีนำไปทดสอบการทำงาน ผลที่เกิดขึ้นจึงเสมือนว่าจะมีของเสียกระจายเกิดขึ้นโดยทั่วทั้งแผ่น ซึ่งแท้จริงแล้วปัญหาอยู่ที่หัวอ่านนั่นเอง ที่ทำการอ่านและเขียนข้อมูลโดยไม่มีประสิทธิภาพ



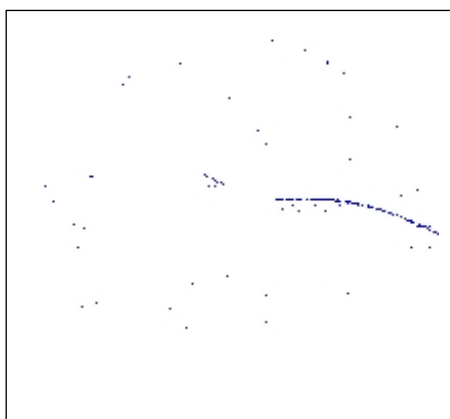
รูปที่ 3.2 แสดงรูปแบบของเสียจากการกระจายตัวแบบเต็มพื้นที่ (Full Pattern)

3.1.3 รูปแบบของเสียจากการกระจายตัวแบบจับกลุ่มเป็นวงแหวน (Circum Pattern) สาเหตุของการเสียในรูปแบบนี้ ส่วนใหญ่อาจเกิดจากการปรับแต่งค่าพารามิเตอร์ต่าง ๆ ของตัวฮาร์ดดิสก์ในบางตำแหน่งไม่เหมาะสม หรืออาจเกิดจากการที่หัวอ่านลงไปสัมผัสกับแผ่นบันทึกข้อมูลโดยตรง ส่งผลให้ของเสียการกระจายตัวเป็นรูปวงแหวน



รูปที่ 3.3 แสดงรูปแบบของเสียจากการกระจายตัวแบบจับกลุ่มเป็นวงแหวน

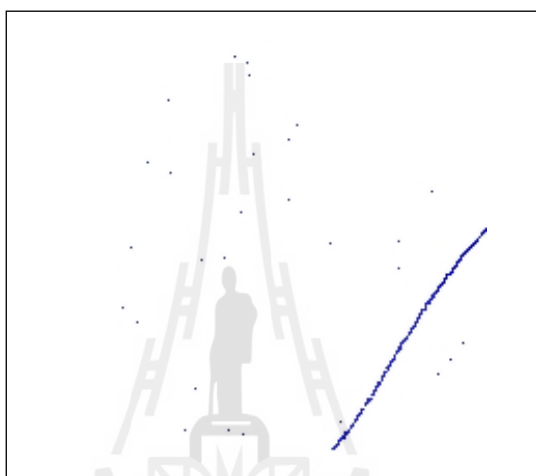
3.1.4 รูปแบบของเสียจากการกระจายตัวแบบจับกลุ่มแนวรัศมี (Radial Pattern) รูปแบบการเสียจากการกระจายตัวลักษณะนี้ อาจมาจากสาเหตุที่หัวอ่านไปสัมผัสกับแผ่นบันทึกข้อมูล ทำให้บริเวณที่ถูกสัมผัสเกิดความเสียหาย ส่งผลทำให้ไม่สามารถทำการบันทึกข้อมูลในบริเวณดังกล่าวได้



รูปที่ 3.4 แสดงรูปแบบของเสียจากการกระจายตัวแบบจับกลุ่มแนวรัศมี (Radial Pattern)



3.1.5 รูปแบบของเสียจากการกระจายตัวแบบจับกลุ่มเป็นเส้น (line Pattern) สาเหตุของการกระจายตัวของของเสียในลักษณะนี้อาจมาจากหลายสาเหตุ ส่วนใหญ่อาจเป็นผลมาจากมีการสัมผัสจุดที่เกิดขึ้นบนแผ่นบันทึกข้อมูลโดยมนุษย์หรืออุปกรณ์บางชนิด ยกตัวอย่างเช่น เครื่องจักร เครื่องมือต่าง ๆ และผู้ปฏิบัติงาน เป็นผลให้บริเวณดังกล่าวเกิดความเสียหายจนไม่สามารถที่จะนำไปใช้ในการเก็บข้อมูลได้ ดังนั้นรูปแบบของเสียที่เกิดขึ้นจึงมีรูปร่างลักษณะเป็นเส้นให้เห็นได้อย่างชัดเจน ซึ่งรูปแบบการกระจายตัวในลักษณะนี้สามารถบ่งชี้ถึงสาเหตุของการเสียได้โดยง่าย



รูปที่ 3.5 แสดงรูปแบบของเสียจากการกระจายตัวแบบจับกลุ่มเป็นเส้น (line Pattern)

### 3.2 การเตรียมข้อมูล

กระบวนการเตรียมข้อมูลก่อนจะทำคั่นหารูปแบบการกระจายตัวเริ่มจากการนำไฟล์ข้อมูลที่มีลักษณะเป็นเท็กไฟล์ (Text File) ซึ่งข้อมูลก่อนจะทำคั่นหารูปแบบการกระจายตัวเริ่มจากการนำไฟล์ข้อมูลที่มีลักษณะเป็นเท็กไฟล์ (Text File) ซึ่งข้อมูลภายในเท็กไฟล์นี้จะระบุตำแหน่งของเสียอยู่ในรูปของแตรคและเซ็กเตอร์เราจะทำการค้นหาข้อมูลตำแหน่งของเสียจากข้อมูลภายในเท็กไฟล์นี้และนำไปประมวลผล ในบางโอกาสอาจมีเหตุการณ์ที่ทำให้ข้อมูลภายในไฟล์นี้มีความไม่สมบูรณ์เราจะทำการตัดชุดข้อมูลที่ไม่สมบูรณ์นี้ออกไป และทำการวิเคราะห์เฉพาะชุดข้อมูลที่มีความสมบูรณ์เท่านั้น หลังจากนั้น เมื่อเราได้ชุดข้อมูลของแตรคและเซ็กเตอร์ มาแล้วเราจะต้องทำการแปลงชุดข้อมูลที่ได้มานี้ ให้อยู่ในรูปแบบของจุดแต่ละจุดที่อยู่บนแกน X และ Y เปรียบเสมือนของเสียที่เกิดขึ้นบนแผ่นบันทึกข้อมูลนั่นเอง และหลังจากนั้นเราจะนำข้อมูลแกน X และ Y นี้ไปใช้วิเคราะห์รูปแบบของการกระจายตัวในขั้นตอนต่อไป เพื่อหารูปแบบของเสียที่เกิดขึ้นบนแผ่นบันทึกข้อมูล โดยพิจารณาจากลักษณะการกระจายตัวของของเสียต่อไป

### 3.3 การคัดแยกรูปแบบ Dent pattern

งานวิจัยขอเสนอการสร้างกฎเพื่อใช้ในการคัดแยกของเสียในรูปแบบนี้โดยจะใช้วิธีการหาอัตราการเกิดของเสียซ้ำในตำแหน่งเดิมหรือการหาค่าความถี่สูงสุดในชุดตำแหน่งของเสียที่เกิดขึ้นว่าค่าสูงสุดของการเกิดของเสียซ้ำในตำแหน่งเดิมนั้นมีค่าเกินกว่าที่กำหนดไว้หรือไม่เพราะถ้าค่าการเกิดของเสียซ้ำในตำแหน่งเดิมมากกว่าที่กำหนดเราจะจัดว่ารูปแบบของของเสียเป็นแบบ Dent pattern โดยมีวิธีการคำนวณดังต่อไปนี้

ขั้นตอนที่ 1 คำนวณหามุมของของเสียแต่ละจุดโดยใช้สมการคำนวณดังต่อไปนี้

$$\theta_i = \tan^{-1}\left(\frac{Y_i}{X_i}\right) \quad (3.1)$$

กำหนดให้  $(X_i, Y_i)$  คือ ตำแหน่งของของเสียแต่ละจุดบนระนาบสองมิติและ  $i$  มีค่าตั้งแต่ 1, 2, 3, ..., n คือ ของเสียที่เกิดขึ้นทั้งหมด

ขั้นตอนที่ 2 คำนวณหาระยะทางของของเสียแต่ละจุดเทียบกับจุดศูนย์กลางโดยใช้สมการในการคำนวณดังต่อไปนี้

$$L_i = \sqrt{X_i^2 + Y_i^2} \quad (3.2)$$

เมื่อพิจารณา  $L_i$  คือ ระยะห่างของของเสียบนจุด  $(X_i, Y_i)$  เทียบกับจุดศูนย์กลางและ  $i$  มีค่าตั้งแต่ 1, 2, 3, ..., n คือของเสียที่เกิดขึ้นทั้งหมด

ขั้นตอนที่ 3 คำนวณหาค่าความถี่สูงสุดในชุดข้อมูลของเสีย

$$Repeat = F_{\max}(L_i, \theta_i) \quad (3.3)$$

โดยที่  $i$  มีค่าตั้งแต่ 1, 2, 3, ..., n จากสมการข้างต้นถ้าเรากำหนดค่า  $Repeat > Threshold$  ที่กำหนดไว้เราจะจัดของเสียกลุ่มนี้ว่าเป็นของเสียในรูปแบบ Dent pattern

### 3.4 การคัดแยกรูปแบบ Full pattern และแบบ Circum pattern ร่วมกับโครงข่ายประสาทเทียม

ตามงานวิจัยที่มีการสร้างกฎเพื่อใช้ในการคัดแยกของเสียที่เกิดขึ้นบนแผ่นเวเฟอร์ กระบวนการคัดแยกข้อมูลจะคัดแยกแบบการกระจายตัวทั่วทั้งแผ่นและการคัดแยกในรูปแบบนี้จะใช้วิธีการหาค่าดัชนีการจับกลุ่มกันของข้อมูลซึ่งจะเป็นการเปรียบเทียบค่าสัมประสิทธิ์ของความแปรปรวนเชิงมุมและค่าสัมประสิทธิ์ของความแปรปรวนเชิงระยะทางซึ่งในงานวิจัยชิ้นนี้ได้นำเอาโครงข่ายประสาทเทียมเข้ามามีส่วนร่วมในการคัดแยกรูปแบบโดยได้ทดลองเลือกโครงข่ายประสาทเทียมแบบ Feed Forward Neural Network (FFNN) โดยกำหนดให้มีอินพุต 2 อินพุต คือ ค่าสัมประสิทธิ์ของความแปรปรวนเชิงมุมและค่าสัมประสิทธิ์ของความแปรปรวนเชิงระยะทาง และจากการทดลองพบว่า การกำหนด Hidden layer เท่ากับ 20 จะให้ผลออกที่ดีที่สุด สำหรับการคำนวณหาค่าสัมประสิทธิ์ของความแปรปรวนเชิงมุมและค่าสัมประสิทธิ์ของความแปรปรวนเชิงระยะทางโดยการคำนวณหาค่าของเสียแต่ละจุดแสดงดังนี้

ขั้นตอนที่ 1 คำนวณหามุมของของเสียแต่ละจุดโดยใช้สมการคำนวณดังต่อไปนี้

$$\theta_i = \tan^{-1}\left(\frac{Y_i}{X_i}\right) \quad (3.4)$$

ขั้นตอนที่ 2 ทำการเรียงลำดับค่าของ  $\theta_i$  จากค่าน้อยไปหามากหลังจากนั้นคำนวณหาค่าแตกต่างของของมุมระหว่างจุดสองจุดที่อยู่ติดกันโดยสมการ

$$A_i = \theta_i - \theta_{(i-1)} \quad (3.5)$$

ขั้นตอนที่ 3 คำนวณหาระยะทางของของเสียแต่ละจุดเทียบกับจุดศูนย์กลางโดยใช้สมการในการคำนวณดังต่อไปนี้

$$L_i = \sqrt{X_i^2 + Y_i^2} \quad (3.6)$$

เมื่อพิจารณา  $L_i$  คือ ระยะห่างของของเสียบนจุด  $(X_i, Y_i)$  เทียบกับจุดศูนย์กลางและ  $i$  มีค่าตั้งแต่ 1, 2, 3, ..., n คือ ของเสียที่เกิดขึ้นทั้งหมด

ขั้นตอนที่ 4 ทำการเรียงลำดับค่าของ  $L_i$  จากค่าน้อยไปหามากหลังจากนั้นคำนวณค่าความแตกต่างของระยะทางระหว่างของเสียดสองจุดที่อยู่ติดกันโดยสมการ

$$D_i = L_i - L_{(i-1)} \quad (3.6)$$

กำหนดให้  $L_{(0)}$  มีค่าเท่ากับ 0

ขั้นตอนที่ 5 คำนวณหาค่าดัชนีการจับกลุ่มเชิงระยะทางโดยสมการ

$$\text{ค่าดัชนีการจับกลุ่มเชิงระยะทาง} = \max \left( \frac{S_D^2}{D} \right) \quad (3.7)$$

ขั้นตอนที่ 6 คำนวณหาค่าดัชนีการจับกลุ่มเชิงมุมโดยสมการ

$$\text{ค่าดัชนีการจับกลุ่มเชิงมุม} = \max \left( \frac{S_A^2}{A} \right) \quad (3.8)$$

โดยที่

$$\bar{A} = \frac{\sum A_i}{n}$$

$$S_A^2 = \frac{\sum (A_i - \bar{A})^2}{n-1}$$

$$\bar{D} = \frac{\sum D_i}{n}$$

$$S_D^2 = \frac{\sum (D_i - \bar{D})^2}{(n-1)}$$

ในงานวิจัยนี้ได้นำตัวอย่างรูปแบบของทั้งสองที่ผ่านการแปลงค่าดัชนีเชิงมุมและดัชนีเชิงระยะทางเพื่อปรับสอนโครงข่ายประสาทเทียมก่อนโดยจะแยกโครงข่ายประสาทเทียมเพื่อคัดแยกรูปแบบ Full pattern และโครงข่ายประสาทเทียมเพื่อคัดแยกแบบ Circum pattern ออกจากกันในการปรับสอนจึงต้องใช้ชุดข้อมูลที่ปรับสอนคนละชุดด้วยเช่นกัน

สำหรับวิธีการในการคัดแยกรูปแบบ Full pattern และแบบ Circum pattern ร่วมกับโครงข่ายประสาทเทียมที่ผ่านการปรับสอนมาเรียบร้อยแล้วเริ่มต้นจะรับอินพุตเข้ามาและตัดสินใจว่าเป็นรูปแบบ Full pattern หรือไม่โดยเอาที่พู่ทของระบบนี้จะมีเพียง 0 หรือ 1 เท่านั้น เป็นจริงระบบก็จะแสดงผลการคัดแยกออกมาเป็นแบบ Full pattern แต่ถ้าเป็นเท็จ หรือ 0 ระบบก็จะไปทำงานในขั้นต่อไปคือนำข้อมูลของของเสียที่ได้ไปคัดแยกในรูปแบบของ Circum pattern ต่อซึ่งถ้าเป็นจริงหรือเอาที่พู่ทเป็น 1 ระบบก็จะแสดงผลการคัดแยกออกมาว่าเป็น Circum pattern แต่ถ้าไม่ใช่ก็จะนำข้อมูลของของเสียไปประมวลผลในขั้นต่อไปเป็นต้น

