

# การยกระดับการตรวจสอบในอุตสาหกรรมด้วยเทคนิคการประมวลผลภาพโดยใช้ OpenCV

## Advancing Industrial Inspection through Image Processing Techniques with OpenCV

คัมภีร์ ลิมปดาพันธุ์<sup>1</sup> และธีรพงศ์ บริรักษ์<sup>2\*</sup>

Khompee Limpadapun<sup>1</sup> and Theerapong Borirak<sup>2\*</sup>

<sup>1</sup>คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลกรุงเทพ

<sup>1</sup>Faculty of Engineering, Rajamangala University of Technology Krungthep

<sup>2</sup>คณะวิศวกรรมศาสตร์ มหาวิทยาลัยอีสเทิร์นเอเซีย

<sup>2</sup>School of Engineering, Eastern Asia University

\*Corresponding author: t\_borirak@eau.ac.th

Received: November 29, 2024

Revised: February 25, 2025

Accepted: March 4, 2025

### บทคัดย่อ

การควบคุมคุณภาพในภาคการผลิตได้พัฒนาก้าวหน้าอย่างเห็นได้ชัดในช่วงที่ผ่านมา แม้ว่าการตรวจสอบด้วยวิธีดั้งเดิมโดยใช้แรงงานคนมักเผชิญกับข้อจำกัดด้านความแม่นยำ ความสม่ำเสมอ และต้นทุนที่สูง ด้วยเหตุนี้ เทคโนโลยีการประมวลผลภาพด้วย OpenCV ได้ถูกนำมาประยุกต์ใช้เพื่อเพิ่มประสิทธิภาพของการตรวจสอบอัตโนมัติ จากการศึกษาพบว่าการบูรณาการระบบที่ใช้ OpenCV เข้ากับสายการผลิตที่มีอยู่สามารถเพิ่มความแม่นยำในการตรวจจับข้อบกพร่องและลดระยะเวลาในการตรวจสอบได้อย่างมีนัยสำคัญ และช่วยลดระยะเวลาในการตรวจสอบเมื่อเทียบกับวิธีดั้งเดิม การทดสอบในอุตสาหกรรมอิเล็กทรอนิกส์ ยานยนต์ และสิ่งทอ แสดงให้เห็นว่าการประยุกต์ใช้อัลกอริทึมคอมพิวเตอร์วิทัศน์สำหรับการวัดขนาด การวิเคราะห์รูปร่าง และการตรวจสอบสี ช่วยให้การควบคุมคุณภาพมีความแม่นยำและสม่ำเสมอยิ่งขึ้น การใช้ OpenCV สามารถลดข้อผิดพลาดในการผลิต ปรับปรุงความแม่นยำของการตรวจสอบ และเพิ่มประสิทธิภาพของกระบวนการควบคุมคุณภาพได้อย่างมีนัยสำคัญ นอกจากนี้ ระบบยังสามารถทำงานได้ดีภายใต้สภาพแวดล้อมการผลิตที่หลากหลาย ส่งผลให้สามารถลดต้นทุนและเพิ่มความสามารถในการแข่งขันของอุตสาหกรรม อย่างไรก็ตาม ข้อจำกัดบางประการ เช่น ความไวต่อสภาพแสงและความซับซ้อนของรูปทรงผลิตภัณฑ์ ยังคงต้องได้รับการพัฒนาเพิ่มเติม แนวทางที่นำเสนอนี้ช่วยเป็นรากฐานสำคัญสำหรับการพัฒนาระบบตรวจสอบอัตโนมัติที่ทันสมัยและมีประสิทธิภาพสูงในภาคอุตสาหกรรม

