



## รายงาน

เรื่อง A Deep Learning System for Recognizing and Recovering Contaminated Slider

Serial Numbers in Hard Disk Manufacturing Processes

จัดทำโดย

640910670 นาย ชัยทรภูมิ นพศรี

640911046 นางสาว พรมวนิช อภิรักษ์ตนกุล

เสนอ

อาจารย์ ดร.อรทัย วัชรากฤชกรณ์

รายงานนี้เป็นส่วนหนึ่งของวิชา 618390 สัมมนา

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์และระบบคอมพิวเตอร์ คณะวิศวกรรมศาสตร์และเทคโนโลยี

อุดสาหกรรม

ภาคการศึกษาที่ 1 ปีการศึกษา 2566

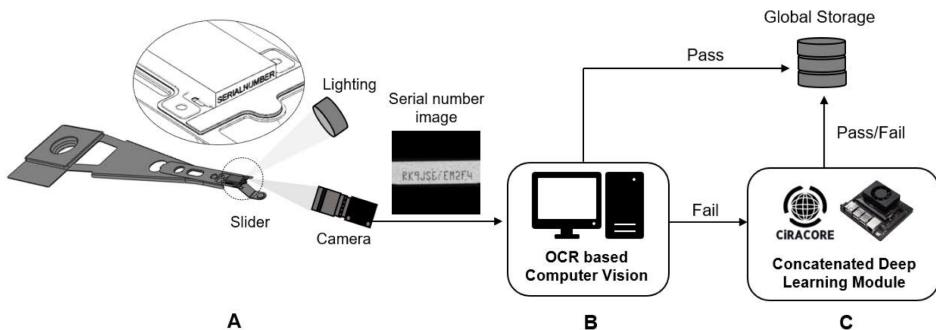
มหาวิทยาลัยศิลปากร พระราชวังสนามจันทร์

## Abstract

บทความนี้อธิบายระบบที่ใช้ในการตรวจหาข้อผิดพลาดในการพิมพ์และการระบุความผิดพลาดในสไลเดอร์ของฮาร์ดดิสก์ไดร์ฟ ซึ่งอาจส่งผลให้เกิดปัญหาในการติดตามสินค้าและการส่งให้กับผู้ใช้ที่ไม่ถูกต้อง ขั้นตอนที่ใช้ในงานวิจัยนี้มีการเสนอการใช้เทคนิคการเรียนรู้เชิงลึก (A Deep Learning) เพื่อระบุตัวตัวเลขบนสไลเดอร์จากรูปภาพที่ถ่ายจากกล้องดิจิทัล วิธีการของเราริเริ่มต้นด้วยขั้นตอนการประมวลผลภาพที่จัดการกับความแตกต่างในแสงและตำแหน่งการพิมพ์ และจากนั้นจึงนำไปสู่การตรวจหาตัวอักษรด้วยอัลกอริทึม You-Only-Look-Once (YOLO) V 4 และในท้ายที่สุดคือการจำแนกตัวอักษรสำหรับการจำแนกตัวอักษร เราได้ทดลองเปรียบเทียบเพื่อดูความแม่นยำและประสิทธิภาพของ Convolutional Neural Networks (CNN) 4 รูปแบบ คือ DarkNet-19, EfficientNet-B0, ResNet-50, และ DenseNet-201 การทดลองกับรูปถ่ายจำนวนเกือบ 15,000 รูปได้ผลลัพธ์ที่มีความถูกต้องมากกว่า 99% ใน 4 ระบบ CNN และพิสูจน์ความเป็นไปได้ของเครื่องข่าย EfficientNet-B0 ที่มีผลลัพธ์ดีกว่าผู้อ่านมนุษย์ที่มีความรู้ความสามารถสูง โดยมีอัตราการถูกคืนที่ดีที่สุด (98.4%) และเวลาการตอบสนองที่เร็วที่สุด (256.91 มิลลิวินาที)

## 1. บทนำ

สไลเดอร์หัว (Head Slider) เป็นเซ็นเซอร์แม่เหล็กที่ติดตั้งที่ปลายของแขนกันสั่น (Suspension Arm) ที่เรียกอีกอย่างว่าหัวกิมบล (Head Gimbal Assembly) ของฮาร์ดดิสก์ไดร์ฟ (Hard Disk Drive, HDD) ซึ่งมีหน้าที่ช่วยในการอ่านและเขียนข้อมูลบนพื้นผิวสื่อแม่เหล็กของฮาร์ดดิสก์ สไลเดอร์ถูกผลิตโดยใช้วาฟเฟอร์ที่ทำจากอลูминีียมและไทเทเนียม วาฟเฟอร์รูปร่างวงกลมจะถูกแบ่งเป็นชิ้นความหนา 700 ไมครอน และเชื่อมต่อกับชุดแขนกันสั่น (Suspension Assembly) กระบวนการผลิตทั้วไปรวมถึงการผลิตวาฟเฟอร์และติดตั้ง การตัดเป็นชิ้น การแบ่งแคว และการแยกหัวเป็นส่วนต่างๆ ในระหว่างกระบวนการประกอบสไลเดอร์ทางกล มักพบข้อผิดพลาดบางอย่าง เช่น ความสกปรกหรือรอยขีดข่วน ดังผลลัพธ์การประกอบของสไลเดอร์ทางกล ระบบการพิสูจน์อัตโนมัติ (Auto ID) ของสไลเดอร์เป็นสิ่งจำเป็นในการติดตามและระบุประวัติกระบวนการขององค์ประกอบ การตรวจสอบข้อผิดพลาดในกระบวนการผลิต และการเรียกคืนผลิตภัณฑ์ ระบบ Auto ID ใช้การพิมพ์หมายเลขลำดับด้วยเลเซอร์บนสไลเดอร์แต่ละชิ้น หมายเลขลำดับมีความยาว 12 ตัวอักษร ประกอบด้วยตัวเลข (0-9) และตัวอักษร (A-Z) โดยทั่วไป หมายเลขลำดับจะถูกบันทึกด้วยกล้องดิจิทัลและเก็บเป็นรูปภาพ (ดูในรูปที่ 1A)



รูปที่ 1. ระบบการกำหนดค่าของวิธีการที่เสนอสำหรับการตรวจสอบข้อผิดพลาดในหมายเลขลำดับบนสไลเดอร์ของ HDD

(A) การรับภาพ OCR

(B) การมองเห็นของคอมพิวเตอร์ของ OCR แบบอ่านตัวอักษรด้วยสายตา

(C) เทคนิคการเรียนรู้เชิงลึกที่ต่อรวม OCR

ในการผลิตอุตสาหกรรมขนาดใหญ่ ระบบการรู้จำตัวอักษรด้วยสายตา (OCR) ที่ใช้เทคนิคซอฟต์แวร์คอมพิวเตอร์เพื่ออ่านหมายเลขลำดับบนสไลเดอร์หัวที่ได้จากเครื่องคิดตั้งสไลเดอร์มากกว่าร้อยเครื่อง (ดูในรูปที่ 1B) มีอัตราการอ่านที่สามารถอ่านได้ทั้งหมดถึง 99.87% นี่เทียบเท่ากับการสูญเสียการผลิตรายปีอย่างน้อย 0.13% ที่เกิดจากสไลเดอร์ที่ไม่สามารถระบุตัวตนได้ เนื่องจากสกปรกหรือรอยขีดข่วนบนหมายเลขลำดับในระหว่างกระบวนการประกอบ เมื่อคิดคำนวณต้นทุนโดยเฉลี่ยของข้อผิดพลาดของผลิตภัณฑ์นี้ทำให้เกิดการสูญเสียมากกว่า 400,000 долลาร์

การเรียนรู้เครื่องจาก OCR กำลังเข้ามายึดงานหลายงาน โดย Huang และคณะเสนอเทคนิคสำหรับการระบุหมายเลขลำดับของบัตรธนาคาร โดยใช้คุณลักษณะ Gradient แบบปกติ-ร่วมกับแบบลักษณะสั้น-ระยะยาว (Long Short-Term Memory, LSTM) ที่มีความแม่นยำในการระบุสตริงตัวเลขอยู่ที่ 90.04% Jang และคณะนำเสนอกระบวนการแบ่งส่วนของตัวอักษรและการระบุหมายเลขลำดับธนาคารในก็อก (Banknote Serial Number, SNR) คุณภาพต่ำ ด้วยความแม่นยำในการระบุอยู่ที่ 99.85% Zhu และคณะเสนอกรอบการระบุปุ่มลิฟต์ที่เรียกว่า OCR-RCNN ซึ่งได้คะแนน F1 ที่มีค่าเท่ากับ 0.94, 1.00, และ 1.00 ในงานแยกส่วนการตรวจพบและความแม่นยำอยู่ที่ 79.6%, 96.5%, และ 96.4% ในงานระบุตัวอักษร Sun และคณะเสนอระบบ SuperOCR สำหรับการระบุตัวอักษร โดยไม่ต้องตรวจหาตำแหน่งของแต่ละตัวอักษร ซึ่งนำมาใช้ในงานระบุทะเบียนรถและระบุตัวอักษรของมิเตอร์น้ำ มีความแม่นยำรวมกว่า 98.7% และ 98.0% ตามลำดับ Laroca และคณะเสนอการนับอัตโนมัติ (ความแม่นยำ 100%) และการระบุ (อัตราการระบุ 99.7%) ของรถไฟด้วย OCR และการเรียนรู้เชิงลึก Kazmi และคณะใช้การเรียนรู้เชิงลึกที่ใช้ OCR สำหรับการตรวจ