### 3.5 การปรับสอนโครงข่ายประสาทเทียม

เพื่อให้โครงข่ายประสาทเทียมทำการเรียนรู้ข้อมูลของรูปภาพและซึ่งข้อมูลที่เอามาให้โครงข่ายประสาทเทียมได้เรียนรู้นั้นจะได้มาจากกระบวนการหา ค่าดัชนีการจับกลุ่มเชิงระยะทาง และ ค่าค่าดัชนีการจับกลุ่มเชิงมุม ซึ่งในงานวิจัยนี้จะใช้ทั้งสองค่านี้มาเป็นค่าอินพุตให้กับโครงข่ายประสาทเทียมซึ่งในงานวิจัยขั้นนี้จะต้องทำการสร้างโครงข่ายประสาทเทียมขึ้นมาสองโครงข่ายคือโครงข่ายประสาทเทียมเพื่อช่วยในการตัดสินใจสำหรับของเสียในรูปแบบเต็มพื้นที่และอีกอันคือโครงข่ายประสาทเทียมเพื่อช่วยในการตัดสินใจสำหรับของเสียในรูปแบบวงแหวนโดยในงานวิจัยนี้จะใช้โมดูลของภาษาไพธอนที่ชื่อ neurolab เพื่อสร้าง โครงข่ายประสาทเทียมขึ้นมา

3.5.1.1 การปรับสอน โคร่งข่ายประสาทเทียมเพื่อตัดสินใจของเสียในรูปแบบเต็มพื้นที่

ตารางที่ 3.1 ชุดข้อมูลเพื่อทำการสอนของเสียในรูปแบบเต็มพื้นที่

No.	ค่าดัชนีการจับกลุ่มเชิงระยะทาง	ค่าดัชนีการจับกลุ่มเชิงมุม	OUTPUT
1	0.982	2.219	1
2	1.026	0.029	1
3	0.999	0.264	1
4	1.035	0.707	1
5	1.666	0.015	1
6	1.505	0.015	1
7	7.62	0.984	0
8	1.899	0.497	0
9	1	0.013	0
10	1	0.007	0
11	1	0.023	0
12	9.832	0.054	0
13	5.743	84.211	0
14	1.342	14.023	0

ตารางที่ 3.1 ชุดข้อมูลเพื่อทำการสอนของเสียในรูปแบบเต็มพื้นที่ (ต่อ)

No.	ค่าดัชนีการจับกลุ่มเชิงระยะทาง	ค่าดัชนีการจับกลุ่มเชิงมุม	OUTPUT
15	3.368	19.927	0
16	1.152	24.344	0
17	1.23	24.268	0
18	1.265	8.009	0
19	3.612	19.006	0
20	8.505	19.884	0
21	4.906	21.49	0
22	1	2.745	0
23	1.964	16.922	0
24	1.053	13.541	0
25	2.438	14.259	0
26	2.092	9.925	0
27	1.769	12.623	0
28	4.332	14.61	0
29	4.397	42.702	0
30	3.28	8.193	0
31	1	41.847	0
32	10.322	28.89	0
33	4.447	15.808	0
34	1.051	6.453	0
35	2.011	14.305	0
36	2.611	18.769	0
37	3.069	21.216	0
38	3.576	16.62	0
39	10.065	37.398	0
40	1.081	34.62	0

### 3.5.1.2 สร้างชุดข้อมูลเพื่อทำการทดสอบโครงข่ายประสาทเทียมของเสียในรูปแบบเต็มพื้นที่

ตารางที่ 3.2 ชุดข้อมูลเพื่อทำการทดสอบโครงข่ายประสาทเทียมของเสียในรูปแบบเต็มพื้นที่

No.	ค่าดัชนีการจับกลุ่มเชิงระยะทาง	ค่าดัชนีการจับกลุ่มเชิงมุม
1	0.998	0.314
2	0.998	0.012
3	0.985	0.642
4	0.974	0.7
5	0.99	0.481
6	1.012	0.902
7	0.995	0.752
8	1.022	0.211
9	0.994	0.308
10	1.069	0.47
11	1.093	0.639
12	0.999	0.008
13	0.994	1.072
14	1.038	0.654

### 3.5.1.3 ทำการสร้างโครงข่ายประสาทเทียมขึ้นมาโดยใช้คำสั่ง

`nl.net.newff ([0, 100],[0,100],[4, 1])` คำสั่ง `nl.net.newff` คือคำสั่งการสร้างโครงข่ายประสาทเทียมแบบ FEED FORWARD NEURAL NETWORK ซึ่งจากคำสั่งจะเป็นการสร้างโครงข่ายประสาทเทียมที่มีอินพุต 2 อินพุตและมีเอาต์พุตเท่ากับ 1 เอาต์พุต โดยมีจำนวนชั้นซ่อนจำนวน 4 ชั้น ซึ่งจากการทดลองปรับเปลี่ยนค่าของจำนวนชั้นซ่อนพบว่าค่าจำนวนชั้นซ่อนที่เหมาะสมที่ทำให้เกิดค่าความผิดพลาดน้อยที่สุดสำหรับการสร้างโครงข่ายประสาทเทียมในการคัดแยกของเสียแบบเต็มพื้นที่คือจำนวนชั้นซ่อนเท่ากับ 4 ชั้นเป็นต้น



### 3.5.1.4 ทำการปรับสอนโดยใช้คำสั่ง

`net.train (inp, tar, epochs=1000, show=1000, goal=0.02)` โดยที่ `inp` ค่าของดัชนีการจับกลุ่มเชิงระยะทาง และ ค่าดัชนีการจับกลุ่มเชิงมุม ซึ่งทั้งสองค่านี้จะได้จากชุดข้อมูลตัวปรับสอนที่สร้างขึ้นมาและค่า `tar` คือ เป้าหมายของข้อมูลแต่ละชุดซึ่งจะแทนค่าด้วย 0 และ 1 โดยที่ 0 หมายถึงไม่ได้เป็นของเสียในรูปแบบเต็มพื้นที่และ 1 จะหมายถึงเป็นของเสียที่เป็นลักษณะเต็มพื้นที่ซึ่งข้อมูลนี้เราก็จะได้มาจากชุดข้อมูลตัวปรับสอนที่สร้างขึ้นมา Epochs คือค่าจำนวนรอบของการเรียนรู้ Goal ค่าผิดพลาดของการเรียนรู้

ตารางที่ 3.3 เปรียบเทียบค่าความผิดพลาด (MSE) โครงข่ายประสาทเทียมของเสียในรูปแบบเต็มพื้นที่

Hidden layer	Mean Square Error (MSE)
1	1.117981946
2	0.158315048
3	0.180840077
4	0.105053146
5	0.151594433
6	0.161353027
7	0.195780137
8	0.532359108
9	0.283507596
10	0.499443081

### 3.5.1.5 ทำการเก็บค่าของโครงข่ายประสาทเทียมโดยใช้คำสั่ง

`net.save ('check_full.net')` จะหมายถึงการเก็บค่าของโครงข่ายประสาทเทียมที่ได้ทำการปรับสอนมาเรียบร้อยแล้วเก็บไว้เป็นไฟล์เพื่อสะดวกในการเรียกมาใช้งานในอันดับต่อไป

### 3.5.2 การปรับสอนโครงข่ายประสาทเทียมเพื่อตัดคลื่นใจของเสียในรูปแบบวงแหวน

#### 3.5.2.1 สร้างชุดข้อมูลเพื่อทำการสอนในรูปแบบวงแหวน

ตารางที่ 3.4 ชุดข้อมูลเพื่อทำการสอนในรูปแบบวงแหวน

No.	ค่าดัชนีการจับกลุ่มเชิงระยะทาง	ค่าค่าดัชนีการจับกลุ่มเชิงมุม	OUTPUT
1	0.982	2.219	0
2	0.999	0.264	0
3	1	0.013	1
4	1	0.007	1
5	1	0.023	1
6	1	2.745	0
7	1	41.847	0
8	1.026	0.029	0
9	1.035	0.707	0
10	1.051	6.453	0



ตารางที่ 3.4 ชุดข้อมูลเพื่อทำการสอนในรูปแบบวงแหวน (ต่อ)

No.	ค่าดัชนีการจับกลุ่มเชิงระยะทาง	ค่าค่าดัชนีการจับกลุ่มเชิงมุม	OUTPUT
11	1.053	13.541	0
12	1.081	34.62	0
13	1.152	24.344	0
14	1.23	24.268	0
15	1.231	15.628	0
16	1.265	8.009	0
17	1.342	14.023	0
18	1.505	0.015	0
19	1.666	0.015	0
20	1.769	12.623	0
21	1.899	0.497	1
22	1.964	16.922	0
23	2.011	14.305	0
24	2.092	9.925	0
25	2.438	14.259	0
26	2.611	18.769	0
27	3.069	21.216	0
28	3.28	8.193	0
29	3.368	19.927	0
30	3.576	16.62	0
31	3.612	19.006	0
32	3.85	13.771	0
33	4.332	14.61	0
34	4.397	42.702	0

ตารางที่ 3.4 ชุดข้อมูลเพื่อทำการสอนในรูปแบบวงแหวน (ต่อ)

No.	ค่าดัชนีการจับกลุ่มเชิงระยะทาง	ค่าค่าดัชนีการจับกลุ่มเชิงมุม	OUTPUT
35	4.447	15.808	0
36	4.906	21.49	0
37	5.743	84.211	0
38	7.62	0.984	1
39	8.505	19.884	0
40	9.832	0.054	1

3.5.2.2 สร้างชุดข้อมูลเพื่อทำการทดสอบโครงข่ายประสาทเทียมของเสียในรูปแบบวงแหวน

ตารางที่ 3.5 ชุดข้อมูลเพื่อทำการทดสอบโครงข่ายประสาทเทียมของเสียในรูปแบบวงแหวน

No.	ค่าดัชนีการจับกลุ่มเชิงระยะทาง	ค่าค่าดัชนีการจับกลุ่มเชิงมุม
1	1	0.019
2	6.499	0.386
3	3.05	0.033
4	3.827	0.425
5	4.199	0.344
6	2.528	0.215
7	2.578	0.106
8	2.256	0.344
9	4.187	0.029
10	1	0.41
11	1	0.333
12	1	0.18

### 3.5.2.3 ทำการสร้างโครงข่ายประสาทเทียมขึ้นมาโดยใช้คำสั่ง

`nl.net.newff` ([[0, 100],[0,100]],[3, 1]) คำสั่ง `nl.net.newff` คือคำสั่งการสร้างโครงข่ายประสาทเทียมแบบ FEED FORWARD NEURAL NETWORK ซึ่งจากคำสั่งจะเป็นการสร้างโครงข่ายประสาทเทียมที่มีอินพุต 2 อินพุตและมีเอาต์พุตเท่ากับ 1 เอาต์พุต โดยมีจำนวนชั้นซ่อนจำนวน 3 ชั้น ซึ่งจากการทดลองปรับเปลี่ยนค่าของจำนวนชั้นซ่อนพบว่าค่าจำนวนชั้นซ่อนที่เหมาะสมที่ทำให้เกิดค่าความผิดพลาดน้อยที่สุดสำหรับการสร้างโครงข่ายประสาทเทียมในการคัดแยกของเสียแบบวงแหวนคือจำนวนชั้นซ่อนเท่ากับ 3 ชั้นเป็นต้น

### 3.5.2.4 ทำการปรับสอนโดยใช้คำสั่ง

`net.train` (inp, tar, epochs=1000, show=1000, goal=0.02) โดยที่ inp ค่าของดัชนีการจับกลุ่มเชิงระยะทาง และ ค่าดัชนีการจับกลุ่มเชิงมุม ซึ่งทั้งสองค่านี้จะได้มาจากชุดข้อมูลตัวปรับสอนที่สร้างขึ้นมาและค่า tar คือ เป้าหมายของข้อมูลแต่ละชุดซึ่งจะแทนค่าด้วย 0 และ 1 โดยที่ 0 หมายถึงไม่ได้เป็นของเสียในรูปแบบเต็มพื้นที่และ 1 จะหมายถึงเป็นของเสียที่เป็นลักษณะวงแหวนซึ่งข้อมูลนี้เราก็จะได้มาจากชุดข้อมูลตัวปรับสอนที่สร้างขึ้นมา

Epochs คือค่าจำนวนรอบของการเรียนรู้

Goal ค่าผิดพลาดของการเรียนรู้

ตารางที่ 3.6 เปรียบเทียบค่าความผิดพลาด (MSE) โครงข่ายประสาทเทียมของเสียในรูปแบบวงแหวน

Hidden layer	mean square error (MSE)
1	1.83103962
2	1.20963587
3	0.398692976
4	0.424955094
5	0.421209851
6	0.406345331
7	0.433620059
8	0.43870555
9	0.446553005
10	0.43664755

### 3.5.2.5 ทำการเก็บค่าของโครงข่ายประสาทเทียมโดยใช้คำสั่ง

`net.save ('check_circum.net')` จะหมายถึงการเก็บค่าของโครงข่ายประสาทเทียมที่ได้ทำการปรับสอนมาเรียบร้อยแล้วเก็บไว้เป็นไฟล์เพื่อสะดวกในการเรียกมาใช้งานในอันดับต่อไป

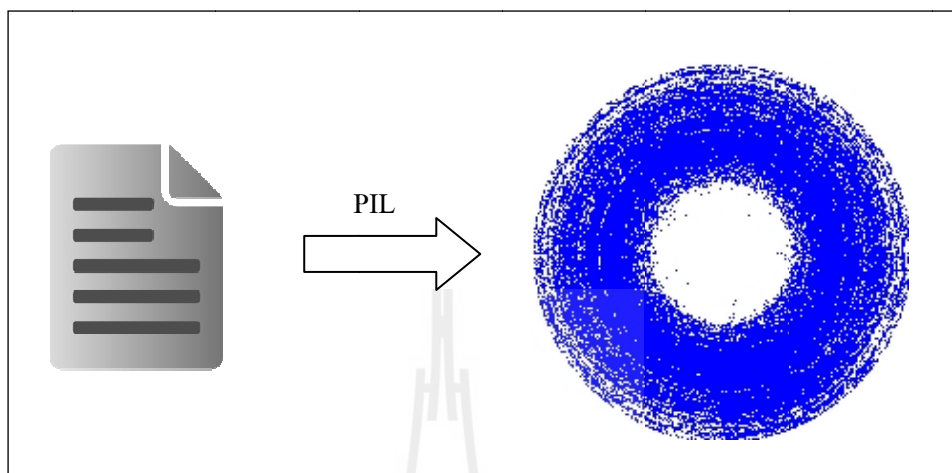
## 3.6 การคัดแยกรูปแบบ Radial line และแบบ Straight line

ในกระบวนการคัดแยกที่กล่าวมาข้างต้นจะเป็นการนำค่าข้อมูลของของเสียที่อยู่ในรูปของ Text file มาทำการแปลงให้อยู่ในรูปพิกัดเพื่อนำมาคำนวณสำหรับงานวิจัยชิ้นนี้ได้เสนอเทคนิคการประมวลผลภาพเพื่อทำการวิเคราะห์และคัดแยกรูปแบบของเสียที่เป็นลักษณะ Radial line และแบบ Straight line โดยเทคนิคการประมวลผลภาพที่งานวิจัยนี้เลือกใช้คือ Hough Transform method ซึ่งเป็น methods ที่ใช้สำหรับการ detect เส้นหรือขอบในภาพได้เป็นอย่างดี