**คำสำคัญ:** ระบบการมองเห็นของเครื่องจักร การประกันคุณภาพ การตรวจสอบอัตโนมัติ

## Abstract

Quality control in manufacturing has significantly progressed in recent years. However, traditional manual inspection methods often face challenges in terms of accuracy, consistency, and operational costs. To address these issues, OpenCV-based image processing technology has been adopted to enhance the efficiency of automated inspection systems. Studies have shown that integrating OpenCV into existing production lines significantly enhances defect detection accuracy while reducing inspection time. Applications in the electronics, automotive, and textile industries have shown that using computer vision algorithms for defect detection, size measurement, shape analysis, and color inspection improves quality control precision and consistency. Additionally, OpenCV-based systems have proven effective in minimizing production errors, optimizing inspection efficiency, and reducing operational costs. These systems also operate reliably across diverse production environments, further enhancing industrial competitiveness. However, certain limitations such as sensitivity to lighting variations and complex object geometries still need further development. This approach lays a strong foundation for improving automated quality control systems, enabling a more accurate, cost-effective, and modern industrial inspection process.

**Keywords:** machine vision, quality assurance, automated inspection



## บทนำ

ในอุตสาหกรรมการผลิตสมัยใหม่ การตรวจสอบคุณภาพผลิตภัณฑ์เป็นกระบวนการที่สำคัญอย่างยิ่งในการรักษามาตรฐานและสร้างความพึงพอใจให้กับลูกค้า (Ivanchenkova & Stasiukova, 2019) แต่ในสภาวะการณปัจจุบัน วิธีการตรวจสอบแบบดั้งเดิมด้วยการใช้แรงงานคนหรือเครื่องมือวัดทางกลนั้นมักจะใช้เวลาาน มีต้นทุนสูงและมีความแม่นยำที่จำกัด (Sachan et al., 2020) ด้วยความก้าวหน้าทางเทคโนโลยีคอมพิวเตอร์วิทัศน์ (computer vision) และการประมวลผลภาพ (image processing) ทำให้มีการพัฒนาการตรวจสอบคุณภาพในกระบวนการผลิตให้เป็นระบบอัตโนมัติ (Reyna et al., 2022) ซึ่งส่งผลไปสู่การปรับปรุงประสิทธิภาพโดยรวม และยังเพิ่มความเร็วและความแม่นยำ และลดข้อผิดพลาดของมนุษย์ในกระบวนการผลิต ซึ่งนำไปสู่คุณภาพของผลิตภัณฑ์ที่สม่ำเสมอและความปลอดภัยในการทำงาน (Rathod & Salehi, 2020)

การพัฒนาการตรวจสอบคุณภาพในงานอุตสาหกรรมผ่านเทคนิคการประมวลผลภาพด้วย OpenCV ซึ่งเป็นไลบรารีโอเพ่นซอร์ส (opensource library) หรือซอฟต์แวร์เสรีที่ได้รับความนิยมอย่างกว้างขวางในการพัฒนางานทาง

ด้านคอมพิวเตอร์วิทัศน์ (Cervera, 2020; Jain, Harisinghani, Gangar & Kampli, 2022; Guillen, 2019) และเทคโนโลยีการเรียนรู้เชิงลึก (deep learning) ซึ่ง OpenCV มีเครื่องมือสำหรับการประมวลผลภาพที่สามารถใช้งานร่วมกับแบบจำลองการเรียนรู้เชิงลึก โดยเฉพาะการตรวจจับข้อบกพร่องโดยอัตโนมัติและปรับปรุงความแม่นยำในการตรวจสอบ (Shaikh et al., 2022) การบูรณาการนี้นำไปสู่คุณภาพของผลิตภัณฑ์และประสิทธิภาพในการดำเนินงานที่ดีขึ้น (Honzátko et al., 2021)

บทความนี้มุ่งนำเสนอแนวคิด เทคนิค และกรณีศึกษาการใช้งาน OpenCV สำหรับการตรวจสอบคุณภาพในงานอุตสาหกรรม พร้อมทั้งนำเสนอจุดแข็งและข้อได้เปรียบเมื่อเทียบกับวิธีการแบบดั้งเดิม (Marques & Pitarma, 2021) และการพิจารณาในทางปฏิบัติสำหรับการนำ OpenCV ไปประยุกต์ใช้งานจริงให้เหมาะสมกับเงื่อนไขเฉพาะของแต่ละอุตสาหกรรม (Khan & Zabeeulla, 2023) เนื้อหาในบทความนี้จะประโยชน์ต่อผู้ที่สนใจพัฒนาระบบการตรวจสอบด้วยการมองเห็นของคอมพิวเตอร์ในบริบทของอุตสาหกรรมการผลิต