ความถูกต้องของรหัสทางของ yan พาหนะที่เคลื่อนที่และการระบุข้อความด้วยการใช้กล้องถนน โดยมีความแม่นยำเฉลี่ยอยู่ที่ 86%

Caldeira และคณะเสนอการจำแนกตัวระบบการรู้จำตัวอักษรด้วยระบบ OCR สำหรับการระบุข้อผิดพลาดในการพิมพ์และการระบุข้อผิดพลาดในการระบุบนม้วนเหล็ก ของเครื่องรับสัญญาณช่วยการหาตำแหน่งที่มีความแม่นยำมากกว่า 98% Cakic และคณะใช้เครื่องมือ Tesseract OCR เพื่อรู้จำหมายเลขลำดับจากนักข่าวด้านในในการติดตามและติดตามขวดไวน์ โดยมีความแม่นยำอยู่ที่ 87.5% Gang และคณะสามารถหันโลกแล้วตัวระบบคอร์สต์เชิงลึกสำหรับการระบุตัวอักษรบนกระดาษจรป์รีนท์ มีความแม่นยำอยู่ที่ 94.7% Li และคณะนำเสนอกรอบเวิร์คของการเรียนรู้เชิงลึกสำหรับการตรวจสอบหมายเลขลำดับของสื่อหุ้มเซรามิก ที่มีคะแนน F-score ที่มากกว่าหรือเท่ากับ 95.6% และความแม่นยำเป็น 96.5% Kwon, Hyun และคณะเสนอใช้เทคนิคการเรียนรู้เชิงลึกเพื่อรักษาความปลอดภัยของระบบ โดยการรวมเอาเกณฑ์ซอฟต์แวร์บนคลาสซิไฟแอร์จำนวนหลายในการจำแนกหลายๆ ครั้งแทนการใช้คลาสซิไฟแอร์แบบเดียวคน อย่างไรก็ตาม ยังไม่มีรายงานกรมที่มีอยู่เกี่ยวกับการรู้จำตัวเลขลำดับที่อาศัย OCR บนสไลเดอร์หัวของ HDD

ในบทความนี้ เราเสนอเทคนิคการรู้จำแบบต่อรวมสำหรับการระบุหมายเลขลำดับบนสไลเดอร์หัวของ HDD ในบริบทของการผลิตในอุตสาหกรรม (ดูในรูปที่ 1C) เราเสนอให้ใช้เทคโนโลยีที่เสนอเพื่อปรับปรุงข้อมูลพร่องของระบบคอมพิวเตอร์ที่ใช้ OCR แบบดั้งเดิม เราใช้เทคนิคการเรียนรู้สองสถานะ ซึ่งรวมระบบตรวจจับวัตถุที่ใช้โนมูล YOLOv4 กับหนึ่งในโนมูลการจำแนกที่สืบย่างกือ DarkNet-19, EfficientNet-B0, Deep Residual Network (ResNet-50), และ DenseNet-201 เราได้ประเมินประสิทธิภาพของเทคนิคการเรียนรู้เชิงลึกที่ต่อรวมไว้กับอินเฟอร์เรนซ์ที่เหมาะสมกับการผลิตในอุตสาหกรรม โดยในที่สุดนี้เป็นครั้งแรกที่เทคนิคการเรียนรู้เชิงลึกถูกใช้ในการตรวจสอบหมายเลขลำดับของสไลเดอร์หัว HDD โดยอัตโนมัติ

## 2. การทบทวนเครือข่าย CNN

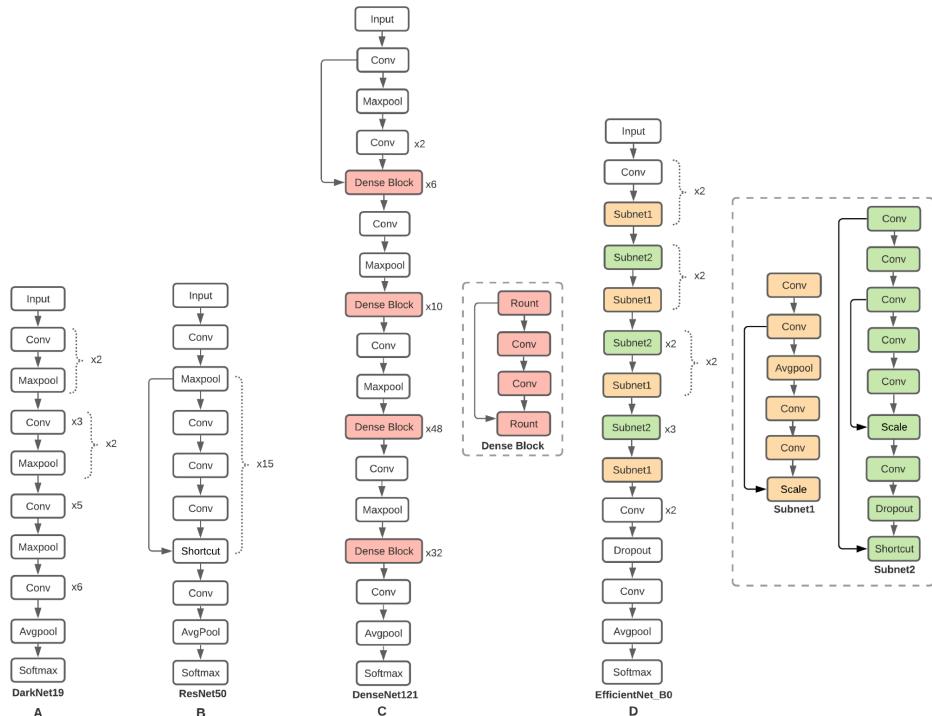
เครือข่ายประสาทคอนโวโลชัน (CNN) ได้รับการพัฒนาเป็นเทคนิคการเรียนรู้ของเครื่องที่สำคัญในการตรวจจับวัตถุเสมือนจริง. ตั้งแต่ LeNet-5 เวอร์ชันต้นแบบ, VGG-19, และ Highway Networks, ความคืบหน้าในอาร์คิวเร็คอมพิวเตอร์, ความเพียงพอของข้อมูลการฝึกฝนในปริมาณมาก, และความคืบหน้าในโครงสร้างของเครือข่าย CNN ได้ช่วยให้การฝึกเข้มข้น. เมื่อความลึกของเครือข่ายเพิ่มขึ้น, ความผิดพลาดในการฝึกฝนเพิ่มขึ้น จนเกิดปัญหาความถดถอย ซึ่งทำให้ความแม่นยำเริ่มเตี้ยและลดลงอย่างรวดเร็ว. โครงข่ายที่เหลือ (ResNets) เป็นโครงข่ายซึ่งคราสที่มีการเชื่อมต่อแบบ short cut ที่ทำการทำภาพเดียวกันโดยการ

เพิ่มผลลัพธ์ของพากเบากับผลลัพธ์ของชั้นที่รวมเข้าด้วยกันเพื่อแก้ปัญหาความถดถอย ResNets จ่ายต่อการปรับใช้และบรรลุความแม่นยำที่ดีขึ้น, ดังนั้น โครงข่ายปกติมีความผิดพลาดในการฝึกฝนมากขึ้นเมื่อความลึกของเครือข่ายเพิ่มขึ้น. เราใช้ ResNets-50 (50 ชั้นคอนโวลูชัน, 1 maxpooling, 1 averagepool, และ การทำ Shortcut 16 ครั้ง) ซึ่งมีความแม่นยำสูงสุด(top-1) 76.0% และความแม่นยำสูงสุด(top-5) 93.0% ในการตรวจสอบ ImageNet

Deep CNN Darknet-19 ลูกเสนอเป็นโมเดลการจำแนกชั้น Darknet-19, ซึ่งมี 19 ชั้นคอนโวลูชัน และ 5 ชั้น maxpooling (ดูในรูปที่ 2B) และต้องใช้แค่ 5.58 พันล้านครั้ง เพื่อประมาณภาพ พร้อมที่รับรู้ความแม่นยำสูงสุด (top-1) 72.9% และความแม่นยำสูงสุด(top-5) 91.2% ในการตรวจสอบ ImageNet Darknet-19 เป็นรูปแบบที่ไม่ซับซ้อนมาก ดังนั้นมันเร็วและมีความแม่นยำมากขึ้น โครงข่าย CNN ที่มีการเชื่อมต่อที่แคบระหว่างชั้นใกล้ผลลัพธ์สามารถถูกขึ้น, มีความแม่นยำมากขึ้นและมีประสิทธิภาพในการฝึกฝนมากขึ้น