กระบวนการประมวลผลภาพจะเริ่มจากการนำข้อมูล text file มาทำการแปลงให้เป็นรูปก่อนเป็นอันดับแรกจากนั้นทำการ pre-processing เพื่อหาขอบของภาพและปรับปรุงคุณภาพของภาพให้ดีขึ้นก่อนที่จะไปเข้ากระบวนการของ Hough Transform ต่อไป

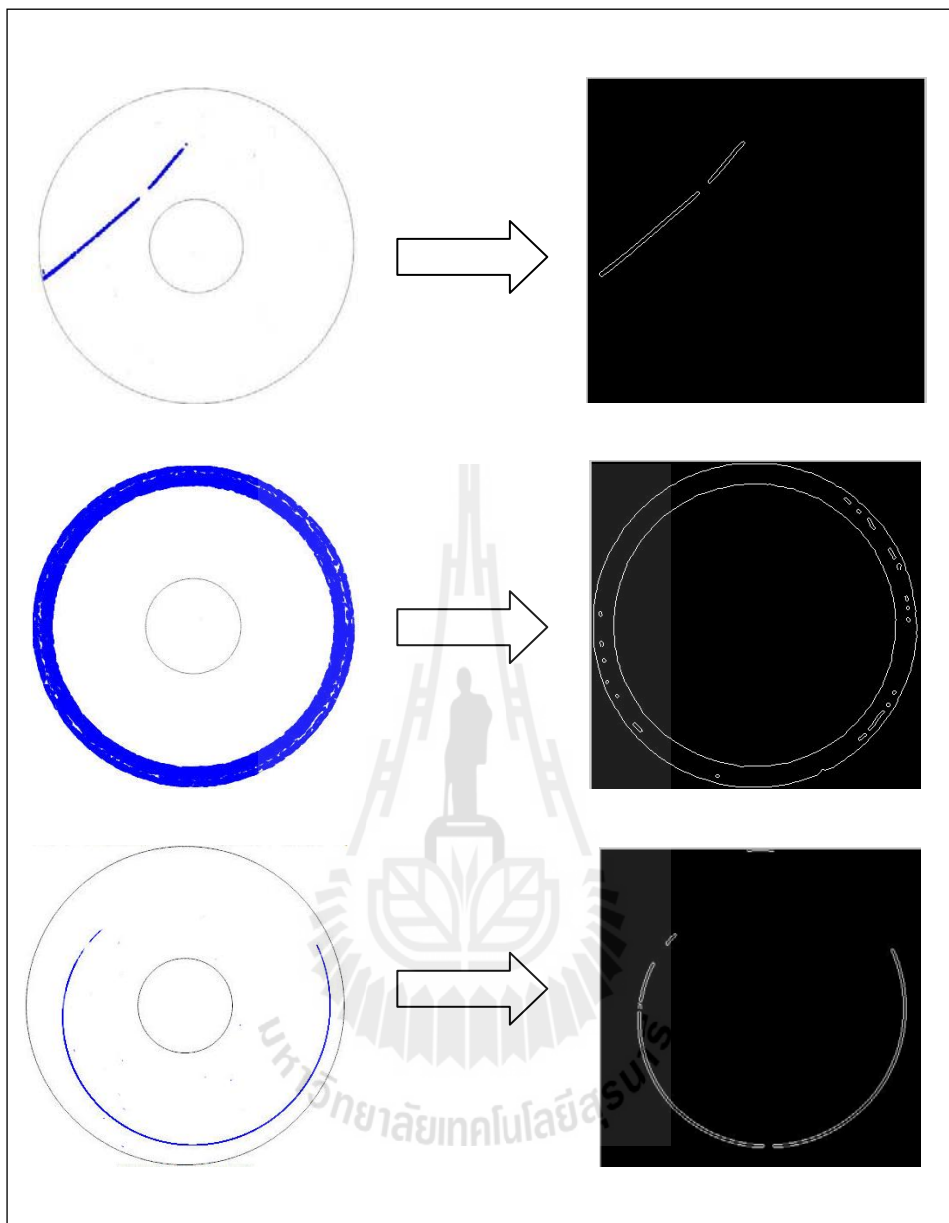
วิธีการของ Hough Transform คือการค้นหาเส้นตรงและวงกลมจากจุดต่างๆโดยแต่ละจุดจะโหวตว่าจุดนั้น ๆ อยู่บนเส้นใดบ้าง เมื่อทุกจุดโหวตแล้ว สมการที่ถูกโหวตมากที่สุดจะเป็นเส้นที่ผ่านจุดดังกล่าว เมื่อพิจารณาเส้นตรงโดยจะทำการเปลี่ยนรูปภาพจากของเดิมที่เป็น Image Space ไปสู่ Parameter Space แต่ก่อนที่แปลงรูปภาพจาก Image Space ไปเป็น Parameter Space ใต้นั้นจะต้องมีผ่านกระบวนการต่างๆเช่น การสร้างรูปภาพจาก Text File จากนั้นทำการ Pre-Processing รูปภาพแล้วจึงจะเข้าสู่กระบวนการประมวลผลภาพต่อไป ซึ่งในการบวนการที่กล่าวมานั้นงานวิจัยชิ้นนี้ได้ใช้แบ่งการทำงานออกเป็นสองส่วนคือส่วนของการสร้างรูปภาพจาก Text File ซึ่งในส่วนนั้นนั้นจะใช้โมดูลของภาษาไพธอนชื่อว่า Python Imaging Library (PIL) ในการสร้างรูปภาพจาก Text File เมื่อได้รูปภาพมาแล้วก็จะเอามาทำในส่วนต่อไปคือส่วนของการประมวลผลภาพไม่ว่าจะเป็น การหาขอบภาพหรือการกำจัดสัญญาณรบกวนในงานวิจัยนี้จะใช้โมดูลที่ชื่อว่า Open CV ซึ่งเป็นไลบรารีสำหรับการประมวลผลภาพ (Image Processing) ซึ่งเป็นไลบรารีโอเพนซอร์ส (Open Source) สามารถดาวน์โหลดใช้งานได้ฟรี ไลบรารีต่างๆ ของ Open CV ได้พัฒนาขึ้นด้วยบริษัทอินเทล (Intel) จุดเด่นในด้านความสามารถของไลบรารี Open CV คือสามารถประมวลผลภาพดิจิทัลได้ทั้งภาพนิ่ง และภาพเคลื่อนไหวเช่น ภาพจากกล้องวิดีโอ หรือไฟล์วิดีโอ เป็นต้น โดยไม่มีขีดจำกัดทางด้านฮาร์ดแวร์ทำให้ Open CV สามารถพัฒนาโปรแกรมได้หลากหลายภาษา

รวมถึงมีฟังก์ชันสำเร็จรูปสำหรับการจัดการข้อมูลภาพ และการประมวลผลภาพพื้นฐานเช่น การหาขอบภาพ การกรองข้อมูลภาพ เป็นต้น



รูปที่ 3.6 แสดงการแปลง Text File ไปเป็น รูปภาพ โดยใช้ Python Imaging Library (PIL)

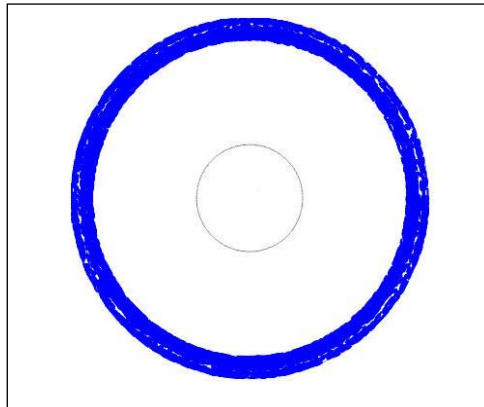
จากข้างต้นเมื่อได้รูปภาพที่แปลงมาจาก Text File แล้วกระบวนการต่อไปคือขั้นตอนการหาขอบภาพซึ่งโดยใช้วิธีการของ Canny.



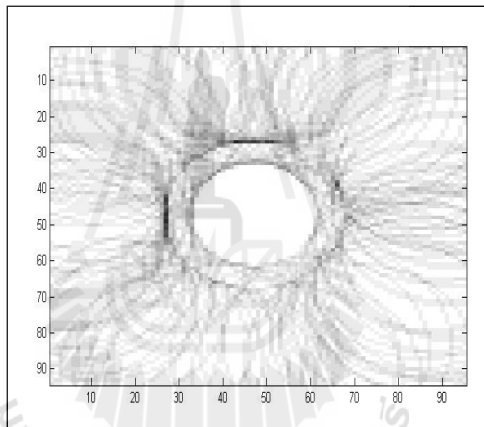
รูปที่ 3.7 แสดงภาพเมื่อผ่านขั้นตอนการหาขอบด้วย Canny Method

เมื่อเราได้รูปภาพที่ผ่านการหาขอบด้วยวิธีการ Canny แล้วขั้นตอนต่อไปคือการนำภาพที่ได้เข้ากระบวนการ Hough Transform โดยในงานวิจัยชิ้นนี้ได้เลือกใช้ Circle Hough Transform (CHT) เพื่อตรวจจับเส้นโค้งหรือวงกลมที่อยู่ในภาพ และใช้ Hough Transform (HT) เพื่อตรวจจับเส้นตรงที่อยู่ในรูปภาพ

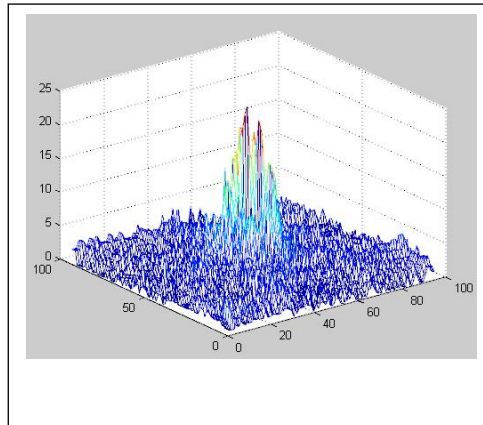




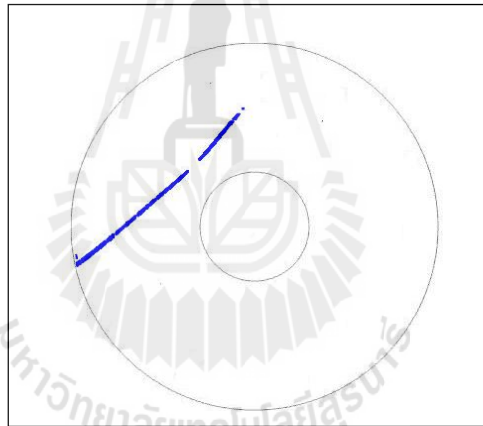
รูปที่ 3.8 แสดงตัวอย่างรูปต้นฉบับ Circum Pattern ก่อนเข้ากระบวนการ Circle Hough Transform



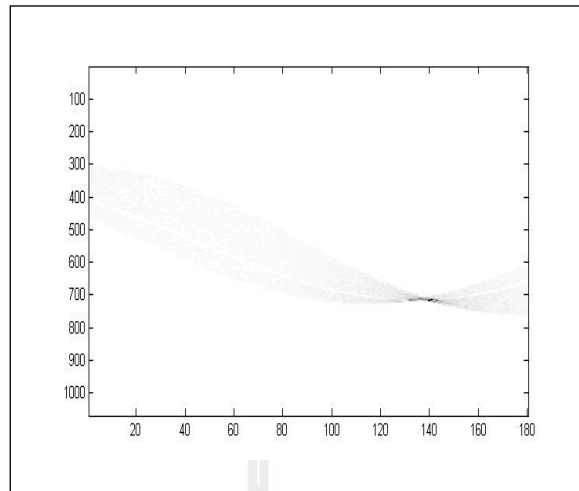
รูปที่ 3.9 แสดง (Circum Pattern) ในรูปแบบ Accumulator 2D หลังผ่านกระบวนการ Circle Hough Transform



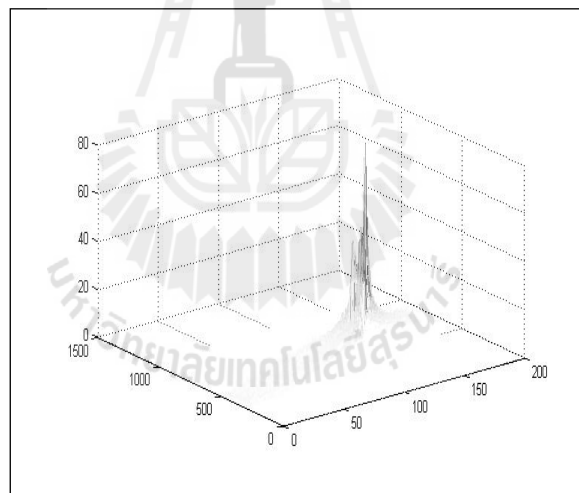
รูปที่ 3.10 แสดง (Circum Pattern) ในรูปแบบ Accumulator 3D หลังผ่านกระบวนการ  
Circle Hough Transform



รูปที่ 3.11 แสดงตัวอย่างรูปต้นฉบับ Line Pattern ก่อนเข้ากระบวนการ Hough Transform



รูปที่ 3.12 แสดง (Line Pattern) ในรูปแบบ Accumulator 2D หลังผ่านกระบวนการ Hough Transform



รูปที่ 3.13 แสดง (Line Pattern) ในรูปแบบ Accumulator 3D หลังผ่านกระบวนการ Hough Transform

ในการคัดแยกรูปแบบ Radial line และแบบ Straight line จะเป็นการค้นหาว่าในรูปแบบนั้นๆ มีเส้นหรือขอบเป็นลักษณะ Radial line และแบบ Straight line หรือไม่ซึ่งวิธีการตรวจจับก็คือ การการนับจำนวนที่มีการ Vote จนเกินค่าที่กำหนดไว้ซึ่งจะดูได้จากรูปกราฟที่ Peak ขึ้นมา ในรูปที่ 3.10 และ 3.13 ซึ่งถ้าระบบตรวจพบว่ามีค่า Peak เกินกว่าที่กำหนดไว้ก็จะรายงานออกมาโดยลำดับขั้นของการค้นหาจะเริ่มหารูปแบบ Radial line ถ้าพบก็จะแสดงผลออกมาเป็น Radial line pattern หรือถ้าตรวจไม่พบ ระบบก็จะนำภาพที่ได้ไปทำการหา Straight line ต่อ ถ้าพบก็จะแสดงผลออกมาเป็น Straight line pattern หรือถ้าไม่พบเลยนั้นหมายความว่ารูปแบบที่อยู่บนแผ่นบันทึกข้อมูลนั้น ไม่ได้อยู่ในรูปแบบดังที่กล่าวมาข้างต้นเลยระบบก็จะแสดงผลออกมาเป็น Other เป็นต้น