## พื้นฐานการประมวลผลภาพ (image processing) ด้วย OpenCV สำหรับการตรวจสอบอุตสาหกรรม

การประมวลผลภาพเพื่อการตรวจสอบในอุตสาหกรรม ประกอบด้วยขั้นตอนสำคัญหลายขั้นตอนที่ต้องดำเนินการอย่างเป็นระบบ กระบวนการพื้นฐานที่จำเป็นสำหรับการประมวลผลภาพในงานอุตสาหกรรม ซึ่งเริ่มตั้งแต่การเตรียมภาพ การปรับปรุงคุณภาพ ไปจนถึงการวิเคราะห์เพื่อตรวจจับข้อบกพร่อง (Ciora & Simion, 2014) ดังภาพ 1 แสดงขั้นตอนพื้นฐานในการประมวลผลภาพสำหรับงานอุตสาหกรรม ซึ่งประกอบด้วยกระบวนการที่เชื่อมโยงต่อเนื่องกัน 4 ขั้นตอน เริ่มจากการรับภาพ (image acquisition) ที่ได้จากอุปกรณ์บันทึกภาพ จากนั้นภาพจะถูกนำไปผ่านการประมวลผลเบื้องต้น (pre-processing) เพื่อปรับปรุงคุณภาพของภาพ ต่อมาในขั้นตอนการแบ่งส่วนภาพ (segmentation) จะทำการแยกส่วนที่สนใจออกจากพื้นหลัง และสุดท้ายคือแยกลักษณะเด่น (feature extraction) เพื่อดึงข้อมูลสำคัญจากภาพสำหรับการวิเคราะห์ต่อไป กระบวนการทั้งหมดนี้เชื่อมต่อกันด้วยลูกศรแสดงลำดับขั้นตอนการทำงานที่ต่อเนื่องและเป็นระบบ

การประยุกต์ใช้ OpenCV ในอุตสาหกรรมช่วยยกระดับการตรวจสอบคุณภาพให้มีความแม่นยำและรวดเร็วยิ่งขึ้น โดยสามารถรองรับสายการผลิตที่มีความเร็วสูงและรูปทรงของผลิตภัณฑ์ที่หลากหลาย ผ่านการทดสอบในภาคอุตสาหกรรมที่แตกต่างกัน พบว่า

**อุตสาหกรรมอิเล็กทรอนิกส์:** ใช้ OpenCV ในการตรวจสอบตำแหน่งชิ้นส่วนอิเล็กทรอนิกส์ การวิเคราะห์ขนาดของแผงวงจร (PCB) (Pineda-Perdomo & Villatoro-Flores, 2023) และการตรวจสอบรอยบัดกรี เพื่อเพิ่มความแม่นยำและลดข้อผิดพลาดที่อาจเกิดขึ้นจากแรงงานคน

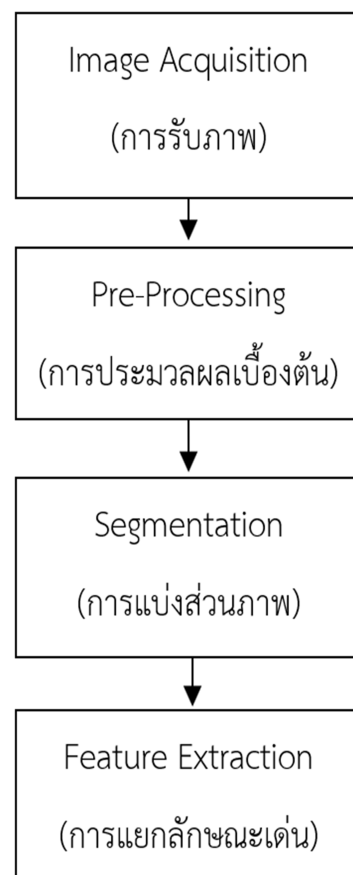
**อุตสาหกรรมยานยนต์:** OpenCV ถูกนำมาใช้ในการตรวจสอบคุณภาพของชิ้นส่วนอะไหล่ (Mohan et al., 2024) การวิเคราะห์พื้นผิวของตัวถังรถยนต์ (Wei et al., 2024) และการตรวจจับข้อบกพร่องของชิ้นส่วนภายในเครื่องยนต์ ซึ่งช่วยลดความจำเป็นในการตรวจสอบด้วยสายตามนุษย์ และเพิ่มความเร็วในการประเมินคุณภาพของชิ้นส่วนที่ผลิต