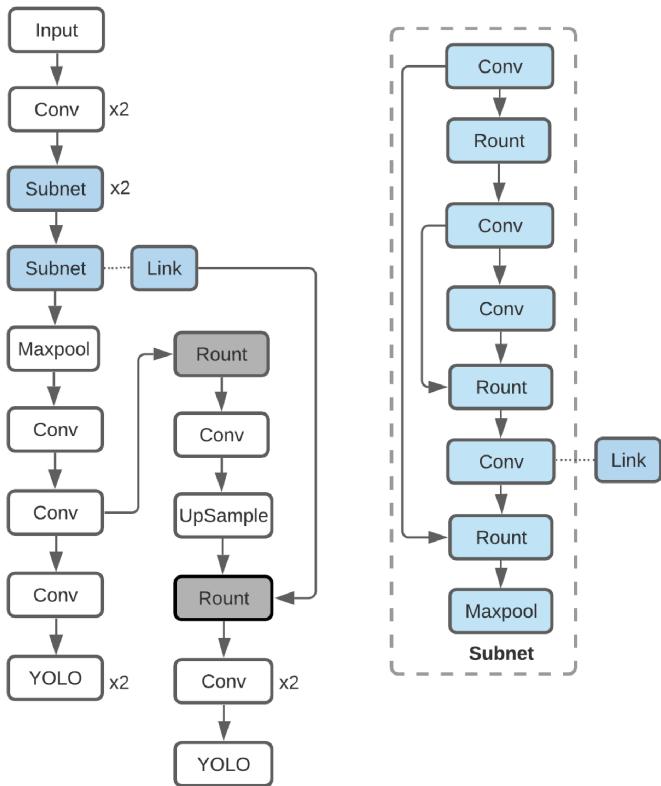
โครงข่าย Dense Convolutional Network (DenseNet) เชื่อมทุกชั้นกับทุกชั้นอื่น ๆ ในแบบ feedforward (ดูในรูปที่ 2C) เพื่อเพิ่มปัญหา gradient ที่หายไป เพิ่มการกระจายคุณสมบัติ ใช้คุณสมบัติอย่างง่าย และลดจำนวนพารามิเตอร์อย่างมีนัยยะ เราใช้ DenseNet-201 ในงานวิจัยนี้ ซึ่งมีชั้นคอนโวลูชัน 179 ชั้น, maxpool 4 ชั้น และ averagepool 1 ชั้น พร้อมที่มีพารามิเตอร์ 20 ล้านพารามิเตอร์

EfficientNet-B0 เป็นวิธีการปรับขนาดใหม่ที่ปรับขนาดทุกมิติ depth/width/resolution อย่างเท่าเทียม (ดูในรูปที่ 2D) เราเลือกใช้ EfficientNet-B0 สำหรับการฝึกการจำแนกภาพ เพราะมีแค่ 5.3 ล้านพารามิเตอร์และ 0.39 พันล้าน FLOPs, ในขณะที่ Resnet-50 มีพารามิเตอร์ 26 ล้านและ FLOPs 4.1 พันล้าน EfficientNet-B0 มีความแม่นยำสูงสุดที่ 77.1% และความแม่นยำที่สูงสุด 93.3%



รุปที่ 2. โครงข่าย CNN ที่ใช้ในงานวิจัยนี้ DarkNet-19 (A), ResNet-50 (B), DenseNet-121 (C), และ EfficientNet-B0 (D)

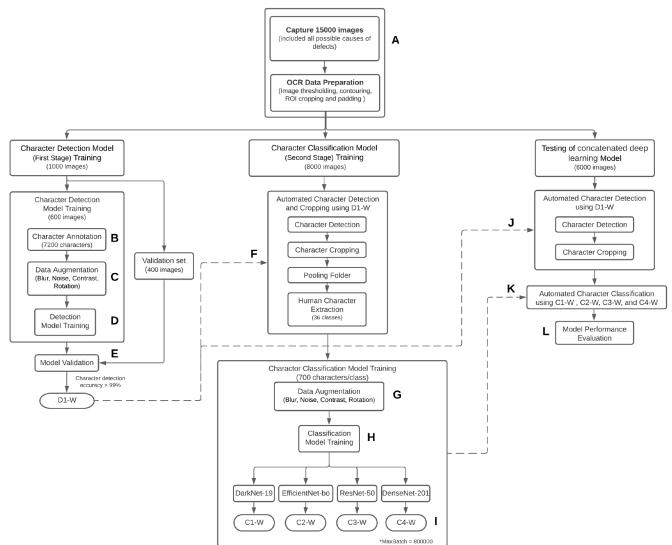
### 3. ไม่เคลียร์การเรียงต่อແລ້ວ



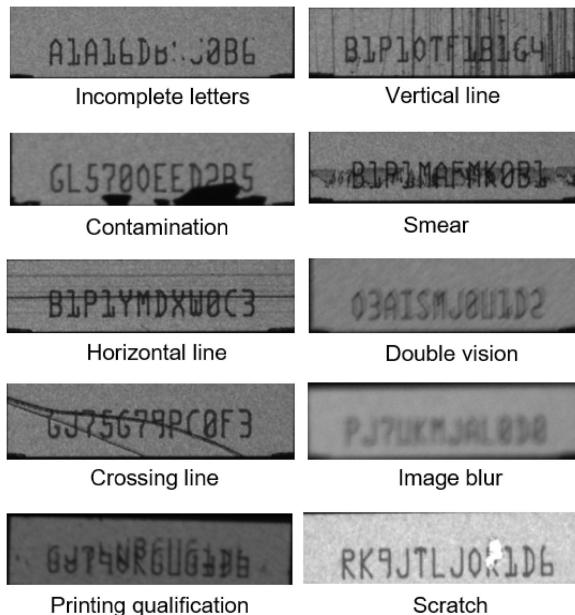
รูปที่ 3. YOLO v4 ใช้สำหรับการตรวจจับวัตถุ

### 3.1. การประมวลผลภาพที่ถูกบันทึกไว้

รูปภาพที่มีข้อมูลพร่อง (15,000 รูปภาพ) ของหมายเลขซีรีลของตัวไลเดอร์ทั่วไปมักถูกปฏิเสธตามมาตรฐานการใช้งานของ OCR ในอุตสาหกรรม โดยทั่วไปรูปภาพเช่นนี้ถูกถ่ายโดยกล้องดิจิทัลอุตสาหกรรม (Point Grey 1/3" CCD; โมโน;  $1288 \times 964$ ; 31 ภาพต่อวินาที) จากเครื่องอ่าน/พิมพ์หัว 100 เครื่อง (รวม 150 รูปภาพต่อเครื่อง) โดยทั่วไปรูปภาพที่มีข้อมูลพร่องถูกระบุด้วยมือเพื่อคัดเลือกตัวไลเดอร์ที่อาจเป็นไปได้จาก การปฏิเสธ รูปที่ 5 แสดงแหล่งที่มาที่เป็นข้อมูลพร่องทั้งหมดในกระบวนการผลิตอุตสาหกรรม (ป้ายกำกับ A ในรูปที่ 4) รูปภาพที่มีข้อมูลพร่องมีความละเอียดของ  $1280 \times 960$  พิกเซล รูปภาพถูกแบ่งเป็นสามหมวดหมู่: 9000 รูปสำหรับชุดข้อมูลการฝึก (1000 สำหรับขั้นตอนการตรวจจับตัวอักษรและ 8000 สำหรับการจำแนกตัวอักษร), 6000 สำหรับการทดสอบโมเดล (ดูรูปที่ 4)



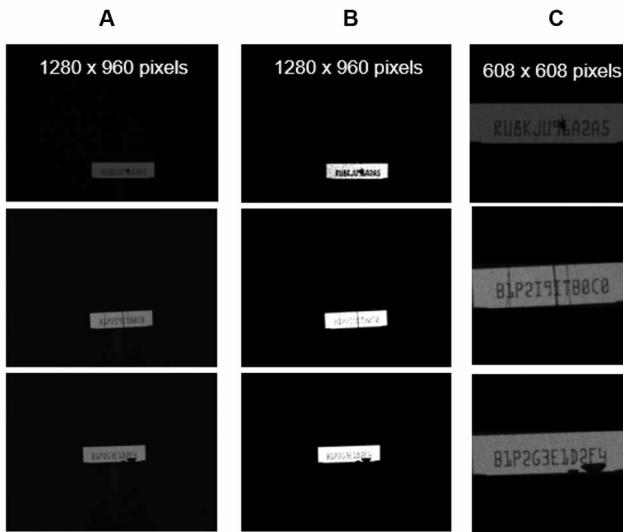
รูปที่ 4. แผนภาพแผนผังของโมเดลการเรียนรู้เชิงลึกที่รวมตัว



รูปที่ 5. ประเภทของข้อกพร่องบนภาพหมายเลขซีเรียลที่เกิดขึ้นในสีนักพัฒนา

ในขั้นตอนการจับภาพในสภาพแวดล้อมอุตสาหกรรมทั่วไป, มีสองด้านแปรที่ไม่สามารถควบคุมได้ในการทดลอง (ดูในรูปที่ 6A) คือ ความแตกต่างในตำแหน่งของหมายเลขซีเรียลในภาพที่ถูกจับและความแตกต่างในแสงในสภาพต่าง ๆ ที่ส่งผลให้ความคมชัดในภาพที่ถูกจับถ่ายแตกต่างกัน เพื่อกำจัดด้านแปรเหล่านี้ ใช้วิธีการประมวลภาพมาตรฐานที่ใช้ opencv-python 4.4.0.44 เพื่อแยกเฉพาะพื้นที่ที่มีหมายเลขซีเรียลและปรับความคมชัดของภาพให้เท่ากัน ในกระบวนการนี้ใช้วิธี Otsu สำหรับการทำภาพให้เป็นรูปภาพขาวดำเพื่อสร้างภาพที่เป็นขาวดำ (ดูในรูปที่ 6B) ใช้เทคนิคในการสร้างขอบเพื่อบกรอบของพื้นที่ที่