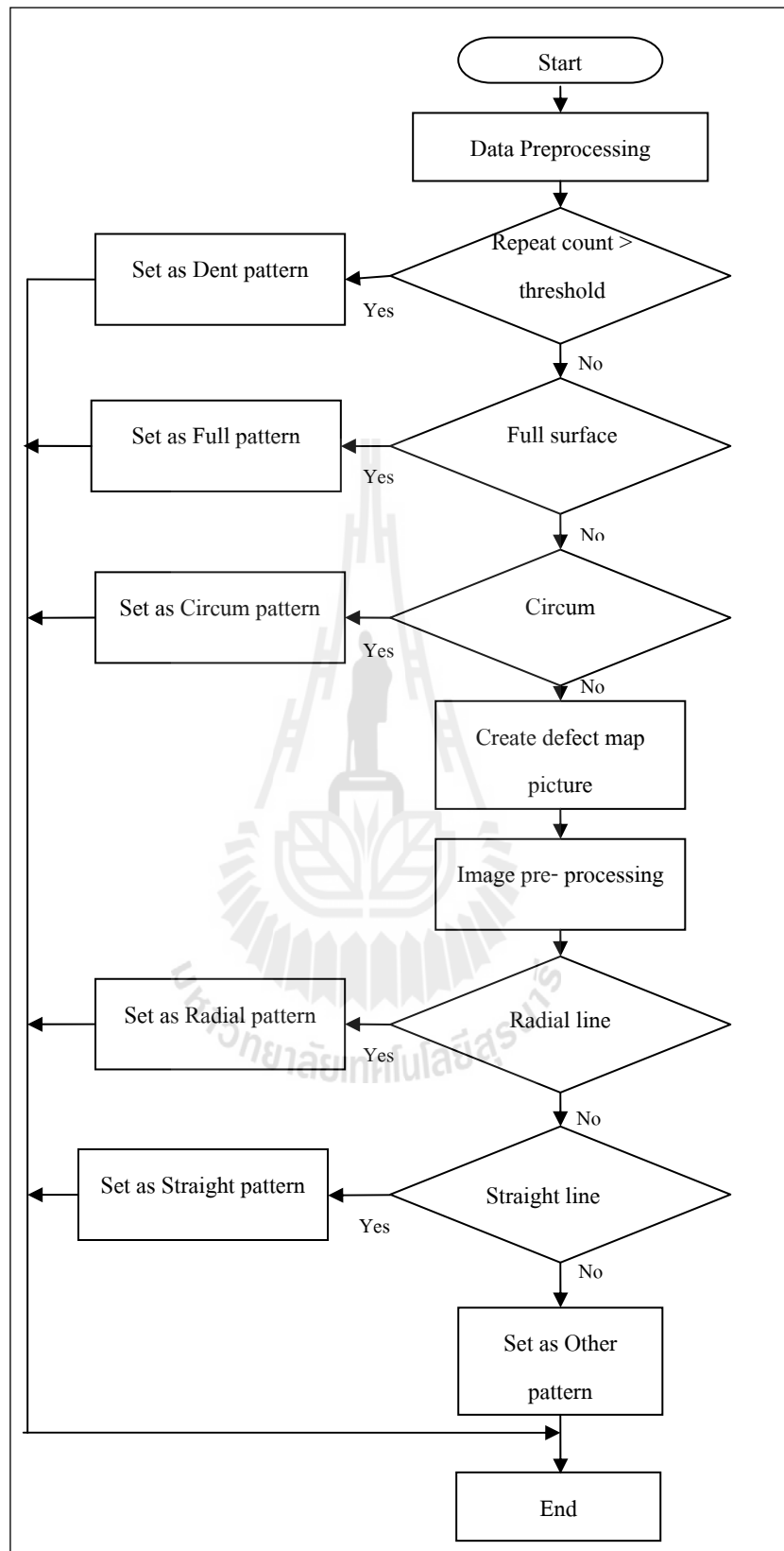
## บทที่ 4

### การพัฒนาเครื่องมือและผลการทดลอง

จากแบบจำลองที่ได้สร้างขึ้นในบทที่แล้วจะถูกนำมาเพื่อทำการพัฒนาการคัดแยกของเสียที่ปรากฏบนแผ่นบันทึกข้อมูล โดยได้แบ่งวิธีการออกเป็นสองส่วนสำคัญได้แก่การแยกรูปแบบการวิเคราะห์ของเสียบนแผ่นบันทึกข้อมูลออกเป็นสามชุดคือ ชุดแรกใช้วิธีการวัดความหนาแน่นของของเสียบนของแผ่นบันทึกข้อมูล โดยการนับจำนวนอัตราการเกิดของเสียซ้ำในจุดเดิมบนแผ่นบันทึกข้อมูลซึ่งถ้าค่าตัวนับเกินกว่าที่กำหนดไว้จะถือว่าของเสียที่เกิดบนแผ่นบันทึกข้อมูลเป็นแบบที่เกิดจากรอยยุบตัวของแผ่นบันทึกข้อมูล ส่วนการวิเคราะห์ของเสียบนแผ่นบันทึกข้อมูลในส่วนที่สองคือการวิเคราะห์แบบการหารค่าดัชนีการแปรปรวนของของเสียที่เกิดขึ้นบนแผ่นบันทึกข้อมูล โดยในงานวิจัยนี้ได้ใช้ค่าดัชนีการแปรปรวนสองค่ามาทำการวิเคราะห์คือ ค่าดัชนีการแปรปรวนในเชิงมุมและค่าดัชนีความแปรปรวนในเชิงระยะทาง โดยใช้โครงข่ายประสาทเทียมมาเป็นตัวช่วยในการตัดสินใจ โดยในรูปแบบนี้เราจะใช้สำหรับการวิเคราะห์สำหรับของเสียที่เกิดบนแผ่นบันทึกข้อมูลในสองรูปแบบคือ ของเสียในรูปแบบที่กระจายเต็มทั่วทั้งแผ่น, ของเสียในรูปแบบเป็นวงรอบ และการวิเคราะห์ของเสียในชุดที่สามที่งานวิจัยนี้ได้นำเสนอคือการวิเคราะห์ของเสียบนแผ่นบันทึกข้อมูล โดยใช้เทคนิคการประมวลผลภาพเข้ามาวิเคราะห์ของเสียบนแผ่นบันทึกข้อมูลในรูปแบบที่เป็นเส้นตรงและเส้นโค้ง

#### 4.1 ขั้นตอนการคัดแยกรูปแบบของเสีย

ขั้นตอนการคัดแยกรูปแบบของเสียบนแผ่นบันทึกข้อมูลสามารถแบ่งเป็นขั้นตอนได้ตามรูป 4.1



รูปที่ 4.1 แสดงผังกระบวนการคัดแยกรูปแบบของเสี้ยนบนแผ่นบันทึกข้อมูล

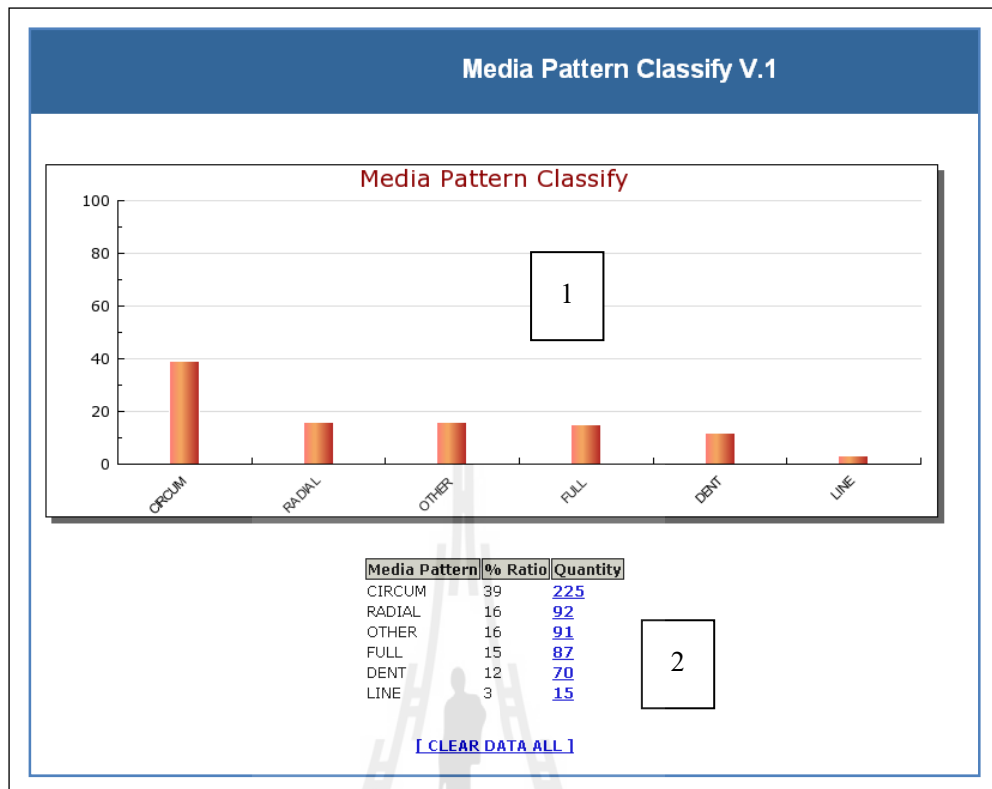
## 4.2 การพัฒนาเครื่องมือเพื่อใช้ในการคัดแยกของเสียบนแผ่นบันทึกข้อมูล

เครื่องมือที่ใช้ในการคัดแยกรูปแบบของของเสียบนแผ่นบันทึกข้อมูลจะเป็นซอฟต์แวร์โดยจะมีหลักการทำงานที่เริ่มมาจากการรับข้อมูลอินพุตที่เป็นเท็กไฟล์ซึ่งบรรจุข้อมูลตำแหน่งของเสียอยู่ภายในและทำการประมวลผลโดยใช้วิธีการตามที่ได้ออกแบบไว้และนำผลลัพธ์ที่ได้แสดงผลให้กับผู้ใช้งานจากการศึกษาคุณลักษณะเด่นของโปรแกรมภาษาไพธอน จึงทำให้เลือกที่จะใช้โปรแกรมภาษาไพธอนในการคัดแยกของเสียบนแผ่นบันทึกข้อมูลโดยคุณลักษณะเด่นของโปรแกรมภาษาไพธอน มีดังต่อไปนี้

- 1) โปรแกรมภาษาไพธอน ไม่คิดมูลค่าการใช้งาน และเป็นภาษาที่มีความยืดหยุ่นสูง
- 2) โค้ด (Code) ที่เขียนด้วยโปรแกรมภาษาไพธอน สามารถนำไปรัน (Run) บนระบบปฏิบัติการอื่น ๆ ได้ เช่น Linux, Ms-Windows (95, 98, NT, 2000, XP), /anuga/, be-OS, OS/2, VMS, QNX และระบบอื่น ๆ อีกมากมาย
- 3) โปรแกรมภาษาไพธอน สนับสนุนเทคโนโลยีของ MS-Windows
- 4) โปรแกรมภาษาไพธอนเป็นไดนามิก (Dynamic Typing) คือ สามารถเปลี่ยนชนิดข้อมูลทำได้ง่ายและสะดวก
- 5) โปรแกรมภาษาไพธอน มีโครงสร้างข้อมูลที่สามารถใช้ได้ภายในโปรแกรมภาษาไพธอน ประกอบด้วย ลิสต์ ดิกชันนารี สตริง ที่ง่ายต่อการใช้งาน และมีประสิทธิภาพสูง
- 6) โปรแกรมภาษาไพธอนมีเครื่องมือต่าง ๆ มากมาย เช่น การประมวลผลเท็กไฟล์ การเรียงข้อมูล การต่อเชื่อมสตริง การตรวจสอบเงื่อนไขของข้อความ การแทนที่ค่า เป็นต้น
- 7) โปรแกรมภาษาไพธอนมีมอดูล (Module) ที่สร้างขึ้นจากนักพัฒนาสนับสนุนมากมาย ได้แก่ COM, Image < XML เป็นต้น
- 8) โปรแกรมภาษาไพธอนจัดการหน่วยความจำอย่างอัตโนมัติ สามารถจัดการพื้นที่หน่วยความจำที่ไม่ต่อเนื่องให้ทำงานได้อย่างมีประสิทธิภาพ

### 4.2.1 หลักการทำงาน

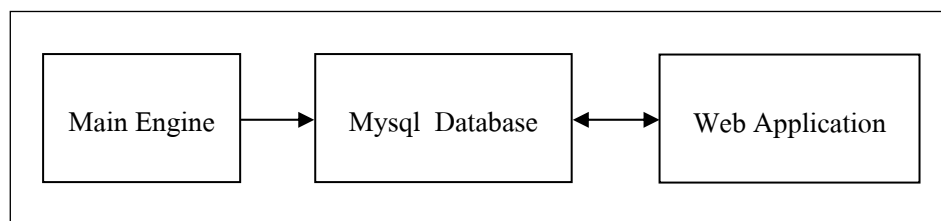
สำหรับการทำงาน Software ที่สร้างขึ้นมานั้นจะประกอบด้วยส่วนที่เป็น Engine หลักที่พัฒนาขึ้นมาโดยใช้ภาษาไพธอน และมีการแสดงผลผ่าน Web Application โดยที่ทั้งตัว Engine และ Web Application จะติดต่อกันผ่านตัวกลางคือ Database Mysql ซึ่งจะทำให้ผู้ใช้มีความสะดวกมากขึ้นเพราะไม่จำเป็นต้องลง Program ไว้ที่เครื่องของตัวเองแต่จะไปใช้ทรัพยากรทางฝั่งของ Server แทนและวิธีนี้ก็สะดวกต่อผู้พัฒนาด้วยเพราะเมื่อมีการพัฒนา Engine ใหม่ก็เพียงแก้ไขแค่ทางฝั่ง Server เท่านั้น



รูปที่ 4.2 แสดงหน้าแสดงผลผ่าน Web Application

ส่วนที่ 1 แสดงผลในรูปแบบ Graph โดยหน่วยที่แสดงในแนวแกน Y จะแสดงในรูปของเปอร์เซ็นต์ และในแนวแกน X จะแสดงรูปแบบของ Pattern

ส่วนที่ 2 จะแสดงผลในรูปแบบของตาราง โดยจะบอกถึงจำนวนและเปอร์เซ็นต์ของแต่ละรูปแบบที่สามารถคัดแยกได้และในช่องของ Quantity เราสามารถคลิกที่ตัวเลขเพื่อให้แสดงผลรายละเอียดการเสียของแต่ละตัวได้ด้วยเป็นต้น



รูปที่ 4.3 แสดงการติดต่อกันระหว่าง Engine และ Web Application



### 4.3 ผลการทดลอง

สำหรับข้อมูลที่จะนำมาใช้ในการทดสอบในงานวิจัยชิ้นนี้คือข้อมูลจากการจำลองรูปแบบของเสียบนแผ่นบันทึกข้อมูลทั้ง 5 รูปแบบและนำมาทำการทดสอบประสิทธิภาพของซอฟต์แวร์ที่สร้างขึ้นมาเพื่อวิเคราะห์และคัดแยกรูปแบบของเสียบนแผ่นบันทึกข้อมูลซึ่งผลการทดลองที่ได้มีดังนี้