**อุตสาหกรรมสิ่งทอ:** ระบบประมวลผลภาพด้วย OpenCV สามารถตรวจจับความผิดปกติของเนื้อผ้า การเปรียบเทียบเฉดสีของผ้าในกระบวนการย้อม และการตรวจ

สอบลวดลายของเนื้อผ้า เพื่อให้มั่นใจว่าผลิตภัณฑ์ตรงตามมาตรฐานคุณภาพ (Varjovi et al., 2022)

แม้ว่าผลลัพธ์ของการทดสอบจะแตกต่างกันไปตามลักษณะของแต่ละอุตสาหกรรม แต่โดยรวมพบว่า OpenCV สามารถเพิ่มประสิทธิภาพของกระบวนการตรวจสอบ ลดเวลาการตรวจสอบ และช่วยลดต้นทุนจากข้อผิดพลาดในการผลิต ทำให้เป็นเครื่องมือที่เหมาะสมกับภาคอุตสาหกรรมที่ต้องการความรวดเร็วและความแม่นยำสูง

การประมวลผลภาพเบื้องต้นเป็นขั้นตอนสำคัญที่ช่วยปรับปรุงคุณภาพของภาพก่อนนำไปวิเคราะห์ในขั้นตอนถัดไป ในการตรวจสอบผลิตภัณฑ์อุตสาหกรรมคุณภาพของภาพมีผลโดยตรงต่อความแม่นยำในการตรวจจับข้อบกพร่อง การวัดคุณลักษณะที่สำคัญสามารถแบ่งได้เป็น 3 ประเภทหลัก ดังนี้



ภาพ 1 ขั้นตอนพื้นฐานในการประมวลผลภาพสำหรับงานอุตสาหกรรม

## การวัดขนาด (size measurement)

การวัดขนาดเป็นหนึ่งในฟังก์ชันพื้นฐานที่สำคัญของ OpenCV สำหรับการตรวจสอบคุณภาพในอุตสาหกรรม โดยมีขั้นตอนการทำงานประกอบด้วย 3 ขั้นตอนหลักที่เชื่อมต่อกัน เริ่มจากการนำเข้าภาพ (input image) จากนั้นผ่านขั้นตอนการประมวลผลเบื้องต้น (preprocessing) ซึ่งรวมถึงการกำหนดค่าขีดแบ่ง (thresholding) และการลดสัญญาณรบกวน (noise reduction) ก่อนที่จะนำไปสู่ขั้นตอนสุดท้าย คือ การวัดขนาด (size measurement) ของวัตถุในภาพ (Yongjian et al., 2020) ดังแสดงในภาพ 2

ตัวอย่างการวิเคราะห์พื้นที่และเส้นรอบรูปด้วย OpenCV แสดงการใช้ฟังก์ชัน `contourArea()` และ `arcLength()` โดยมีขั้นตอนการทำงานที่สำคัญเริ่มจากการเตรียมภาพโดยแปลงเป็นภาพขาวดำและใช้ `threshold` เพื่อแยกวัตถุออกจากพื้นหลัง จากนั้นทำการหาเส้นรอบรูป (Contours) ด้วยฟังก์ชัน `findContours()` โดยใช้พารามิเตอร์ `RETR_EXTERNAL` เพื่อหาเฉพาะขอบนอกสุด หลังจากได้ขอบของวัตถุแล้ว สามารถคำนวณหาพื้นที่ด้วยฟังก์ชัน `contourArea()` ซึ่งเป็นการคำนวณพื้นที่วัตถุที่ตรวจจับได้โดยใช้สมการพื้นที่ของรูปหลายเหลี่ยมจากจุดพิกัด  $(x_i, y_i)$  ดังนี้

$$A = \frac{1}{2} \left| \sum_{i=1}^n (x_i y_{i+1} - x_{i+1} y_i) \right| \quad (1)$$