น่าสนใจ (ROI) คือพื้นหลังของหมายเลขซีเรียล สุดท้าย ตามที่แสดงในรูปที่ 6C พื้นที่ที่มีหมายเลขซีเรียลถูกวางไว้ตรงกลางของรูปภาพ และทุกภาพถูกกำหนดให้มีความสูงเท่ากัน (608 พิกเซล) และความกว้างเท่ากัน (608 พิกเซล)



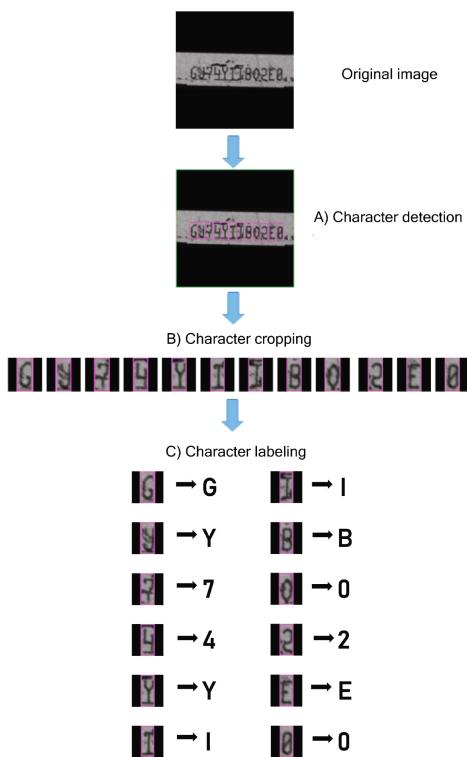
รูปที่ 6. การเติมข้อมูล OCR (Optical Character Recognition)

- (A) ภาพถ่ายเดิมที่มีความแตกต่างในความคมชัดของภาพและการเลื่อนพื้นที่ที่สนใจ (ROI)
- (B) การกำหนดค่าความคมชัดและการตรวจจับขอบ
- (C) การตัดส่วนพื้นที่ที่สนใจ (ROI cropping) และการเติมพื้นที่ (padding)

### 3.2 โมเดลขั้นแรก : โมเดลการตรวจจับตัวอักษร

จุดประสงค์ของในการสร้างโมเดลเพื่อตรวจจับตัวอักษรในภาพหมายเลขซีเรียลคือการเติมชุดข้อมูลและฝึกโมเดลเพื่อตรวจจับพื้นที่ตัวอักษรในรูปภาพหมายเลขซีเรียลที่ผ่านการประมวลภาพตามขั้นตอนก่อนหน้า การเติมชุดข้อมูลเริ่มต้นด้วยการประมวลข้อมูลและการเพิ่มข้อมูลในชุดข้อมูล 600 รูปถูกเลือกแบบสุ่มจากกลุ่มข้อมูลแรก (label B ในรูปที่ 4) การประมวลข้อมูลถูกดำเนินการโดยผู้เชี่ยวชาญ ซึ่งผลในรูปภาพตัวอักษรที่ถูกเน้นไว้ 7200 รูปภาพตัวอักษรแต่ละอันถูกคาดคะเนเพื่อใช้ก่อร่องขอบเพื่อแสดงตำแหน่งแต่ละตัวอักษร (คลาส A-Z และ 1-9) บนภาพ (12 ตัวอักษร/ภาพ) (ขนาด  $35 \times 75$  พิกเซล) จุดกึ่งกลางของกล่องขอบของอักษรจะอยู่กลางใจของแต่ละตัวอักษรและรวมถึงพื้นที่ขอบของตัวอักษร การเพิ่มข้อมูลถูกดำเนินการโดยพิจารณาความแตกต่างที่เป็นไปได้ (เช่นการหมุน, การเบลอ, ความคมชัด, และรับ光) จากภาพเดิม (label C ในรูปที่ 4) ภาพเดิมถูกหมุนจาก  $-180^\circ$  ถึง  $180^\circ$  ( $90^\circ$  ต่อขั้น); การเปลี่ยนค่าเบลอของ การหมุนกัสเซียร์ คือค่าส่วนเบี่ยงเบนมาตรฐานของ 3, 6, 9, 12, 15, 18, 21, 24, 27, และ 30; การปรับความคมชัดโดยการคูณค่าพิกเซลทั้งหมดด้วย 0.8, 1.0, 1.2, 1.4, 1.6, 1.8, 2.0, และ 2.2; และการเพิ่มการกระจาย

ของระบบกัสเซี่ยนด้วยค่าส่วนเบี่ยงเบนมาตรฐานของ 10 ผลลัพธ์คือ มีตัวอักษรที่เพิ่มขึ้นถึง 1,728,000 ตัวอักษร นอกจานี้ ตัวอักษรที่เพิ่มข้อมูลลูกน้ำเข้าไปในแพลตฟอร์มการพัฒนาโมเดลเรียนรู้ลึกภายใน (CiRA CORE) เพื่อฝึกโมเดลตรวจจับตัวอักษรที่ใช้โมเดล YOLO v4 (label D ในรูปที่ 4) ในการฝึกฝน มีการใช้เซิร์ฟเวอร์ CiRA CORE ที่มีซีพียู Intel Xeon® silver 4210 2@2.2GHzx40 และหน่วยความจำ RAM 125.6 GB รวมเวลาการคำนวณทั้งหมดคือ 36 ชั่วโมง 15 นาที และ 52 วินาที พารามิเตอร์ที่ระบุว่า D1-weighted (D1-W) ได้รับหลังจากการฝึกโมเดลเพื่อเลือกโมเดลการตรวจจับตัวอักษรที่ให้ประสิทธิภาพที่ดีที่สุด ในที่สุด โมเดลการตรวจจับตัวอักษรลูกทดสอบกับข้อมูลการตรวจจับ 400 รูปภาพ (label E ในรูปที่ 4) อัตราการตรวจจับที่ได้คือ 100% และพื้นที่ที่ตรวจจับของแต่ละตัวอักษรสามารถครอบคลุมกล่องของทั้งหมด (รูปที่ 7A)



รูปที่ 7. ตัวอย่างของกระบวนการตรวจสอบขั้นตัวอักษร, การตัดรูปและตั้งชื่อในกระบวนการ

### 3.3. ขั้นตอนที่สองของไมโครเดลการจัดหมวดหมู่ตัวอักษร

### 3.3.1 การเตรียมข้อมูลชุดข้อมูล

ขั้นตอนที่สองของ โนมเดลการจัดหมวดหมู่ตัวอักษร ใช้กลุ่มข้อมูลที่ได้จากการปูภาคข้อมูลพร่องที่ผ่านการประเมินภาพ 8,000 รูปภาพเพื่อพัฒนา โนมเดลการจัดหมวดหมู่ตัวอักษร โนมเดลการตรวจจับตัวอักษรที่อธิบายในส่วน 3.2 ถูกใช้พร้อมพารามิเตอร์ D1-weighted ที่ถูกปรับให้เหมาะสม ชุดข้อมูลที่มี 96,000 ตัวอักษรถูกได้รับโดยการตัดภาพที่ตรวจจับมาจากการพื้นที่ภาพตัวอักษร (รูปที่ 7B) หลังจากนั้น ภาพตัวอักษรที่ถูก

อกตัดเป็นส่วนๆที่ถูกป้ายข้อด้วยมือ (รูปที่ 7C) และจัดเรียงเป็นกลุ่มตามคลาส 36 กลุ่ม (คลาสตัวเลข 0-9 และคลาสตัวอักษร A-Z). แต่ละคลาสมีรวมตัวอักษร 700 ตัว ชุดข้อมูลได้รับการรับรองจากผู้สังเกต ประสบการณ์สูง มีตัวอักษรรวม 25,200 ตัวที่ใช้ในการฝึกโมเดลการจัดหมวดหมู่ในขั้นตอนถัดไป หลังจากนั้น กระบวนการเพิ่มข้อมูลถูกดำเนินการโดยพิจารณาความแตกต่างที่เป็นไปได้จากภาพเดิม (label G ในรูปที่ 4) เพื่อให้ได้ชุดข้อมูลตัวอักษร 630,000 ตัว (17,500 ตัวต่อคลาส) เราได้เปลี่ยนค่าของตัวกรอง Gaussian blur โดยการเปลี่ยนค่าของค่ามาตรฐานเป็น 3, 6, 9, 12, และ 15, ปรับความคมชัดโดยการคูณค่าพิกเซลทั้งหมดด้วย 1.0, 1.3, 1.6, 1.9, และ 2.1

### 3.3.2. การฝึกโมเดล

ในการฝึกโมเดลการจัดหมวดหมู่ เรานำเข้าชุดข้อมูลการฝึกเพิ่มเติมของตัวอักษร 36 คลาส 630,000 ตัวอักษรลงในกรอบการเรียนรู้ภายในของเรา (แพลตฟอร์ม CiRA CORE) (แท็ก H ในรูปที่ 4) สำหรับการประเมินประสิทธิภาพเราใช้โมเดล CNN 4 แบบ (คือ DarkNet-19, EfficientNet-B0, ResNet-50, และ DenseNet-201) โดยใช้ชุดข้อมูลเดียวกัน พารามิเตอร์การฝึกคือ ขนาดของชุดข้อมูล 64, การมอเมนตัม 0.9, ค่าความลดลง 0.0005, จำนวนสูงสุดของแบบที่ 800,000, และอัตราการเรียนรู้ 0.1 (ยกเว้นสำหรับ EfficientNet-B0 ซึ่งมีอัตราการเรียนรู้เท่ากับ 0.256) การฝึกโมเดลการจัดหมวดหมู่นี้ใช้เซิร์ฟเวอร์ CiRA CORE (Intel Xeon®silver 4210 CPU2@.2GHzx40, 125.6 GB RAM)

ตาราง 1 สรุปเวลาการฝึกของโมเดล CNN 4 รายบนชุดข้อมูลการฝึกเดียวกัน และการตั้งค่าพารามิเตอร์และฮาร์ดแวร์ที่ใช้ในการคำนวณ โมเดล 4 แบบ C1-W, C2-W, C3-W และ C4-W ได้รับจาก DarkNet-19, EfficientNet-B0, ResNet-50, และ DenseNet-201 ตามลำดับ และถูกใช้ในการทดสอบประสิทธิภาพในระหว่างกระบวนการทดสอบโมเดล (แท็ก I ในรูปที่ 4)

CNN Networks	Training Time		
	Hours	Minutes	Seconds
DarkNet-19	49	58	51
EfficientNet-B0	133	13	4
ResNet-50	82	57	45
DenseNet-201	128	3	4

ตาราง 1. เวลาการฝึกฝนของสี่รุ่นของ CNN networks.