- การคัดแยกรูปแบบของเสียบนแผ่นที่เกิดจากรอยยวบยัวตัวของแผ่นบันทึกข้อมูล มีความแม่นยำ 100%
- การคัดแยกรูปแบบของเสียบนแผ่นกระจายตัวทั่วทั้งแผ่น มีความแม่นยำ 85%
- การคัดแยกรูปแบบของเสียบนแผ่นเป็นวงรอบ มีความแม่นยำ 83%
- การคัดแยกรูปแบบของเสียบนแผ่นเส้นตรง มีความแม่นยำ 100%
- การคัดแยกรูปแบบของเสียบนแผ่นโค้ง มีความแม่นยำ 90%



## บทที่ 5

### สรุปและข้อเสนอแนะ

#### 5.1 สรุปผลงานวิจัย

การวิเคราะห์อาการเสียบนแผ่นบันทึกข้อมูลในฮาร์ดดิสก์ที่งานวิจัยนี้ได้นำเสนอนั้น เป็นวิเคราะห์อาการเสียบนแผ่นบันทึกข้อมูลของฮาร์ดดิสก์อีกวิธีหนึ่ง โดยได้จำลองจากการอาศัยข้อมูลที่ได้ออกจกเก็บไว้ในกระบวนการผลิตมาใช้วิเคราะห์หาสาเหตุของปัญหาที่อาจจะเกิดขึ้นซึ่งข้อมูลที่น่ามาพิจารณาในงานวิจัยชิ้นนี้คือข้อมูลของเสียบนแผ่นบันทึกข้อมูลโดยที่ลักษณะของอาการเสียบนแผ่นบันทึกข้อมูลนั้นจะสามารถอธิบายได้โดยอาศัยรูปแบบการเกิดของเสียบนแผ่นบันทึกข้อมูลซึ่งลักษณะที่เกิดขึ้นจะสามารถบ่งชี้ถึงสาเหตุที่แท้จริงของการเสียบนแผ่นบันทึก

ในการพิจารณาลักษณะของตำแหน่งที่จุดเสียบนแผ่นบันทึกข้อมูลนั้น จะพิจารณาโดยอาศัยเครื่องมือที่ได้มาจากการศึกษา การออกแบบตามที่ได้นำเสนอและมีการพัฒนาเป็นซอฟต์แวร์ ซึ่งในส่วนของการออกแบบและพัฒนาเครื่องมือนี้ได้นำเอาความรู้และเทคนิคการวิเคราะห์โดยใช้วิธีการทางการประมวลผลภาพและองค์ความรู้ด้านอื่นๆมาใช้ประกอบกันซึ่งเป็นการนำเอาองค์ความรู้ที่ได้ในสถานศึกษานำมาใช้งานให้เกิดผลในทางปฏิบัติ

โดยผลที่ได้จากการทดลองนั้น ได้ผลเป็นที่น่าพอใจในระดับหนึ่งและมีความรวดเร็วในการทำงานสูงแต่เมื่อนำไปใช้จริงควรจะมีการปรับแต่งค่าตัวแปรและค่าพารามิเตอร์ต่างๆและจำเป็นต้องมีการปรับสอนนิรอนเน็ตเวิร์คใหม่ในกรณีที่มีรูปแบบของเสียบนแผ่นบันทึกข้อมูลใหม่ และข้อควรระวังในการปรับค่าพารามิเตอร์นั้นควรจะต้องใช้ความรอบคอบและควรมีการวางแผนการปรับแต่งและการทดสอบหลังการปรับแต่งทุกครั้งเพื่อให้ไม่ให้เกิดผลกระทบกับความสามารถของระบบ

## 5.2 ข้อเสนอแนะในงานวิจัยต่อไป

รูปแบบของเสียที่สามารถคัดแยกได้ในงานวิจัยนี้ยังเป็นแบบกว้างๆซึ่งในความเป็นจริงแล้วยังมีรูปแบบอื่นๆอีกมากมายซึ่งสามารถแบ่งแยกย่อยลงไปได้อีกดังนั้นการพัฒนาเครื่องมือเพื่อเพิ่มความสามารถและความแม่นยำในการคัดแยกรูปแบบได้มากขึ้นย่อมทำให้เครื่องมือที่ทำขึ้นมามีประสิทธิภาพมากยิ่งขึ้นด้วย

นอกจากการพัฒนาซอฟต์แวร์เพื่อวิเคราะห์สาเหตุของการเสียเพียงอย่างเดียวควรจะมีการนำอัลกอริทึมที่ได้ไปประยุกต์ใช้ในงานในกระบวนการผลิตแบบอัตโนมัติเพื่อทำการคัดแยกแผ่นบันทึกข้อมูลที่เสียออกจากแผ่นบันทึกข้อมูลที่ดีเป็นต้น



## รายการอ้างอิง

- วราชัย สิงห์จั้งหรีด. การวิเคราะห์อาการเสียของฮาร์ดดิสก์โดยวิธีการเหมืองข้อมูล. วิทยานิพนธ์ปริญญาโทมหาบัณฑิต. ภาควิชาวิศวกรรมไฟฟ้า บัณฑิตวิทยาลัย มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2550.
- Abdullah A. M., Guo X. G. and Chao B. **Hard Disk Drive Mechatronics and Control**. New York : CRC Press , 2007.
- Ballard D. H. **Generalizing the Hough transform to detect arbitrary shapes**. In the proceedings of Pattern Recognition, pp. 111-122, 1981.
- Bergen J. R. and Shvaytser H. **A probabilistic algorithm for computing Hough Transforms**. In the Journal of Algorithms, pp. 639-656, 1991.
- Bryan, S. M. **Lecture 15: Segmentation (edge based, hough transform)**. Brigham Young University: Lecture Notes, 2000.
- Cheng Z. and Lin Y. **Efficient technique for ellipse detection using restricted randomized Hough transform**. In the proceedings of the International Conference on Information Technology (ITCC'04), Volume 2, pp. 714-718, 2004.
- Duda R. O. and Hart P. E. **Use of the Hough Transform To Detect Lines and Curves In Pictures**. In the Communications of ACM, pp. 11-15, 1972.
- Howard E. B. **Digital Imaging THEORY AND APPLICATIONS**. USA : McGraw-Hill , 1997.
- Illingworth J. and Kittler j. **A survey of the Hough Transform**. In the proceedings of International Conference on Computer Vision, Graphics and Image Processing, pp. 87-116, 1988.
- Kimme, C., Ballard D. and J. Sklansky. **Finding Circles by an Array of Accumulators**. In the Communications of ACM, pp. 120-122, 1975.
- Kiryati N., Eldar Y., and Bruckstein A. M. **A probabilistic Hough Transform**. In the proceedings of Pattern Recognition, pp. 303-316, 1991.
- Mohamed R. **Detecting Coins with Different Radii based on Hough Transform in Noisy and Deformed Image**. In the proceedings of GVIP Journal, Volume 7, Issue 1, April, 2007.

Nick E. **Digital Image Processing A Practical Introduction Using Java.** England : Peason Education Limited , 2000.

Shylaja, S. S., Balasubramanya M., Natarajan N., Muthuraj R., Ajay S. **Feed Forward Neural Network Base Eye Localization and Recognition Using Hough Transform.** IJACSA Vol. 2, No.3, pp. 104-109, March 2011.

Tong Y., Wang H., Pi D. and Zhang Q., **Fast Algorithm of Hough Transform- Based Approaches for Fingerprint Matching.** In the Proceedings of the Sixth World Congress on Intelligent Control and Automation, Volume 2, pp. 10425-10429, 2006.

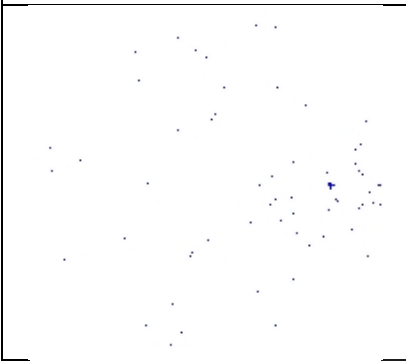
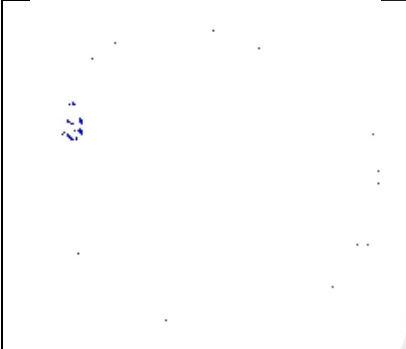
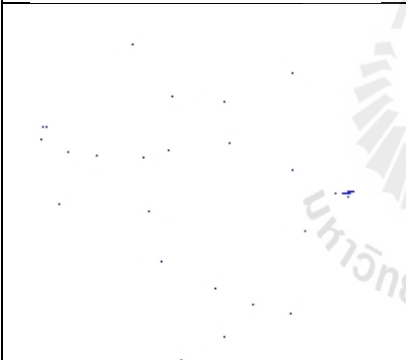
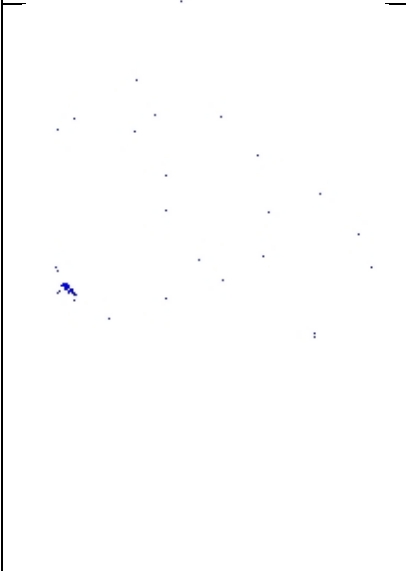
Wilhelm B. and Mark J. B. **Principles of Digital Image Processing.** UK : Springer Press, 2009.

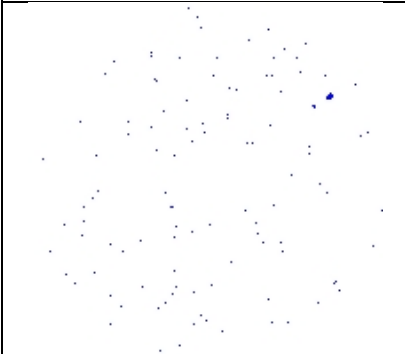
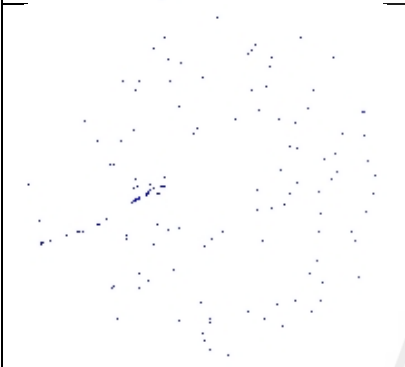

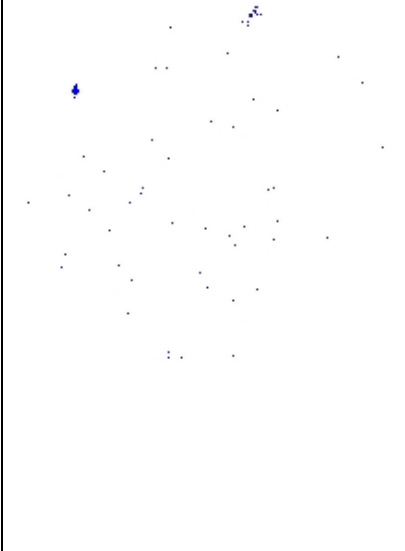




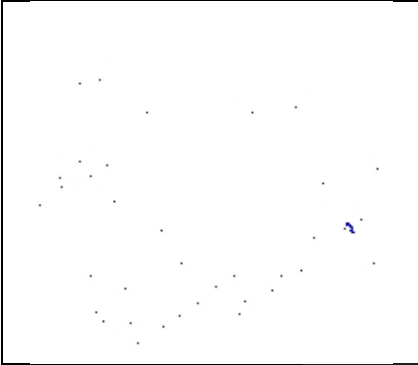
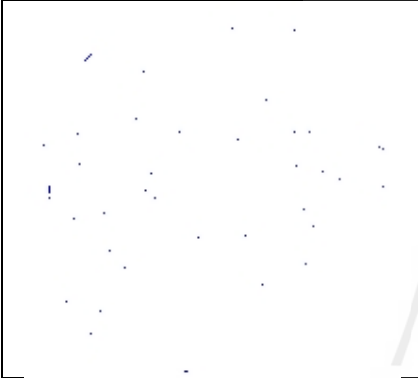

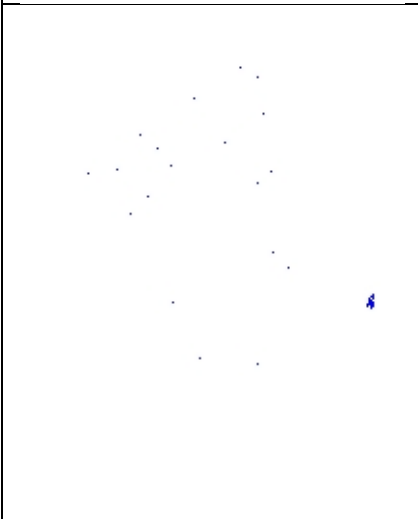
ภาคผนวก ก

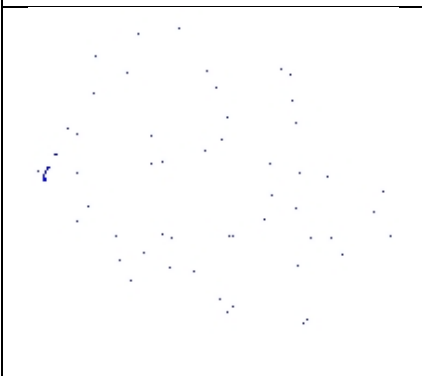
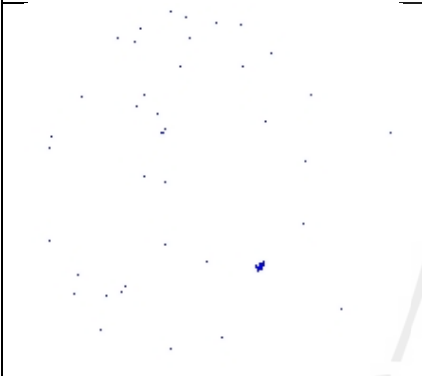
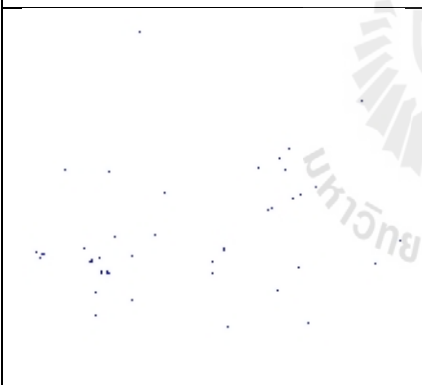
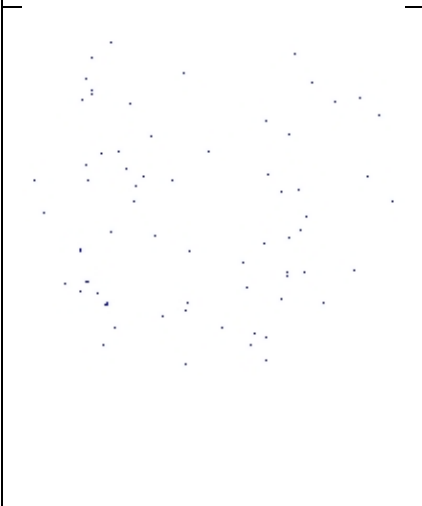
ข้อมูลที่ใช้ตรวจสอบโปรแกรมที่สร้างขึ้น