โดยที่:

- $(x_i, y_i)$  เป็นพิกัดของจุดที่สร้างเส้นรอบรูป
- $n$  คือ จำนวนจุดในเส้นรอบรูป (contour)
- จุด  $(x_{n+1}, y_{n+1})$  คือ จุดเริ่มต้นซ้ำ (วนรอบปิด)

คำนวณเส้นรอบรูปด้วยฟังก์ชัน `arcLength()` ซึ่งมีสมการคำนวณ ดังนี้

$$P = \sum_{i=1}^n \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (2)$$

โดยที่:

- $P$  คือ เส้นรอบรูป
- $(x_i, y_i)$  และ  $(x_{n+1}, y_{n+1})$  เป็นจุดที่ต่อกันในเส้นรอบรูป
- ค่ารวมเป็นผลรวมของระยะห่างระหว่างจุดทุกคู่ที่เชื่อมกัน

นอกจากนี้ยังสามารถใช้ฟังก์ชัน `boundingRect()` เพื่อหาขนาดพื้นที่สี่เหลี่ยมที่ครอบคลุมวัตถุโดยใช้การเปรียบเทียบกับกรวัดพื้นที่ในแบบสามมิติ ซึ่งแสดงรูปแบบการเขียนโค้ดดังภาพ 3

จากภาพ 4 แสดงการวัดขนาดของวัตถุ 2 รูปแบบ ได้แก่ สี่เหลี่ยมผืนผ้าและวงกลม โดยแสดงให้เห็น โดยผลจากการวัด เส้นสีเขียว คือ Contour ที่ตรวจจับได้ ข้อความสีแดง คือ พื้นที่ในหน่วย pixel ข้อความสีน้ำเงิน คือ เส้นรอบรูป (perimeter) ในหน่วย pixel และเส้นประสีเทา แสดงการวัดขนาดมิติต่าง ๆ

ในการประยุกต์ใช้งานจริง จำเป็นต้องมีการปรับเทียบ (calibration) เพื่อแปลงหน่วย pixel เป็นหน่วยวัดจริง เช่น มิลลิเมตร โดยใช้วัตถุอ้างอิงที่ทราบขนาดแน่นอน ตัวอย่างการประยุกต์ใช้งานในอุตสาหกรรมได้แก่ การตรวจสอบขนาดของชิ้นส่วนในสายการผลิต การวัดพื้นที่ของตำหนิบนผิวผลิตภัณฑ์ การควบคุมคุณภาพในกระบวนการตัดเฉือนวัสดุ

## การวิเคราะห์รูปร่าง (shape analysis)

การวิเคราะห์รูปร่างด้วย OpenCV เป็นเทคนิคสำคัญในการตรวจสอบคุณภาพผลิตภัณฑ์ในอุตสาหกรรม (Saif et al., 2020) โดยสามารถวิเคราะห์คุณลักษณะทางเรขาคณิตของวัตถุ เช่น พื้นที่ เส้นรอบรูป จุดศูนย์กลางมวล และความสมมาตร โดยใช้เทคนิคหลัก 3 ประการ ได้แก่ การวิเคราะห์โมเมนต์ (moments) ผ่านฟังก์ชัน `cv2.moments()` เพื่อคำนวณ Spatial moments และจุดศูนย์กลางมวล ซึ่งในการคำนวณโมเมนต์เป็นไปตามสมการนี้