### 3.4. การประเมินประสิทธิภาพของโมเดล

ทางทีมวิจัยได้ดำเนินการทดสอบเพื่อประเมินประสิทธิภาพของวิธีการที่เสนอสำหรับการระบุตัวละครอัตโนมัติและการจัดกลุ่มหมายเหตุเรียลจากภาพข้อผิดปกติ โมเดลที่เสนอได้ถูกนำไปทดสอบโดยใช้รูปภาพข้อผิดปกติ 6000 รูป เพื่อระบุตัวอักษรและครอบบริเวณ โดยใช้โมเดล D1-weighted (เครื่องหมาย J ในรูปที่ 4) จากนั้นได้ใช้น้ำหนักที่มีคุณสมบัติมากกว่าสี่ค่า (C1-W, C2-W, C3-W, และ C4-W) สำหรับแต่ละโมเดลเพื่อการจัดกลุ่มตัวอักษรโดยอัตโนมัติ (เครื่องหมาย K ในรูปที่ 4) การทดสอบนี้ดำเนินการบนเซิร์ฟเวอร์ที่ใช้หน่วยประมวลผล Intel Xeon® Silver 4210 CPU2@2.2GHzx40 และหน่วยความจำ 125.6 GB RAM ชุดข้อมูลทดสอบสามารถให้ครบครันได้ที่

<https://github.com/Chousak/OCR-Data-Validation-Noise-.git>

เพื่อวัดประสิทธิภาพในการจัดกลุ่มตัวอักษร (เครื่องหมาย L ในรูปที่ 4) มีการคำนวณดังนี้ประสิทธิภาพของ Precision, Recall, และ Accuracy ดังนี้

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

โดยที่ TP คือ true positive, FP คือ false positive, และ FN คือ false negative สามารถนิยาม Accuracy ได้ดังนี้โดยคำนึงถึงความแม่นยำและความจำครอ卜:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{TN} + \text{FN})$$

โดยที่ TN คือ true negative ความแม่นยำเป็นตัววัดความสามารถของโมเดลที่ดีที่สุดจากมุมของการผลิต HDD แต่การประเมินประสิทธิภาพสำหรับข้อมูลที่ไม่สมดุล เช่น ในการศึกษานี้สามารถที่จะประเมินได้มีประสิทธิภาพมากขึ้นเมื่อใช้ F1 score ที่เป็นค่าเฉลี่ยแพร์โมนิกของอัตราความแม่นยำและความจำครอ卜 และสามารถนิยามได้ดังนี้:

$$\text{F1 score} = (2 * \text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

สำคัญที่จะมีอย่างน้อย 99% ความแม่นยำ ความแม่นยำในการจำครอ卜 ความจำครอ卜 และ F1 score สำหรับการใช้งานแบบเบรล ไทน์บนสายการผลิต เพื่อประเมินความเป็นไปได้ของวิธีการที่เสนอ ความเร็วในการทดสอบความสามารถในการสรุปที่มีประสิทธิภาพควรใช้เวลาอย่างน้อย 300 มิลลิวินาที (ms) ซึ่งเป็นเวลาในการพยากรณ์ของ OCR-based computer vision ได้ทำการเปรียบเทียบเวลาในการพยากรณ์ (ต่อรูปภาพ) ในสี โมเดล CNN ที่แตกต่างกันคือ DarkNet-19, EfficientNet-B0, ResNet-50, และ DenseNet-201

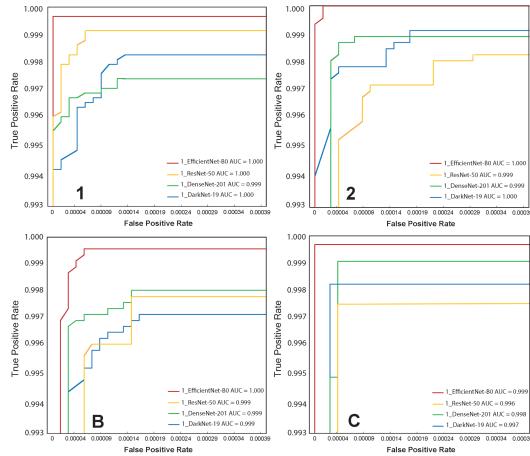
#### 4. ผลการประเมินประสิทธิภาพ

ตาราง 2 แสดงผลการประเมินประสิทธิภาพสำหรับโมเดล CNN สี่รุ่นที่แตกต่างกัน จากตาราง 2 แสดงว่าโดยใช้เงื่อนไขเดียวกัน วิธีการทั้งหมดที่เสนอให้ผลการทดสอบที่ดีที่สุด โดยมีความแม่นยำ ความแม่นยำในการจำลอง ความแม่นยำในการสืบค้น และค่า F1 มากกว่า 99% ทั้งสี่โมเดล CNN ได้ความแม่นยามากกว่า 99.90% (ระหว่าง 99.98% ถึง 99.99%) โดยที่ EfficientNet-B0 ได้คะแนน F1 สูงที่สุด (99.96%)

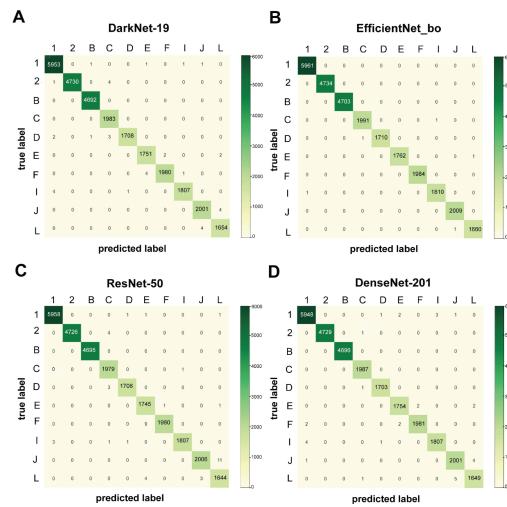
CNN Networks	Accuracy	Precision	Recall	F1 Score
DarkNet-19	99.98	99.66	99.73	99.69
EfficientNet-B0	99.99	99.96	99.96	99.96
ResNet-50	99.98	99.61	99.71	99.66
DenseNet-201	99.98	99.70	99.75	99.72

ตาราง 2. ตัวชี้วัดประสิทธิภาพการประเมินของวิธีการที่เสนอเปรียบเทียบระหว่างสี่รุ่นของ CNN networks.

การประเมินผลการทำงานของ Receiver Operating Characteristic (ROC) ถูกดำเนินการเพื่อประเมินความแม่นยำของวิธีที่เสนอในกลุ่มสีกลุ่มสูงสุด (คือ 1, 2, B, และ C) ซึ่งถือว่าเป็นตัวคละครที่มีข้อผิดพลาดในกระบวนการผลิตมากที่สุด (คุณปที่ 8) จากตารางที่ 8 ค่าพื้นที่ใต้เส้น (AUC) สำหรับโมเดล CNN ทุกรุ่นมีค่ามากกว่า 0.99 ในทุกกลุ่มสีกลุ่ม แต่โมเดล EfficientNet-B0 มีค่า AUC สูงสุด (ระหว่าง 0.999 ถึง 1.000) ในทุกกลุ่มสีกลุ่ม (คุณลักษณะในรูปที่ 8) การนับข้อผิดพลาดในการระบุจากกลุ่มตัวคละคร 36 กลุ่มถูกนับและเรียงลำดับเพื่อประเมินประสิทธิภาพในการจัดหมายของโมเดล สร้างเมทริกซ์ความสัมสัปดาห์เพื่อกลุ่มสูงสุด 10 กลุ่ม (คือ 1, 2, B, C, D, E, F, I, J, และ L) ที่มีการรับรู้ข้อผิดพลาดมากที่สุด คุณปที่ 9 แสดงแผนผลการสัมสัปดาห์สำหรับแต่ละโมเดล ประกอบด้วยจำนวนตัวคละครที่ถูกทำนายในแต่ละกลุ่มที่ถูกเปรียบเทียบในโมเดล CNN สี่รุ่น พบว่า EfficientNet-B0 มีประสิทธิภาพในการจัดหมายที่ดีที่สุด พบข้อบกพร่องทั่วไปในการระบุตัวคละคร คือ ResNet-50 (11 ครั้งที่ขาดหาย), DenseNet-121 (4 ครั้งที่ขาดหาย), และ DarkNet-19 (4 ครั้งที่ขาดหาย), โดยตัวคละคร "J" ถูกสะกดออกเป็น "L" หลายครั้ง ในขณะที่โมเดล EfficientNet-B0 ไม่มีข้อบกพร่องเช่นนี้ แม้ว่าโมเดล EfficientNet-B0 ให้ประสิทธิภาพที่ดีที่สุด แต่ก็ต้องใช้เวลาในการฝึกฝนมากที่สุด (มากกว่า 133 ชั่วโมง) และมีค่าใช้จ่ายที่แพงขึ้นประมาณ 2.66 เท่าของโมเดล DarkNet-19 (มากกว่า 49 ชั่วโมง) (คุณปที่ 1)