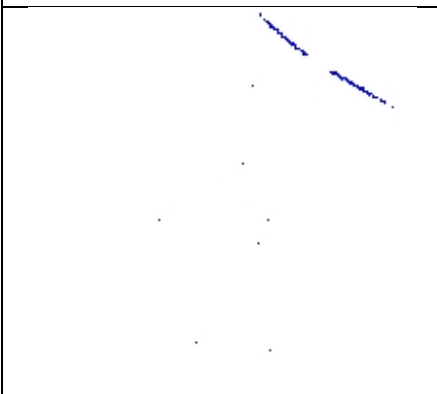
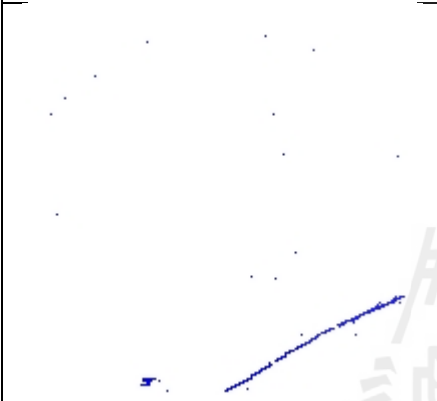
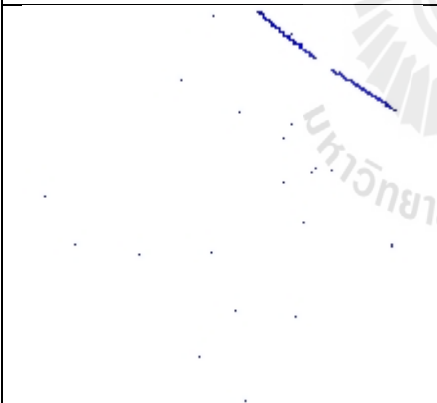
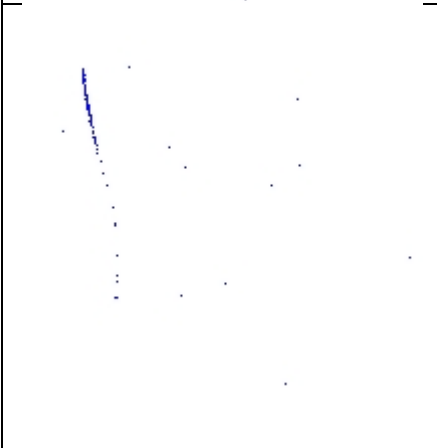
รูปแบบของเสีย	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	3.749	17.301	DENT
	1.636	33.361	DENT
	4.66	24.201	DENT
	6.251	29.424	DENT

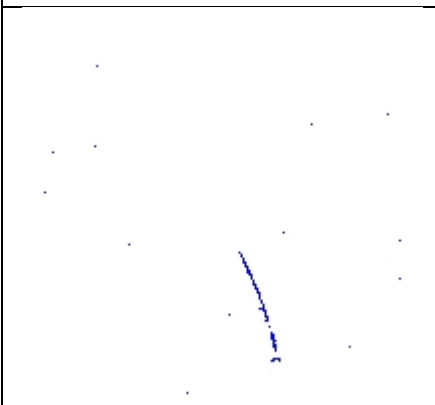
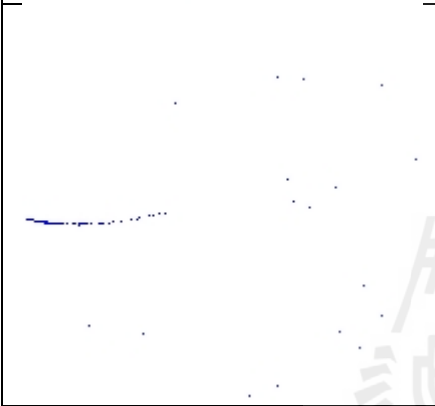
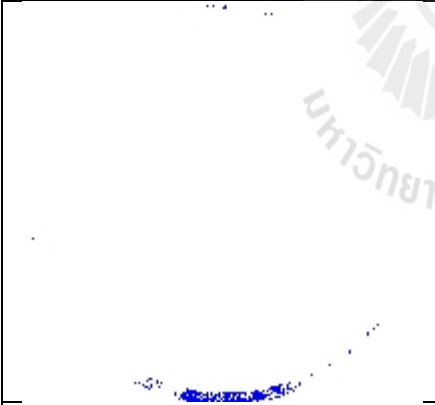
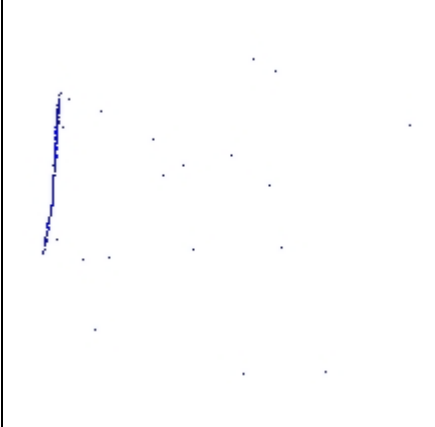
รูปแบบของเสีย	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	1.424	6.938	DENT
	1.318	6.196	DENT
	5.651	27.108	DENT
	2.65	12.945	DENT

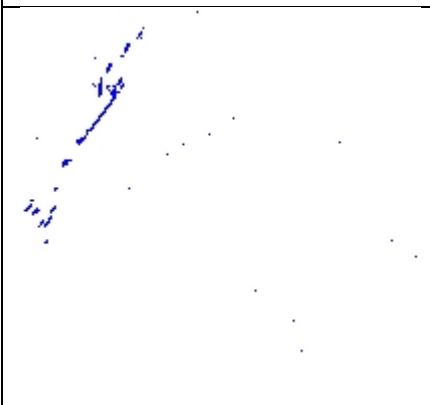
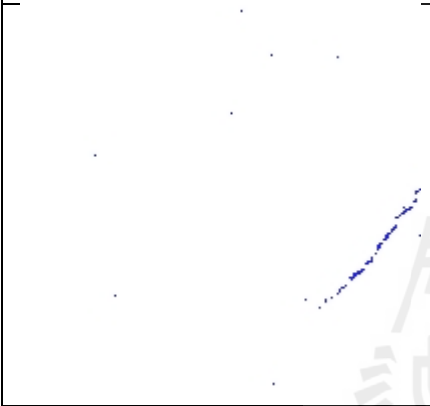
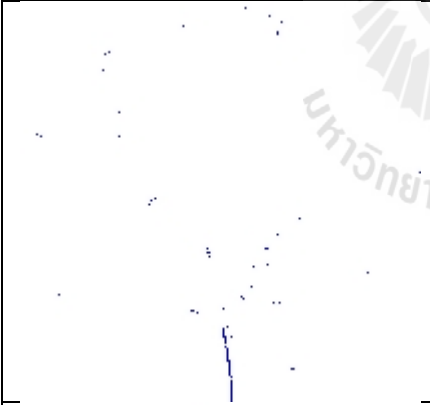
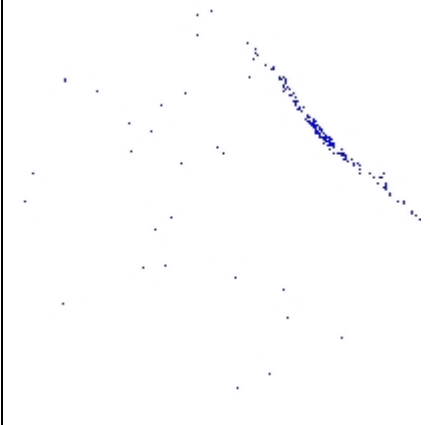


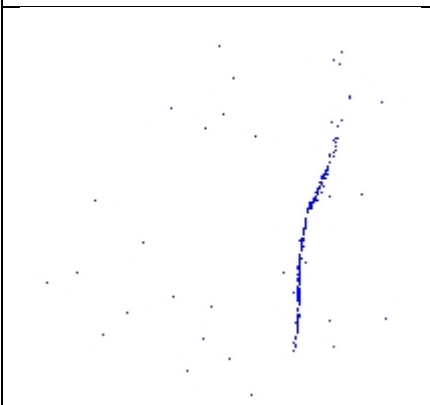
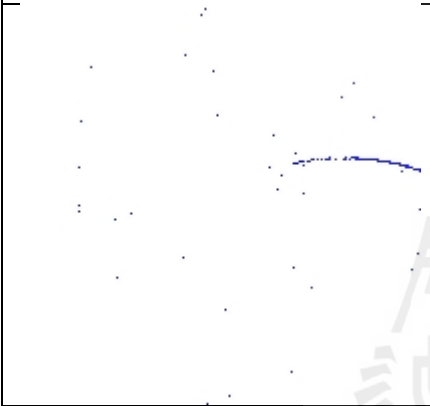
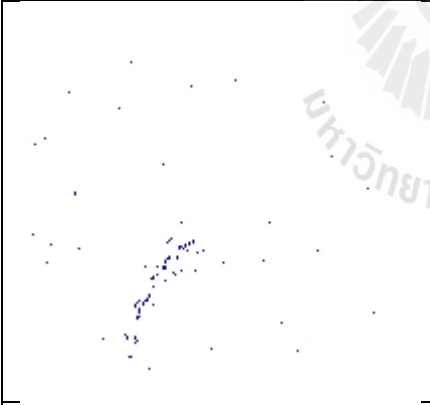
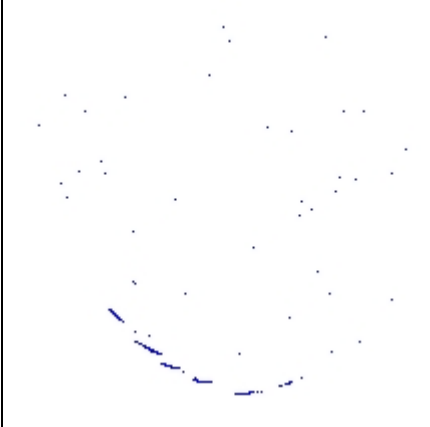
รูปแบบของเสีย	ค่าดัชนีการจับ กลุ่มเชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	3.614	26.352	DENT
	3.967	15.766	DENT
	2.196	14.663	DENT
	6.65	33.348	DENT

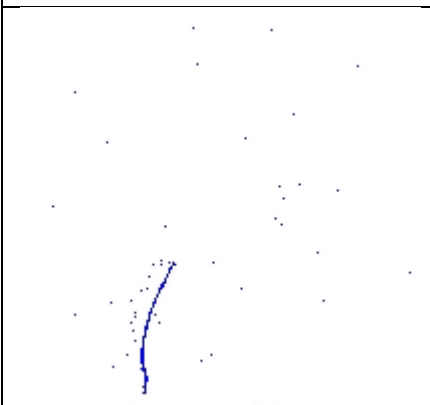
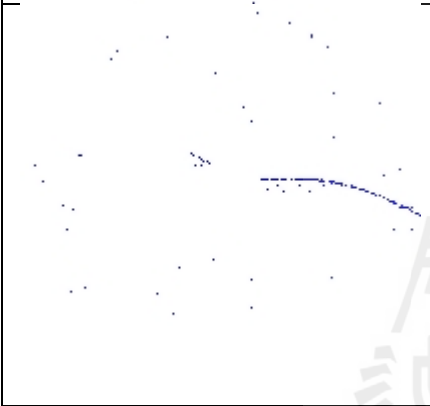
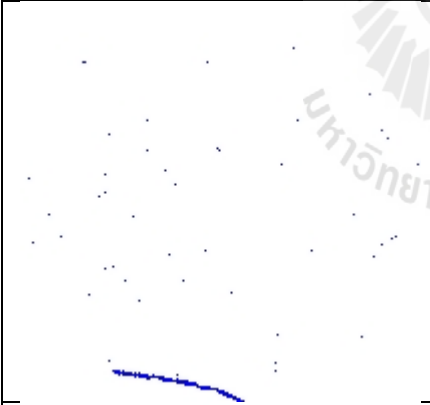
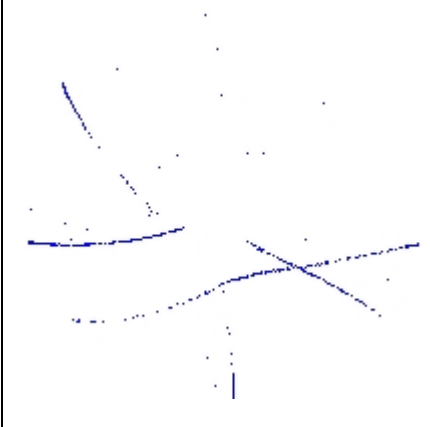
รูปแบบของเสีย	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	3.05	9.499	DENT
	3.735	17.752	DENT
	3.503	33.352	DENT
	2.221	15.66	DENT

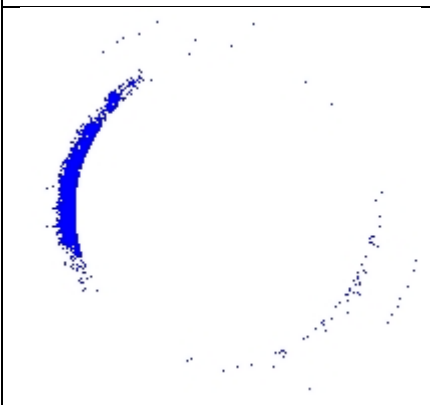
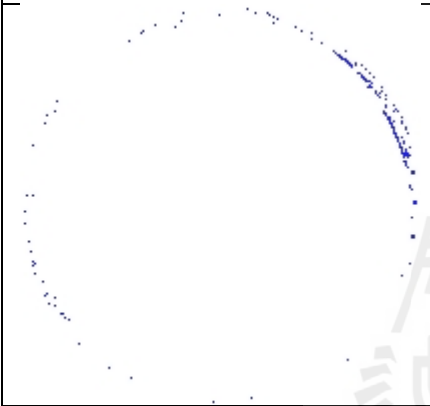
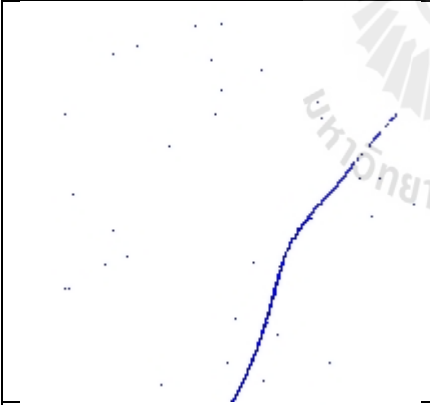
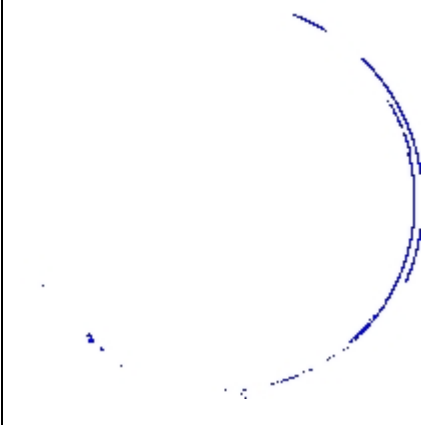
รูปแบบของเสีย	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	13.787	64.278	LINE
	10.322	28.89	LINE
	5.366	31.465	LINE
	3.548	34.513	LINE