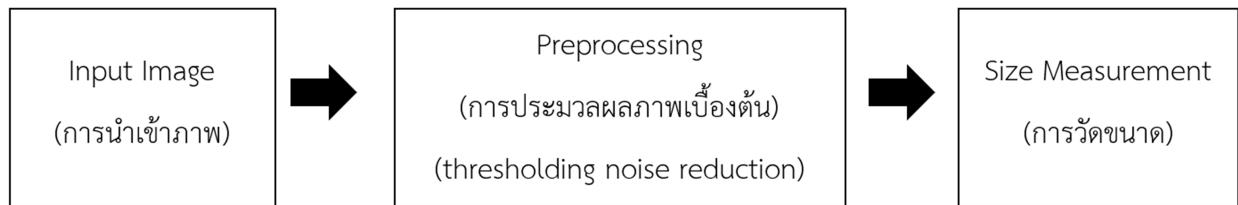
$$M_{pq} = \sum_x \sum_y x^p y^q I(x, y) \quad (3)$$

โดยที่:

- $M_{pq}$  คือ ค่าของ Moment ลำดับที่  $p, q$
- $I(x, y)$  คือ ค่าความเข้มของพิกเซลที่จุด  $(x, y)$
- $p, q$  เป็นเลขชี้กำลังที่กำหนดให้ moment แต่ละตัว

การประมาณรูปทรงเรขาคณิตด้วยฟังก์ชัน cv2.approxPolyDP() สำหรับการวิเคราะห์เส้นขอบและจุดมุม และการวิเคราะห์คุณสมบัติเพิ่มเติม เช่น อัตราส่วนความ

กว้างต่อความสูง ความกลม และความสมมาตร ดังแสดงในตัวอย่างโค้ด รูปแบบโค้ดการวิเคราะห์รูปร่างและการหาจุดศูนย์กลาง ดังแสดงในภาพ 5

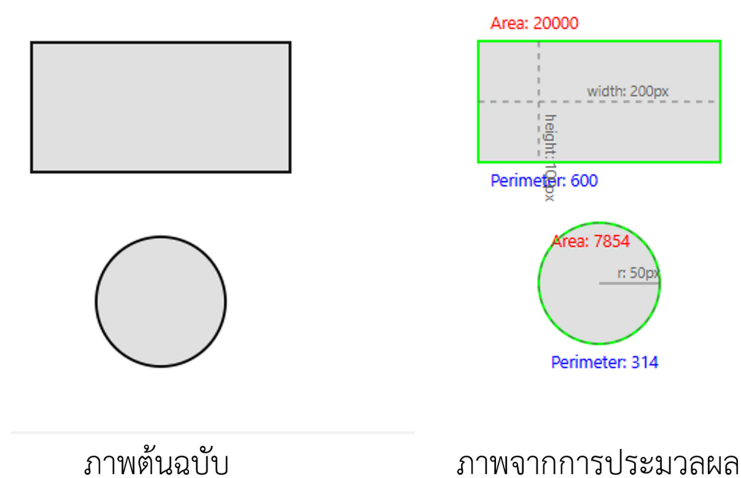


ภาพ 2 ขั้นตอนการวัดขนาดด้วยวิธีการประมวลผลภาพ

```

import cv2
# อ่านภาพ
img = cv2.imread('sample.jpg')
# แปลงเป็นภาพขาวดำ
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
# หาขอบของวัตถุ
contours, _ = cv2.findContours(gray, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
# วัดขนาด
area = cv2.contourArea(contours[0])
perimeter = cv2.arcLength(contours[0], True)
  
```

ภาพ 3 รูปแบบโค้ดการวัดขนาดด้วย OpenCV



ภาพ 4 ตัวอย่างแสดงการวัดขนาดของวัตถุ 2 รูปแบบ ได้แก่ สีเหลี่ยมผืนผ้าและวงกลม

```
# คำนวณ Moments
M = cv2.moments(contour)

# หาจุดศูนย์กลาง
cx = int(M['m10']/M['m00'])
cy = int(M['m01']/M['m00'])
```