รูปที่ 8. กราฟ ROC ของคลาสนาเสนอข้อมูลเพิ่มเติมของคลาสชั้นนำ 4 จาก 4 CNN networks



รูปที่ 9. ตารางความสับสนของการจำแนกตัวอักษรของคลาส 10 อันดับจาก 4 รุ่นของ CNN networks: DarkNet-19 (A), EfficientNet-B0 (B), ResNet-50 (C), และ DenseNet-201 (D)

## 5. การอภิปราย

### 5.1. การนำไปใช้ในการผลิตอุตสาหกรรม

เราได้นำระบบนี้มาใช้โดยใช้แนวคิดของการเรียกใช้บริการผ่านอินเทอร์เน็ตแอปพลิเคชัน (API)

ผ่านโครงสร้างคลาวด์เพื่อช่วยในการถูกคุ้นชึ้นข้อมูลของตำแหน่งเดียวจากระบบคอมพิวเตอร์วิชันการมองหาด้วย OCR เพื่อลดต้นทุนหารดแวร์คอมพิวเตอร์ที่จำเป็นในกระบวนการผลิต ในกระบวนการที่เสนอเข้ามาใช้ในกระบวนการจัดหมายหมายเลขซีเรียลจากภูมิภาคที่มีข้อเสียในส่วนผลิต ต้องพิจารณาต้นทุนในการดำเนินการ เช่น เวลาตอบสนอง (คือ ปริมาณเวลาในกระบวนการคำนวณและการลีส์สาร) และหารดแวร์คอมพิวเตอร์ที่มีประสิทธิภาพสูง อย่างไรก็ตาม เวลาในการสื่อสารระหว่างระบบคอมพิวเตอร์วิชันการมองหาด้วย OCR ปกติและเชิงรุกอาจมีอัตราลดลงมากต่อเวลาตอบสนอง อย่างไรก็ตาม, วิธีการที่เสนอ

สามารถแทนที่ระบบการมองหาด้วย OCR แบบดั้งเดิมเมื่อต้นทุนสาร์ดแวร์ลดลงและการลงทะเบียนนั้นคุ้มค่า ในกระบวนการ OCR ส่วนใหญ่จะใช้การวัดเช็คชั้มเพื่อให้แน่ใจว่าการอ่านถูกต้อง ในงานวิจัยนี้ การคำนวณ เช็คชั้มถูกนำมาใช้เพื่อตรวจสอบผลลัพธ์ของโมเดลที่เสนอ เราสามารถกำหนดค่าเช็คชั้มของหมายเลขซีเรียลได้ดังนี้:

$$\text{Checksum} = ((8 * \text{Checksum}) + (\text{ค่า ASCII} - 32)) \% 59 \quad (\text{สมการ 5})$$

ต้องตรวจสอบหมายเลขซีเรียล (12 ตัวอักษร) ที่ทำนายด้วยวิธีการที่เสนอ ในกรณีที่ค่าเช็คชั้มไม่เท่ากับสูญเสียแสดงข้อผิดพลาดในการอ่าน ตารางที่ 3 แสดงตัวอย่างการคำนวณเช็คชั้ม

X

**Table 3. Checksum calculation of the serial number "FJ72Q4A9P2F4".**

Serial	ASCII	Checksum Equation	Checksum
F	70	$((8 \times 0) + (70 - 32)) \% 59$	38
J	74	$((8 \times 38) + (74 - 32)) \% 59$	51
7	55	$((8 \times 51) + (55 - 32)) \% 59$	18
2	50	$((8 \times 18) + (50 - 32)) \% 59$	44
Q	81	$((8 \times 44) + (81 - 32)) \% 59$	47
4	52	$((8 \times 47) + (52 - 32)) \% 59$	42
A	65	$((8 \times 42) + (65 - 32)) \% 59$	15
9	57	$((8 \times 15) + (57 - 32)) \% 59$	27
P	80	$((8 \times 27) + (80 - 32)) \% 59$	28
2	50	$((8 \times 28) + (50 - 32)) \% 59$	6
F	70	$((8 \times 6) + (70 - 32)) \% 59$	27
4	52	$((8 \times 27) + (52 - 32)) \% 59$	0
Final result = "0"			

ตาราง 3. การคำนวณเช็คชั้มของหมายเลขซีเรียล "FJ72Q4A9P2F4"

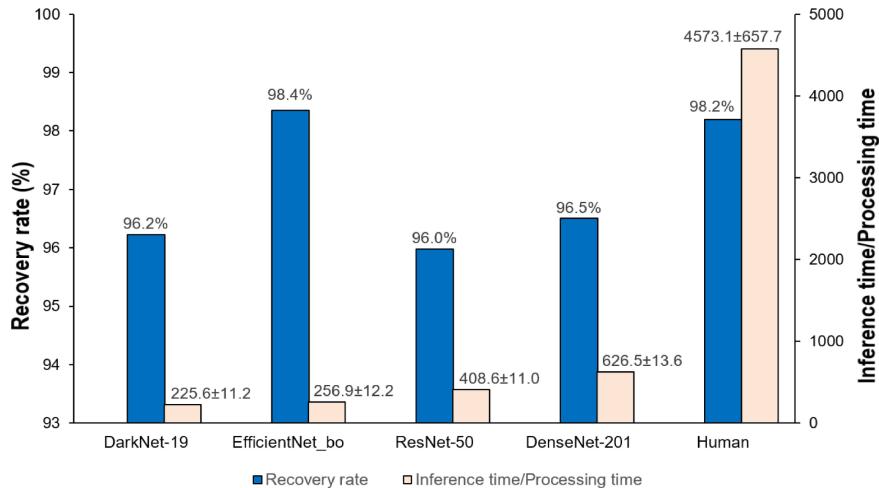
เพื่อประเมินประสิทธิภาพของการที่เสนอในการระบุภาพข้อเสียของหมายเลขซีเรียล ทางผู้วิจัยได้รับรูปภาพจำนวน 5000 รูปจากเครื่องอ่าน/เขียนหัว 100 เครื่องบนเส้นผลิต (50 รูป/เครื่อง) อัตราการถูกคืนถูกประมาณดังนี้:

$$\text{อัตราการถูกคืน} = \frac{\text{จำนวนรูปภาพที่ระบุได้}}{\text{จำนวนรูปภาพที่ระบุได้}} \times 100\% \quad (\text{สมการ 6})$$

จำนวนรูปภาพทั้งหมดที่ถูกบันทึกรวมทั้งรูปภาพที่ระบุได้และรูปภาพที่ไม่ระบุ อัตราการถูกคืนของโมเดล CNN ทั้งคู่ร่วมและการอ่านข้อมูลโดยมนุษย์ถูกเปรียบเทียบกับเวลาการอนุรักษ์ ดังแสดงในรูปที่ 10

Darknet-19 และ EfficientNet-80 มีเวลาการอนุรักษ์เฉลี่ยที่เปรียบเทียบได้ดีกว่า 300 มิลลิวินาที ดังที่แสดงในรูปที่ 10 ด้านการอนุรักษ์เวลาทุกโมเดลมีผลการดำเนินงานที่ดีกว่ามนุษย์อย่างไรก็ตาม หากเราให้ลำดับความสำคัญกับอัตราการถูกคืน EfficientNet-B0 (98.4%) จะเป็นโมเดลที่ดีกว่าการอ่านข้อมูลโดยมนุษย์

(98.2%) และมีเวลาการอ่านรักน์ที่เร็วกว่า น่าอัศจรรย์ที่โมเดล EfficientNet-B0 สามารถแยกแยะภาพข้อเสียงของหมายเลขซึ่งโมเดลที่มนุษย์ไม่สามารถตรวจสอบได้ รูปที่ 11 แสดงตัวอย่างรูปภาพหมายเลขซึ่งโมเดลที่มนุษย์ไม่สามารถระบุได้ ตามข้อมูลที่ได้รับเข้ามา



รูปที่ 10. อัตราการถูกคืน (%) และเวลาการตีความ (วินาที) ของสิ่งที่รุ่นของ CNN networks และการอ่านของมนุษย์

Database	B1P1LTKSP0H1	06GI9MDQB0A4	PL5KC3BMD1H0	BV91K086F2G5
Captured images				
OCR-based CDL (EfficientNet_B0)				
Human reading	B1P1LT?SP0H1	06GI9M?QB0A4	PL5KC3BMD1??	BV91??86F2G5