รูปแบบของเสีย	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	2.323	45.074	LINE
	2.21	34.083	LINE
	1.0	69.818	LINE
	8.128	33.249	LINE

รูปแบบของเสีย	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	7.294	27.144	LINE
	5.743	84.211	LINE
	2.205	27.328	LINE
	3.368	19.927	LINE

รูปแบบของเสีย	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	1.723	19.815	Radial
	1.813	20.866	Radial
	1.76	17.867	Radial
	3.85	13.771	Radial

รูปแบบของเสีย	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	1.486	19.072	Radial
	1.231	15.628	Radial
	3.558	13.715	Radial
	0.999	16.293	Radial

รูปแบบของเสีย	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสีย
	2.519	13.707	Radial
	1.0	12.666	Radial
	1.821	13.713	Radial
	1.4	52.958	Radial



รูปแบบของเสี้ยน	ค่าดัชนีการจับกลุ่ม เชิงระยะทาง	ค่าค่าดัชนีการจับ กลุ่มเชิงมุม	ชนิดของเสี้ยน
	1.0	7.959	Radial





ภาคผนวก ข

บทความที่ได้รับการตีพิมพ์เผยแพร่

## รายชื่อบทความที่ได้รับการตีพิมพ์เผยแพร่ในขณะศึกษา

Liangwongsan, S., Oonsivilai, A. (2012). **Straight Line Defect Detection with Feed Forward Neural Network**. World Academy of Science, Engineering and Technology. Issue 69 September. (Submitted)



# Straight Line Defect Detection with Feed Forward Neural Network

S. Liangwongsan, A. Oonsivilai

**Abstract**—Nowadays, hard disk is one of the most popular storage components. In hard disk industry, the hard disk drive must pass various complex processes and tested systems. In each step, there are some failures. To reduce waste from these failures, we must find the root cause of those failures. Conventional data analysis method is not effective enough to analyze the large capacity of data. In this paper, we proposed the Hough method for straight line detection that helps to detect straight line defect patterns that occurs in hard disk drive. The proposed method will help to increase more speed and accuracy in failure analysis.

**Keywords**—Hough Transform; Failure Analysis; Media; Hard Disk Drive

## I. INTRODUCTION

**H**ARD Disk Drives (HDD) are components for data storage used in computers. They are response to various uses that the demand of HDD is inverse variation to the cost.

Therefore, the HDD industry has to balance demand with high quality production, as such to increase capacity and production technology. The defined process of HDD production is complex. It consists of various components, namely read-head, media for data storage, and motor. Due to its complexity, there are some failures occurring in each process. Most commonly found failure is from defect on the media which cannot be repaired and wastes production cost. Failure Analysis (FA) is utilized to identify the failure of defect on media and classify the pattern of defect. FA helps technicians to access problems rapidly, decreasing waste and production cost. However, FA where failures are analyzed by human is not rapid enough. Consequently, this study is to introduce a method of Failure Pattern Detection based on the Hough Transform and Image Processing for detecting straight line pattern on media to help detecting defect in HDD media for the loaded case of FA process.

## II. HOUGH TRANSFORM

The Hough transform is a technique used to find shapes in a binary digital image. By Hough Transform it is possible to find all kind of shapes that can be mathematical expressed, for instance lines, circles and ellipses, but only straight lines will be considered here. If having a white pixel in a binary image, infinity many straight lines can go through that single pixel, and each of these lines can go through other white pixels in the same image, and the more white pixels on the same line the more is this line represented in the image.

S. Liangwongsan is with the School of Mechanical Engineering, Suranaree University of Technology, Nakhon Ratchasima Thailand (e-mail: sirisak.rong@gmail.com).

A. Oonsivilai is with Center of Excellence in Electric Energy, Smart Materials, and Health Science, Postharvest Technology Research Center, School of Electrical Engineering, Institute of Engineering, Suranaree University of Technology, Nakhon Ratchasima, 30000 Thailand. (e-mail: amant@sut.ac.th).

This is the principle of the Hough transform for straight lines. As mentioned above a shape can be found if a mathematical expression can be set for the shape, and in this case where the shape is a straight line, an expression can be set as:

$$y = a * x + b \quad (1)$$

Where  $a$  is the slope, and  $b$  is where the line intersects the  $y$ -axis. These parameters,  $a$  and  $b$ , can be used to represent a straight line as single point  $(a, b)$  in the parameter-space spanned by the two parameters  $a$  and  $b$ . The problem by represent a line as a point in the  $(a, b)$  parameter-space, is that both  $a$  and  $b$  goes toward infinity when the line becomes more and more vertical, and thereby the parameter space becomes infinity large. Therefore it is desirable to find another expression of the line with some parameters that have limited boundaries. It is done by using an angle and a distance as parameters, instead of a slope and an intersection.

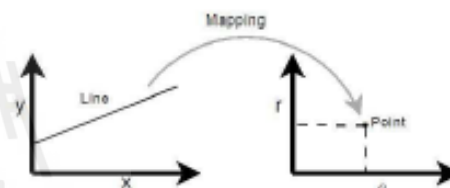


Fig. 1 Mapping of one unique line to the Hough space

An important concept for the Hough transform is the mapping of single points. The idea is that a point is mapped to all lines that can pass through that point. This yields a sine-like line in the Hough space. The principle is illustrated for a point  $p_0 = (40, 30)$  in Figure 3.

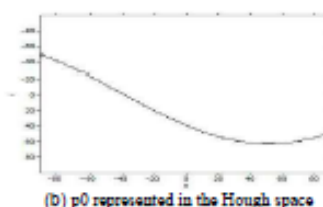
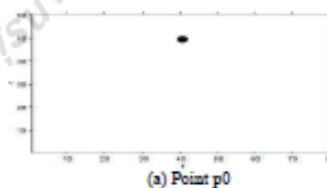


Fig. 2 Transformation of a single point ( $p_0$ ) to a line in the Hough space. The Hough space line represents all possible lines through  $p_0$ .

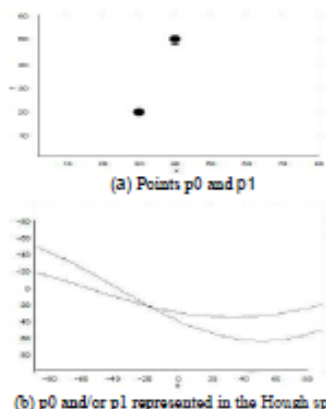


Fig. 3 Transformation of two points ( $p_0$  and  $p_1$ ) to two lines in the Hough space. The intersection of the Hough space lines indicates the line that pass through both  $p_0$  and  $p_1$ .

If the distance  $\rho$  (*rho*) is the distance from the origin to the line along a vector perpendicular to the line, and the angle  $\theta$  (*theta*) is the angle between the  $x$ -axis and the  $r$  vector (see Figure 1), Equation 1 can be written as:

$$y = -\frac{\cos(\theta)}{\sin(\theta)}x + \frac{\rho}{\sin(\theta)} \quad (2)$$



Fig. 3 *Rho* and *theta* representation of a straight line. Each line has a unique parameter set ( $\rho$ ,  $\theta$ )

The expressions here, instead of  $a$  and  $b$ , is found by trigonometrically calculations. To get an expression of  $\rho$ , Equation 2 can be rearranged to:

$$\rho = x \cdot \cos(\theta) + y \cdot \sin(\theta) \quad (3)$$

Contrary to when the parameters is  $a$  and  $b$ , the values that

$\rho$  and  $\theta$  can have are limited to:  $\theta \in [0, 180]$  in degrees or  $\theta \in 2 [0, \pi]$  in radians, and  $\rho \in 2 [-D, D]$  where  $D$  is the diagonal of the image. A line can then be transformed into a single point in the parameter space with the parameters  $\theta$  and  $\rho$ , this is also called the Hough space.

If, instead of a line, having a pixel in an image with the position ( $x$ ,  $y$ ), infinity many lines can go through that single pixel. By using Equation 3 all these lines can be transformed into the Hough space, which gives a sinusoidal curve that is unique for that pixel. Doing the same for another pixel, gives another curve that intersect the first curve in one point, in the Hough space. This point represents the line, in the image space, that goes through both pixels. This can be repeated for all the pixels on the edges, in a edge detected image.

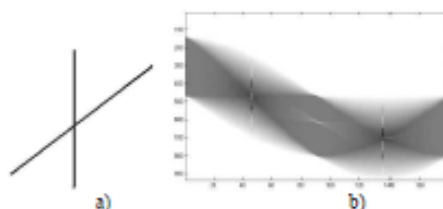


Fig. 4 An example of straight line (a) and Hough space (b)

### III. PROPOSED METHOD

#### A. Preprocessing

Raw data of media defect are created and stored in database in each local hard disk factory. We used those data for input of image and preprocess image by initially cropped to size  $100 \times 100$  resolutions and used smoothing image, edge detection, filter noise, by Canny Edge Detection Algorithm.

The Canny algorithm uses an optimal edge detector based on a set of criteria which include finding the most edges by minimizing the error rate, marking edges as closely as possible to the actual edges to maximize localization, and marking edges only once when a single edge exists for minimal response [20]. According to Canny, the optimal filter that meets all three criteria above can be efficiently approximated using the first derivative of a Gaussian function.

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (4)$$

$$\frac{\partial G(x, y)}{\partial x} = -\frac{x}{\sigma^2} G(x, y) \quad \frac{\partial G(x, y)}{\partial y} = -\frac{y}{\sigma^2} G(x, y) \quad (5)$$

where:  $g(k, l)$  = convolutional kernel

$I(x, y)$  = original image

$I'(x, y)$  = filtered image

$2N + 1$  = size of convolutional kernel

Both the Gaussian mask and its derivative are separable, allowing the 2-D convolution operation to be simplified. This optimization is not limited to software implementation only, but applies to hardware implementation as well, as shown in the next section.

The non-maximal suppression stage finds the local maxima in the direction of the gradient, and suppresses all others, minimizing false edges. The local maxima is found by comparing the pixel with its neighbors along the direction of the gradient. This helps to maintain the single pixel thin edges before the final thresholding stage.

Instead of using a single static threshold value for the entire image, the Canny algorithm introduced hysteresis thresholding, which has some adaptivity to the local content of the image. There are two threshold levels,  $t_h$  high and  $t_l$  low where  $t_h > t_l$ . Pixel values above the  $t_h$  value are immediately classified as edges. By tracing the edge contour, neighboring pixels with gradient magnitude values less than  $t_h$  can still be marked as edges as long as they are above  $t_l$ . This process alleviates problems associated with edge discontinuities by identifying strong edges, and preserving the relevant weak edges, in addition to maintaining some level of noise suppression. While the results are desirable, the hysteresis stage slows the overall algorithm down considerably.

The performance of the Canny algorithm depends heavily on the adjustable parameters,  $\sigma$ , which is the standard deviation for the Gaussian filter, and the threshold values,  $t_h$  and  $t_l$ .  $\sigma$  also controls the size of the Gaussian filter. The bigger the value for  $\sigma$ , the larger the size of the Gaussian filter becomes. This implies more blurring, necessary for noisy images, as well as detecting larger edges. As expected, however, the larger the scale of the Gaussian, the less accurate is the localization of the edge. Smaller values of  $\sigma$  imply a smaller Gaussian filter which limits the amount of blurring, maintaining finer edges in the image. The user can tailor the algorithm by adjusting these parameters to adapt to different environments with different noise levels.

Results can be further improved by performing edge detection at multiple resolutions using multi-scale representations, similar to the Marr-Hildreth algorithm [2]. This is achieved using different standard deviations, which correspond to different resolution versions of the image. Edges have zero crossing a multiple scale values. Combining maxima information from different scales allows better classification of true edges.

Convolution at multiple resolutions with large Gaussian filters require even more computation power. This may prove to be challenging to implement as a software solution for real-time applications.

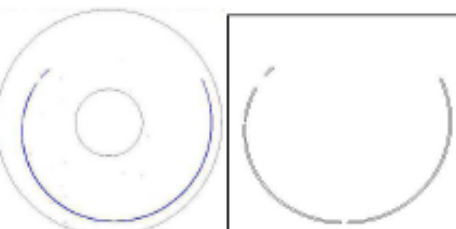
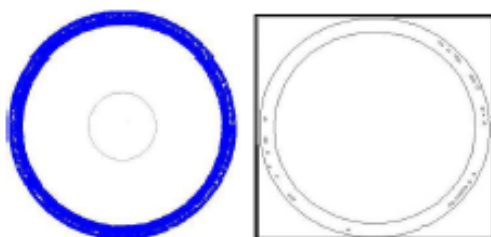


Fig. 5 defect circle pattern after passed Canny Edge Detection Algorithm

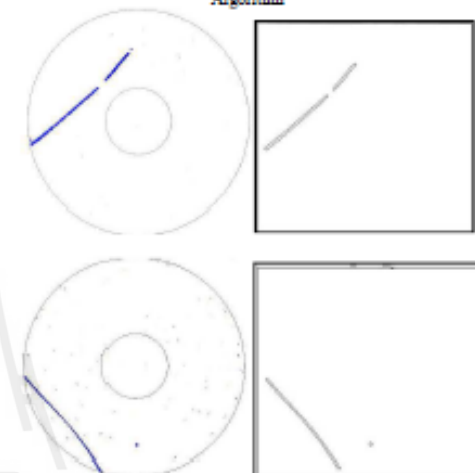


Fig. 6 defect straight line pattern after passed Canny Edge Detection Algorithm

#### B. Feature extraction by Circle Hough transform

This step used Hough transform techniques to identify position of arbitrary shape. The purpose of this technique is to find imperfect instances of object in parameter space within a certain class of shape by voting procedure. This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform. For each input image in the  $x$ - $y$  plane, a spherical filter is applied and the outcome of this is a set of points in the parameter space. These points represent the centers of probabilistic circles of pre-defined radii in the  $x$ - $y$  plane. Mapping is done for each point in the original space to a corresponding point in the original space to a corresponding point in accumulator space and increment the value of the accumulator bin.

To determine the areas where most Hough space lines intersect, an accumulator covering the Hough space is used. When an edge point is transformed, bins in the accumulator is incremented for all lines that could pass through that point. The resolution of the accumulator determines the precision with which lines can be detected.



In general, the number of dimensions of the accumulator corresponds to the number of unknown parameters in the Hough transform problem. Thus, for ellipses a 5-dimensional space is required (the coordinates of its center, the length of its major and minor axis, and its angle). For lines 2 dimensions suffice ( $r$  and  $\theta$ ). This is why it is possible to visualize the content of the accumulator.