ภาพ 5 รูปแบบโค้ดการวิเคราะห์รูปร่างและการหาจุดศูนย์กลาง

จากภาพ 6 แสดงการวิเคราะห์รูปทรงเรขาคณิตด้วย OpenCV โดยเปรียบเทียบระหว่างภาพต้นฉบับ (ซ้าย) และผลลัพธ์จากการประมวลผล (ขวา) ซึ่งแสดงการระบุจุดศูนย์กลางมวล (centroid) และการประมาณรูปทรงเรขาคณิตของวัตถุรูปแบบต่าง ๆ จากโค้ดตัวอย่างข้างต้น เราสามารถคำนวณจุดศูนย์กลางของวัตถุได้จากค่า Moments โดยใช้อัตราส่วนระหว่าง การตรวจสอบสี

การตรวจสอบสีเป็นองค์ประกอบสำคัญในการตรวจสอบคุณภาพผลิตภัณฑ์โดย OpenCV สนับสนุนการวิเคราะห์ในหลายพื้นที่สี (color spaces) เพื่อความเหมาะสมกับงานที่แตกต่างกัน (Tiuftiakov et al., 2021) การแปลงพื้นที่สีเป็นพื้นฐานสำคัญในการวิเคราะห์ เช่น การแปลงจาก BGR เป็น HSV ซึ่งแยกองค์ประกอบสี ความอิ่มตัว และความสว่างออกจากกัน หรือการใช้พื้นที่สี LAB ที่สอดคล้องกับการรับรู้ของมนุษย์ (Abhigna, 2024) รูปแบบโค้ดการตรวจสอบสี ดังแสดงในภาพ 7 ในโค้ดนี้ใช้ฟังก์ชัน cv2.cvtColor() เพื่อแปลงพื้นที่สีจาก BGR (Blue-Green-Red) ไปเป็น HSV (Hue-Saturation-Value) และ LAB (Lightness-a-b) ซึ่งมีสมการ ดังนี้

$$H = a \tan \left( \frac{\sqrt{3}(G - B)}{2R - G - B} \right) \quad (4)$$

$$S = 1 - \frac{3}{(R + G + B)} \cdot \min(R, G, B)$$

$$V = \max(R, G, B)$$

โดยที่

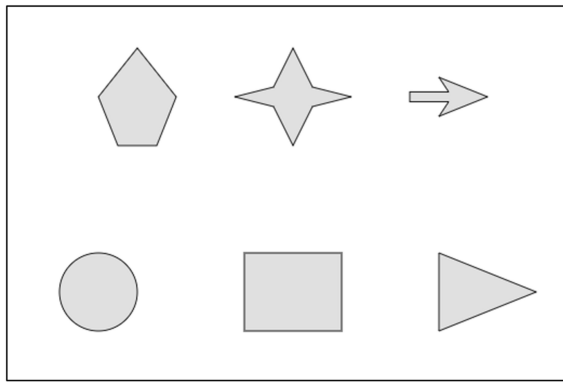
- $H$  คือ ค่าของสี (Hue) เป็นมุมของสีในวงล้อสี ( $0^\circ$  ถึง  $360^\circ$ )
- $S$  คือ ค่าความอิ่มตัวของสี (Saturation)
- $V$  คือ ค่าความสว่าง (Value)

$m10/m00$  สำหรับพิกัด  $x$  และ  $m01/m00$  สำหรับพิกัด  $y$  ซึ่งผลลัพธ์จะปรากฏเป็นจุดสีแดงที่จุดศูนย์กลางของแต่ละรูปทรง ดังแสดงในภาพ 6

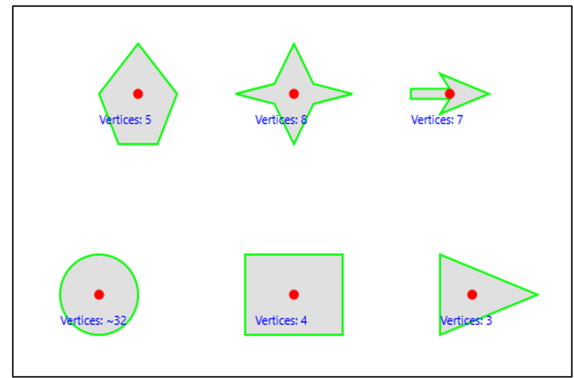
จากภาพ 8 แสดงการเปรียบเทียบผลการวิเคราะห์สีในพื้นที่สีต่าง ๆ ได้แก่ BGR HSV และ LAB พร้อมทั้งแสดงอัลกอริทึมการกระจายตัวของสีสำหรับวัตถุตัวอย่าง 6 สี โดยแต่ละพื้นที่สีแสดงคุณลักษณะเฉพาะที่เหมาะสมกับการวิเคราะห์ที่แตกต่างกัน