รูปที่ 11. ตัวอย่างการจำแนกที่ดีขึ้นของ EfficientNet-B0 ที่เหนือกว่าการอ่านของมนุษย์

เพื่อให้แน่ใจว่า โมเดลของเรา มีความสามารถทั่วไป เราได้ทำการทดสอบแบบ Cross-validation ที่มีห้ารอบเพื่อตรวจสอบว่า โมเดลที่เราเลือกสามารถให้ความแม่นยำโดยไม่มีการคาดเดาผิด การทำ Cross-validation ตามความสำนึกของเรารายลักษณะการเรียนรู้เกินไป เนื่องจาก Cross-validation สามารถช่วยลดข้อผิดพลาดที่ถูกตรวจสอบในการสร้าง โมเดลที่ดีที่สุด ทำให้ค่า parametric ทางสถิติไม่แตกต่างกันระหว่างการทดลองใด ๆ (ตาราง 4) นอกจากนี้เพื่อยืนยันความคงทนของ โมเดลที่เสนอ เราได้ประเมิน โมเดลด้วยชุดข้อมูลหมายเลขซึ่งโมเดลที่มีการเพิ่มเสียงแยกแยะ (Adversarial Noise) ค่า Adversarial Noise Scale (ANS) ที่สร้างขึ้นได้ถูกปรับจาก 0.01 ถึง 0.5 ประสิทธิภาพของ โมเดลที่เสนอด้วยการตั้งค่า ANS ต่าง ๆ และในตาราง 5 ความสามารถในการทำงานของ โมเดลที่เสนอของความสามารถใหม่อ่อนคิดโดยไม่มีเสียงใด ๆ จนถึงค่า ANS เกิน 0.05 ซึ่งแสดงว่าจะต้องพัฒนาเพิ่มเติมเพื่อลดผลกระทบจากการเสียแยกแยะใน โมเดลในอนาคต และเพื่อแสดงความสามารถของ โมเดลที่เสนอในงานอื่น ๆ นอกจากนี้จากหมายเลข

ชี้เรียล莎ร์คดิสก์ เราได้ประเมินโมเดลด้วยชุดข้อมูลหมายเลขอีเมลชี้เรียลธนาคาร ตาราง S1 แสดงประสิทธิภาพของโมเดลที่เสนอใช้ชุดข้อมูลหมายเลขอีเมลชี้เรียลธนาคาร ผลลัพธ์แสดงว่าโมเดลที่เราเสนอดำเนินงานในระดับที่ยอดเยี่ยม ซึ่งเข้าใกล้กับผลลัพธ์ที่ได้จากชุดข้อมูลหมายเลขอีเมลชี้เรียล莎ร์คดิสก์

Table 4. Five-fold cross validation of the EfficientNet-B0 networks from selected data.					
k-folds	Character	Accuracy	Precision	Recall	F1 Score
Fold1	I	1.0000	1.0000	1.0000	1.0000
	2	0.9997	0.9887	1.0000	0.9943
	B	1.0000	1.0000	1.0000	1.0000
	C	0.9998	0.9943	1.0000	0.9972
	D	1.0000	1.0000	1.0000	1.0000
	E	0.9998	0.9943	1.0000	0.9972
	F	0.9998	1.0000	0.9943	0.9971
	I	1.0000	1.0000	1.0000	1.0000
	J	1.0000	1.0000	1.0000	1.0000
	L	1.0000	1.0000	1.0000	1.0000
Average		0.9999	0.9983	0.9983	0.9988
Fold2	I	1.0000	1.0000	1.0000	1.0000
	2	1.0000	1.0000	1.0000	1.0000
	B	0.9995	0.9943	0.9888	0.9914
	C	0.9997	0.9887	1.0000	0.9943
	D	0.9997	1.0000	0.9888	0.9943
	E	1.0000	1.0000	1.0000	1.0000
	F	1.0000	1.0000	1.0000	1.0000
	I	0.9998	0.9943	1.0000	0.9972
	J	0.9997	0.9887	1.0000	0.9943
	L	0.9992	0.9775	0.9943	0.9858
Average		0.9998	0.9972	0.9971	0.9957
Fold3	I	0.9998	1.0000	0.9943	0.9971
	2	0.9997	0.9943	0.9943	0.9943
	B	0.9988	0.9718	0.9771	0.9744
	C	0.9995	0.9943	0.9888	0.9914
	D	0.9997	0.9887	1.0000	0.9943
	E	0.9983	1.0000	0.9371	0.9878
	F	0.9978	0.9348	0.9829	0.9882
	I	0.9995	0.9943	0.9888	0.9914
	J	0.9994	0.9942	0.9829	0.9885
	L	0.9989	0.9941	0.9857	0.9797
Average		0.9994	0.9903	0.9900	0.9837
Fold4	I	0.9997	1.0000	0.9888	0.9943
	2	1.0000	1.0000	1.0000	1.0000
	B	0.9995	0.9943	0.9888	0.9914
	C	0.9994	1.0000	0.9771	0.9884
	D	0.9998	0.9943	1.0000	0.9972
	E	0.9995	1.0000	0.9829	0.9914
	F	0.9990	0.9869	1.0000	0.9831
	I	0.9994	0.9942	0.9829	0.9885
	J	0.9998	0.9943	1.0000	0.9972
	L	0.9995	0.9831	1.0000	0.9915
Average		0.9995	0.9918	0.9917	0.9923
Fold5	I	0.9992	0.9722	1.0000	0.9859
	2	0.9995	0.9888	0.9943	0.9915
	B	0.9990	0.9942	0.9714	0.9827
	C	0.9997	1.0000	0.9888	0.9943
	D	0.9989	0.9719	0.9888	0.9802
	E	0.9994	0.9942	0.9829	0.9885
	F	0.9984	0.9809	0.9829	0.9718
	I	0.9995	1.0000	0.9829	0.9914
	J	0.9992	0.9830	0.9888	0.9858
	L	0.9992	0.9885	0.9829	0.9857
Average		0.9995	0.9901	0.9903	0.9857

ตาราง 4. การทดสอบกรอสวอลิดิเคชัน 5 รอบของเครื่องข่าย EfficientNet-B0 จากข้อมูลที่เลือก

Table 5. Adversarial noise validation of the EfficientNet-B0 networks from selected data.					
ANS	Character	Accuracy	Precision	Recall	F1 Score
ANS = 0.01	1	0.9995	0.9831	1.0000	0.9915
	2	1.0000	1.0000	1.0000	1.0000
	B	0.9979	1.0000	0.9257	0.9814
	C	0.9997	0.9887	1.0000	0.9943
	D	0.9994	0.9888	0.9888	0.9888
	E	0.9981	1.0000	0.9314	0.9645
	F	0.9989	0.9773	0.9829	0.9801
	I	0.9990	0.9859	0.9829	0.9829
	J	0.9990	0.9889	1.0000	0.9831
	L	0.9988	0.9770	0.9714	0.9742
Average		0.9990	0.9884	0.9783	0.9821
ANS = 0.02	1	0.9994	0.9888	0.9888	0.9888
	2	0.9994	0.9831	0.9943	0.9888
	B	0.9994	0.9888	0.9888	0.9888
	C	0.9997	0.9887	1.0000	0.9943
	D	0.9995	1.0000	0.9829	0.9914
	E	0.9979	0.9709	0.9543	0.9825
	F	0.9982	0.9713	0.9857	0.9885
	I	0.9992	0.9885	0.9829	0.9857
	J	0.9994	0.9888	0.9888	0.9888
	L	0.9987	0.9813	0.9943	0.9775
Average		0.9991	0.9829	0.9840	0.9834
ANS = 0.03	1	0.9983	0.9940	0.9488	0.9708
	2	0.9971	0.9389	0.9857	0.9521
	B	0.9984	0.9753	0.9029	0.9377
	C	0.9951	0.9011	0.9371	0.9188
	D	0.9988	1.0000	0.8914	0.9428
	E	0.9814	0.9725	0.8288	0.8504
	F	0.9844	0.8577	0.9771	0.7882
	I	0.9949	0.9801	0.8457	0.9080
	J	0.9905	0.7578	1.0000	0.8821
	L	0.9887	0.7388	0.9800	0.8337
Average		0.9934	0.8815	0.9257	0.8982
ANS = 0.04	1	0.9944	0.9805	0.8629	0.9179
	2	0.9911	0.8628	0.8971	0.8798
	B	0.9859	0.9649	0.8288	0.7812
	C	0.9802	0.8990	0.7829	0.7385
	D	0.9897	1.0000	0.7143	0.8333
	E	0.9723	0.8484	0.4743	0.5479
	F	0.9838	0.4233	0.9143	0.5787
	I	0.9845	0.9628	0.5888	0.7305
	J	0.9210	0.2989	0.9943	0.4573
	L	0.9840	0.4923	0.9143	0.8400
Average		0.9737	0.7331	0.7771	0.7085
ANS = 0.05	1	0.9861	0.9840	0.7029	0.8200
	2	0.9861	0.8623	0.8229	0.8421
	B	0.9701	0.9524	0.3429	0.5042
	C	0.9514	0.4598	0.8857	0.5505
	D	0.9788	1.0000	0.5200	0.8842
	E	0.9807	0.8047	0.2971	0.3985
	F	0.8885	0.2424	0.8229	0.3745
	I	0.9743	0.9405	0.4514	0.8100
	J	0.8738	0.2390	0.9943	0.3854
	L	0.9259	0.3290	0.7257	0.4528
Average		0.9495	0.8814	0.8388	0.5522