#### C. Classification by Feed Forward Neural Network (FFNN)

The Back propagation neural network used here is Feed Forward Neural Network (FFNN). FFNN was the first and arguably simplest type of artificial neural network devised. In this network, The information moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and do the output nodes. The number of input Layers, hidden layers and output layers are adjusted to fit the data point to the curve. During the training phase the training data in the accumulator array is fed into the input layer. The data is propagated to the hidden layer and then to the output layer. This is the forward pass of the back propagation algorithm. Each node in the hidden layer gets input from all the nodes from input layer which are multiplexed with appropriate weights and summed. The output of the hidden node is the nonlinear transformation of the resulting sum. Similar procedure is carried out in the output layer. The output values are compared with target values are compared with target values and the error between two is propagated back towards the hidden layer. This is the backward pass of the back propagation algorithm. The procedure is repeated to get the desired accuracy. During the testing phase the test vector is fed into the input layer. The output of FFNN is compared with training phase output to match the correct one. This can serve as need to do recognition of facial objects.

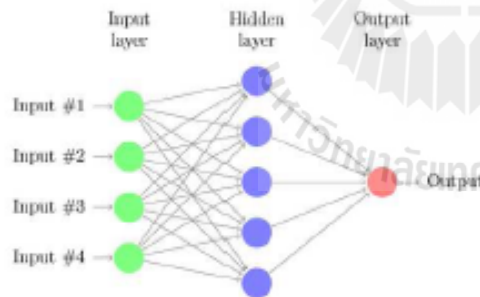


Fig. 7 Block diagram feed forward neural network

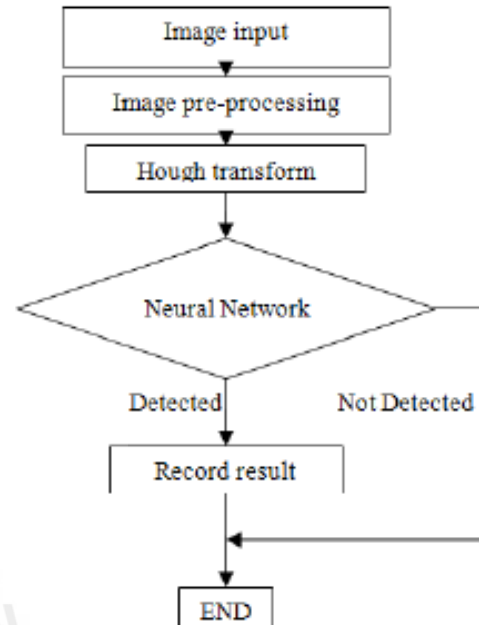


Fig. 8 Block diagram of the entire process

#### IV. EXPERIMENT RESULT

The 140 sample pictures of media defect with straight line pattern were sampled and pre - processed by image processing techniques. The samples were smoothed by Canny edge detection technique, and their noise were eliminated by Gaussian filter. It was shown that the Canny edge detection was able to both eliminating the noise and smoothing the border edge. To decrease the sample borders, the magnitude and orientation of them were calculated by the first order differentiation, then the nonmaxima suppression was employed in searching for the magnitude of gradient. To identify the pixel border and connect the border, double thresholding algorithm was employed. The sample images then were resized to 100x100 pixels for a fast image processing response.

The peak values of Straight line detection algorithm were searched in the Accumulator arrays. A threshold value was defined to represent the boundary of the straight line pattern. Then, the maximum values from the accumulator array were compared with the threshold to identify for the straight line pattern. Fig. 9 demonstrated the straight line pattern defect and the accumulator. Fig. 10 demonstrated the circle pattern depicting the peak value voted from the accumulator array, that the defect was the straight line pattern. A feed forward neural network with 30 hidden layers was trained from Table I with hidden layer is 30 network is best accuracy from this neural network the accuracy obtained is very high with computed value of 99.29% and the performance plot of neural network each layer as shown in figure 11.

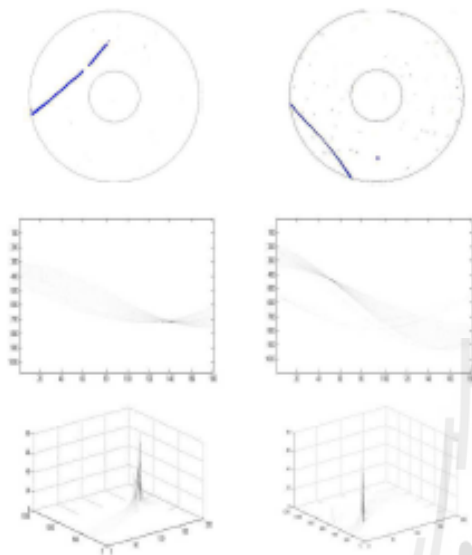


Fig. 9 Straight line pattern & Accumulator 2D,3D view

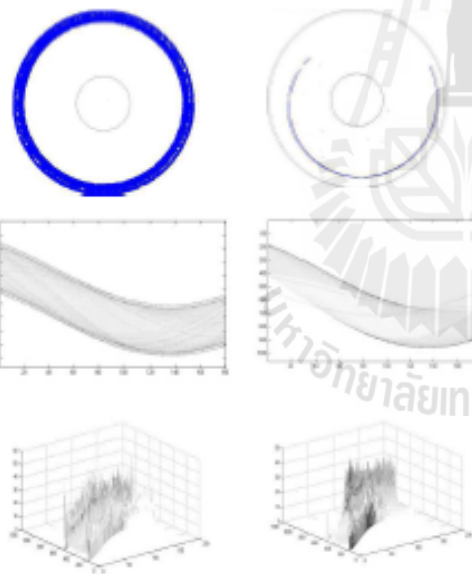


Fig. 10 Sample of circum & radial pattern & Accumulator 2D,3 Dview (this group is not straight line algorithm not detect)

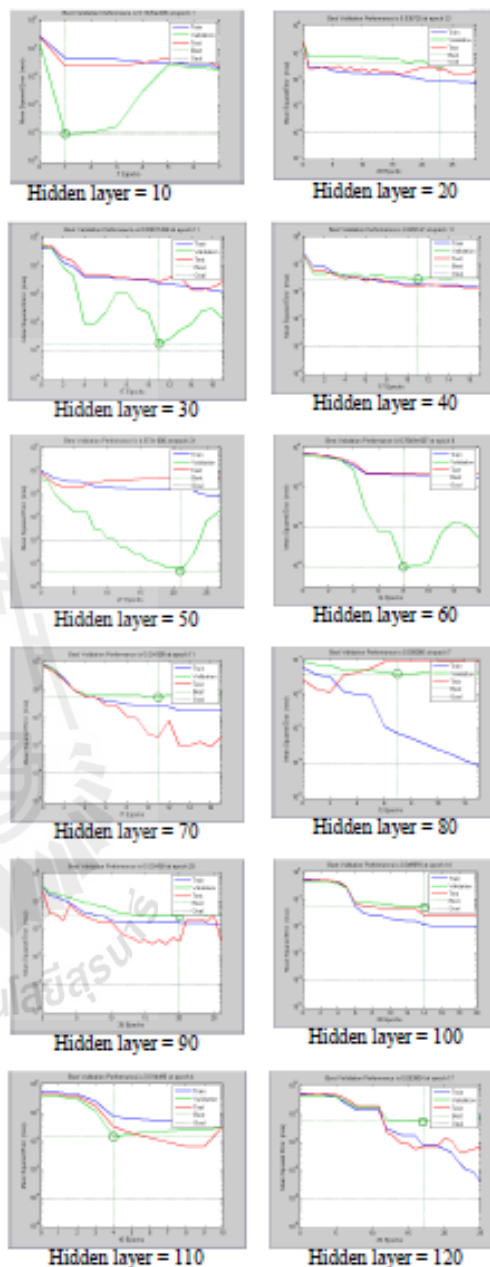


Fig. 11 Performance plot of neural network



TABLE I

NO.OF IMAGE	COMPARE ACCURACY 10-120 HIDDEN LAYER		FAULTY	%ACCURACY
	NO. HIDDEN LAYER	NO. OF DETECTED		
140	10	133	7	95.00%
140	20	137	3	97.86%
140	30	139	1	99.29%
140	40	138	2	98.57%
140	50	138	2	98.57%
140	60	133	7	95.00%
140	70	134	6	95.71%
140	80	138	2	98.57%
140	90	138	2	98.57%
140	100	135	5	96.43%
140	110	135	5	96.43%
140	120	134	6	95.71%

### V. CONCLUSION

This study's aim was to introduce an algorithm for straight line pattern detection on HDD media based on the Hough Transform method and feed forward neural network (FFNN). The experiment demonstrated a satisfying result. The developed algorithm was advantage for the failure analysis of HDD production process, such as help to access problems rapidly, decrease waste, and reduce the production cost of HDD industry. To minimize the search time duration and yield the effective search, the border lines of samples were identified by the pre - process of the image processing feed forward neural network (FFNN) method.

### ACKNOWLEDGMENT

This research was supported by research and development institute funds, Suranaree University of Technology, Nakhon Ratchasima, Thailand.

### REFERENCES

- [1] Jack Bresenham. A linear algorithm for incremental digital display of circular arcs. Communications of the ACM February, Volume 20, Number 2, February 1977.
- [2] Chester F. Carlson. Lecture 10: Hough circle transform. Rochester Institute of Technology: Lecture Notes, October 11, 2005.
- [3] Rafael C. Gonzalez and Richard E. Woods. Digital Image Processing. Prentice Hall, 2007. ISBN 0-13-168728-x.
- [4] Carolyn Kimme, Dana Ballard, and Jack Sklansky. Finding circles by an array of accumulators. Communication of the ACM, Volume 18, Number 2, February 1975.
- [5] Bryan S. Moese. Lecture 15: Segmentation (edge based, hough transform). Brigham Young University: Lecture Notes, 2000.
- [6] Shylaja S S, K N Balasubramanya Murthy, S Natarajan Nischith, Mathuraj R, Ajay S. Feed Forward Neural Network Base Eye Localization and Recognition Using Hough Transform. IJACSA Vol. 2, No. 3, pp. 104-109, March 2011
- [7] Ballard DH. "Generalizing the Hough transform to detect arbitrary shapes", In the proceedings of Pattern Recognition, pp. 111-122, 1981.
- [8] J. Illingworth and J. Kittler. "A survey of the Hough Transform", In the proceedings of International Conference on Computer Vision, Graphics and Image Processing, pp. 87-116, 1988.
- [9] JR Bergen and H Shvaytser, "A probabilistic algorithm for computing Hough Transforms", In the Journal of Algorithms, pp. 639-656, 1991.
- [10] N Kiryati, Y Eldar, and AM Bruckstein, "A probabilistic Hough Transform", In the proceedings of Pattern Recognition, pp. 303-316, 1991.
- [11] Duda RO, Hart PE, "Use of the Hough Transform To Detect Lines and Curves In Pictures", In the Communications of ACM, pp. 11-15, 1972.
- [12] Kimme, C., D. Ballard and J. Sklansky, "Finding Circles by an Array of Accumulators", In the Communications of ACM, pp. 120-122, 1975.

- [13] Mohamed Roushdy "Detecting Coins with Different Radii based on Hough Transform in Noisy and Deformed Image" In the proceedings of GVIP Journal, Volume 7, Issue 1, April, 2007.
- [14] Cheng Z. and Lin Y., "Efficient technique for ellipse detection using restricted randomized Hough transform", In the proceedings of the International Conference on Information Technology (ITCC'04), Vol.2, pp.714-718, 2004.
- [15] Yu Tong, Hui Wang, Daoying Pi and Qili Zhang, "Fast Algorithm of Hough Transform- Based Approaches for Fingerprint Matching", In the Proceedings of the Sixth World Congress on Intelligent Control and Automation, Vol. 2, pp.10425-10429, 2006.
- [16] Lam, W.C.Y., Yuen, S.Y., "Efficient technique for circle detection using hypothesis filtering and Hough transform", IEE Proceedings of Vision, Image and Signal Processing, vol. 143, Oct. 1996.
- [17] Torii, A. and Imiya A., "The Randomized Hough Transform Based Method for Great Circle Detection on Sphere", In the Proceedings of Pattern Recognition, pp. 1186-1192, 2007.
- [18] Cheng Z. and Lin Y., "Efficient technique for ellipse detection using restricted randomized Hough transform", Proc. of the International Conference on Information Technology (ITCC'04), Vol.2, pp.714-718, 2004.
- [19] Yip, R.K.K.; Leung, D.N.K. and Harold, S.O., "Line segment patterns Hough transform for circles detection using a 2-dimensional array", International Conference on Industrial Electronics, Control, and Instrumentation, Vol.3, pp. 1361-1365, 1993.
- [20] Canny, J., "A Computational Approach to Edge Detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.8, No. 6, Nov. 1986.
- [21] Simon Just Kjeldgaard Pedersen, Circular Hough Transform, Aalborg University, Vision, Graphics, and Interactive Systems, November 2007.
- [22] Marr, D., Hildreth E., "Theory of edge detection", Proceedings Royal Soc. London, vol. 207, 187-217, 1980
- [23] Sirisak L., Boonruang M., Ratchadaporn O., Anant O. "Straight Line Defect Detection Algorithm with Hough Transform" WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY ISSUE 60, DECEMBER 2011
- [24] Sirisak L., Boonruang M., Ratchadaporn O., Anant O. "Extracted Circle Hough Transform and Circle Defect Detection Algorithm" WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY ISSUE 60, DECEMBER 2011



**Sirisak Liangwongsan**, was born in Saraburi Province, Thailand, in 1982. He received B.Eng from Rajabhat Suranaree University of Technology Thanyaburi, Thailand, in Telecommunication Engineering. And currently a student M. Eng. in school of Mechanical Suranaree University of Technology, Thailand.



**Anant Oonsivilai**, was born in Khon Kaen Province, Thailand, in 1963. He received his B.Eng from Khon Khan University, and M.Eng. from King Mongkut Institute of Technology North Bangkok, Thailand and Ph.D. from Dalhousie University, Canada, all in Electrical Engineering, in 1986, 1992 and 2000, respectively. Dr. Oonsivilai is currently an Assistant Professor in school of Electrical Engineering, Suranaree University of Technology, Thailand.

## ประวัติผู้เขียน

นายสิริศักดิ์ เลียงวงษ์สันต์ เกิดวันที่ 5 กันยายน พ.ศ. 2524 ที่จังหวัดสระบุรี จบการศึกษา  
ชั้นมัธยมต้นจากโรงเรียนเทศบาลบ้านม่วง และศึกษาต่อในระดับประกาศนียบัตรวิชาชีพ (ปวช.)  
ปีที่ 1-3 ที่ วิทยาลัยเทคนิคสระบุรี ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.) ปีที่ 1-2 ที่วิทยาลัยเทคนิค  
ท่าหลวงจันทบุรีไทยอนุสรณ์ สำเร็จการศึกษาในระดับวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรม  
อิเล็กทรอนิกส์-โทรคมนาคม จากมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี เมื่อปี 2549 และได้ศึกษา  
ต่อในระดับวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาแมคคาทรอนิกส์ ในขณะที่ศึกษาต่อในระดับ  
มหาบัณฑิต ได้มีประสบการณ์การทำงานเป็นพนักงานบริษัท ซีเกท เทคโนโลยี จังหวัด  
นครราชสีมา มีผลงานวิชาการที่ได้รับการตีพิมพ์เผยแพร่ 1 บทความ ดังที่ได้รวบรวมไว้ใน  
ภาคผนวก ข.