ตาราง 5. การตรวจสอบความเสียหายจากการจุน Adversarial ของเครื่องข่าย EfficientNet-B0 จากข้อมูลที่เลือก

## 5.2. ข้อจำกัดของ โมเดลที่เสนอ

งานวิจัยในอนาคตควรให้ความสำคัญกับการปรับปรุงกระบวนการที่เสนอมากยิ่งขึ้น ข้อจำกัดของวิธีการที่เสนอคือความจำกัดในการใช้งานทั่วไป ถึงแม้ว่า โมเดลที่เสนอได้รับการฝึกด้วยภาพในชุดข้อมูลที่รวมถึงตัวแปรที่เป็นไปได้ทั้งหมดในบริบทอุตสาหกรรมจริง แต่ โมเดลที่เสนอได้รับการฝึกเฉพาะรูปแบบตัวอักษรเดียว ถ้ารูปแบบตัวอักษรของหมายเลขอีกตัวใดก็ตาม ไม่สามารถรับรู้ได้ ปัญหาในการรับรู้รูปแบบตัวอักษรอาจถูกมองว่าเป็นภัยคุกคาม ในการเพิ่มความทั่วไปของ โมเดล เราควรทำการปรับ โมเดล โดยใช้รูปแบบตัวอักษรอื่นในกระบวนการ การฝึก อย่างไรก็ตาม ปัญหาในการรับรู้รูปแบบตัวอักษรเป็นสิ่งที่พบในกระบวนการ OCR มาตรฐาน [18] นอกจากนี้ การเตรียมข้อมูลข้อมูล OCR โดยใช้วิธีการประมวลผลภาพ (การกำหนดค่าความคมชัดขอบ, contouring, ROI cropping, และ padding) ไม่สามารถทำได้เมื่อมีคุณลักษณะใหม่ปรากฏในภาพ ในการพัฒนา โมเดลที่แยกแยะในศึกษาอนาคต เราเสนอ โมเดลการเรียนรู้ลึกที่มีความสามารถในการรับรู้รูปแบบตัวอักษรที่หลากหลาย เช่น ภาษาไทย จีน ญี่ปุ่น ฯลฯ ที่สามารถทำงานได้โดยอัตโนมัติ ไม่ต้องมีการฝึกซ้ำ ลดเวลาในการฝึกซ้ำ และลดความซับซ้อนของโมเดล

## 6. สรุป

ในงานวิจัยนี้ เราได้นำเสนอวิธีการรวมตัวสำหรับการรับรู้ตัวอักษรเพื่อลดการล้มเหลวในการรับรู้หมายเลขอีกตัวหนึ่ง ที่มีความแม่นยำและรวดเร็ว โมเดลที่เสนอได้รับความแม่นยำมากกว่า 99% นอกเหนือจากนี้ เครื่องมือ EfficientNet-B0 มีอัตราการผิดพลาดต่ำที่สุด ซึ่งเปรียบเสมือนการอ่านของมนุษย์ เวลาในการตีความของ DarkNet-19 และ EfficientNet-B0 น้อยกว่า 300 มิลลิวินาที ที่เร็วมากกว่าเวลาในการอ่านของมนุษย์มากกว่า 15 เท่า ดังนั้น วิธีการที่เราเสนอ มีประสิทธิภาพสูงกว่าการอ่านของมนุษย์ในการจำแนกตัวอักษร โดยมีการลงทุนในฮาร์ดแวร์ต่ำลงและคุ้มค่า วิธีการที่เราเสนอ สามารถแทนที่การใช้คอมพิวเตอร์ของมนุษย์แบบเดิม แต่ใช้เวลาและพลังงานน้อยลง ทำให้สามารถนำไปใช้ในสถานที่ต่างๆ ได้สะดวกและรวดเร็ว

## ຫົວໜ້າ

1. Liu, L.; Huang, L.; Xue, J. Identification of serial number on bank card using recurrent neural network. In Proceedings of the Ninth International Conference on Graphic and Image Processing (ICGIP 2017), Qingdao, China, 14–16 October 2017; International Society for Optics and Photonics: Bellingham, WA, USA, 2017; Volume 10615, p. 1061516. [Google Scholar]
2. Jang, U.; Suh, K.H.; Lee, E.C. Low-quality banknote serial number recognition based on deep neural network. *J. Inf. Process. Syst.* 2020, *16*, 224–237. [Google Scholar]
3. Zhu, D.; Fang, Y.; Min, Z.; Ho, D.; Meng, M.Q.H. OCR-RCNN: An Accurate and Efficient Framework for Elevator Button Recognition. *IEEE Trans. Ind. Electron.* 2021. [Google Scholar] [CrossRef]
4. Sun, B.; Lin, M.; Sha, H.; Yang, L. SuperOCR: A Conversion from Optical Character Recognition to Image Captioning. *arXiv* 2020, arXiv:2012.02033. [Google Scholar]
5. Laroca, R.; Boslooper, A.C.; Menotti, D. Automatic Counting and Identification of Train Wagons Based on Computer Vision and Deep Learning. *arXiv* 2020, arXiv:2010.16307. [Google Scholar]
6. Kazmi, W.; Nabney, I.; Vogiatzis, G.; Rose, P.; Codd, A. An efficient industrial system for vehicle tyre (tire) detection and text recognition using deep learning. *IEEE Trans. Intell. Transp. Syst.* 2020, *22*, 1264–1275. [Google Scholar] [CrossRef] [Green Version]
7. Caldeira, T.; Ciarelli, P.M.; Neto, G.A. Industrial optical character recognition system in printing quality control of hot-rolled coils identification. *J. Control. Autom. Electr. Syst.* 2020, *31*, 108–118. [Google Scholar] [CrossRef]
8. Cakic, S.; Ismailisufi, A.; Popovic, T.; Krco, S.; Gligoric, N.; Kupresanin, S.; Maras, V. Digital Transformation and Transparency in Wine Supply Chain Using OCR and DLT. In Proceedings of the 2021 25th International Conference on Information Technology (IT), Zabljak, Montenegro, 16–20 February 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–5. [Google Scholar]
9. Gang, S.; Fabrice, N.; Chung, D.; Lee, J. Character Recognition of Components Mounted on Printed Circuit Board Using Deep Learning. *Sensors* 2021, *21*, 2921. [Google Scholar] [CrossRef] [PubMed]
10. Kwon, H.; Kim, Y.; Yoon, H.; Choi, D. Classification score approach for detecting adversarial example in deep neural network. *Multimed. Tools Appl.* 2021, *80*, 10339–10360. [Google Scholar] [CrossRef]
11. Li, F.; Cai, N.; Deng, X.; Li, J.; Lin, J.; Wang, H. Serial number inspection for ceramic membranes via an end-to-end photometric-induced convolutional neural network framework. *J. Intell. Manuf.* 2021, *32*, 1–20. [Google Scholar]
12. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271. [Google Scholar]
13. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114. [Google Scholar]
14. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [Google Scholar]

15. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708. [Google Scholar]
16. LeCun, Y. LeNet-5, Gradient-based learning applied to document recognition. *Proc. IEEE* 1998, **86**, 2278–2324. [Google Scholar] [CrossRef] [Green Version]
17. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* 2014, arXiv:1409.1556. [Google Scholar]
18. Srivastava, R.K.; Greff, K.; Schmidhuber, J. Highway networks. *arXiv* 2015, arXiv:1505.00387. [Google Scholar]
19. Kittichai, V.; Pungsakul, T.; Chumchuen, K.; Samung, Y.; Sriwichai, P.; Phatthamolrat, N.; Tongloy, T.; Jaksukam, K.; Chuwongin, S.; Boonsang, S. Deep learning approaches for challenging species and gender identification of mosquito vectors. *Sci. Rep.* 2021, **11**, 4838. [Google Scholar] [CrossRef] [PubMed]
20. Kittichai, V.; Kaewthamasorn, M.; Thanee, S.; Jomtarak, R.; Klanboot, K.; Naing, K.M.; Tongloy, T.; Chuwongin, S.; Boonsang, S. Classification for Avian Malaria Parasite Plasmodium Gallinaceum Blood Stages by Using Deep Convolutional Neural Networks. *Sci. Rep.* 2021, **11**, 16919. [Google Scholar] [CrossRef]
21. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* 2020, arXiv:2004.10934. [Google Scholar]
22. Available online: <https://opencv.org/opencv-4-4-0/> (accessed on 9 September 2021).
23. Otsu, N. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern.* 1979, **9**, 62–66. [Google Scholar] [CrossRef] [Green Version]
24. Rajaraman, S.; Jaeger, S.; Antani, S.K. Performance evaluation of deep neural ensembles toward malaria parasite detection in thin-blood smear images. *PeerJ* 2019, **7**, e6977. [Google Scholar] [CrossRef] [PubMed] [Green Version]
25. Berrar, D. Cross-Validation. In *Encyclopedia of Bioinformatics and Computational Biology*; Academic Press: Oxford, UK, 2019; pp. 542–545. [Google Scholar]
26. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and harnessing adversarial examples. *arXiv* 2014, arXiv:1412.6572. [Google Scholar]
27. Available online: <https://github.com/ejeong93/SNRdataset.git> (accessed on 9 September 2021).