จากภาพ 8 แสดงให้เห็นการวิเคราะห์สีใน 2 ระบบที่สำคัญ ได้แก่ ระบบ HSV ที่แยกองค์ประกอบของสี ความอิ่มตัว และความสว่างออกจากกัน ทำให้ง่ายต่อการกำหนดช่วงและการแยกแยะสี และระบบ LAB ที่ออกแบบให้สอดคล้องกับการรับรู้สี

การประยุกต์ใช้ในอุตสาหกรรม การวิเคราะห์สีด้วย OpenCV ได้ถูกนำไปประยุกต์ใช้อย่างกว้างขวางในภาคอุตสาหกรรม โดยเฉพาะในการตรวจสอบคุณภาพของผลิตภัณฑ์ที่เป็นปัจจัยสำคัญ เช่น ในอุตสาหกรรมสิ่งทอ การตรวจสอบความสม่ำเสมอของสีผ้าและการควบคุมคุณภาพการย้อม หรือในอุตสาหกรรมการพิมพ์ที่ต้องการความแม่นยำสูงในการควบคุมสี ระบบสามารถตรวจจับความผิดปกติของการพิมพ์และความคลาดเคลื่อนของสีได้อย่างรวดเร็ว นอกจากนี้ ในสายการผลิตบรรจุภัณฑ์ การตรวจสอบสีของฉลากและการพิมพ์บนบรรจุภัณฑ์ช่วยให้นักผลิตมั่นใจได้ว่าผลิตภัณฑ์ที่ออกสู่ตลาดมีคุณภาพตรงตามมาตรฐานที่กำหนด การคัดแยกผลิตภัณฑ์ตามสีก็เป็นอีกหนึ่งการประยุกต์ใช้ที่ช่วยเพิ่มประสิทธิภาพในกระบวนการผลิตและการควบคุมคุณภาพ



ภาพต้นฉบับ



ภาพจากการประมวลผล

ภาพ 6 การวิเคราะห์รูปทรงเรขาคณิตด้วย OpenCV

```
import cv2
import numpy as np

# อ่านภาพต้นฉบับ
img = cv2.imread('sample.jpg')

# แปลงพื้นที่สี BGR เป็น HSV และ LAB
hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
lab = cv2.cvtColor(img, cv2.COLOR_BGR2LAB)

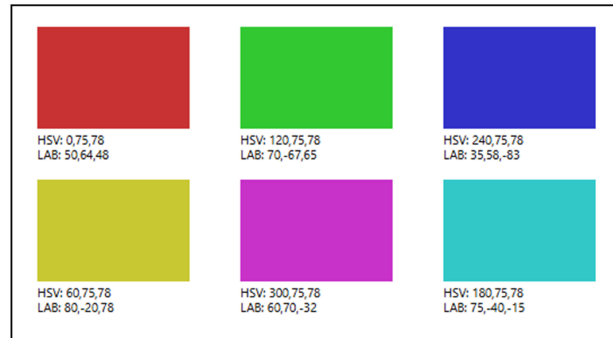
# คำนวณค่าเฉลี่ยสีในแต่ละพื้นที่สี
bgr_mean = cv2.mean(img)[:3]
hsv_mean = cv2.mean(hsv)[:3]
lab_mean = cv2.mean(lab)[:3]

# วิเคราะห์การกระจายตัวของสี
hist = cv2.calcHist([img], [0,1,2], None, [8,8,8], [0,256,0,256,0,256])
```

ภาพ 7 รูปแบบโค้ดการวิเคราะห์สี



ภาพต้นฉบับ



ภาพจากการประมวลผล

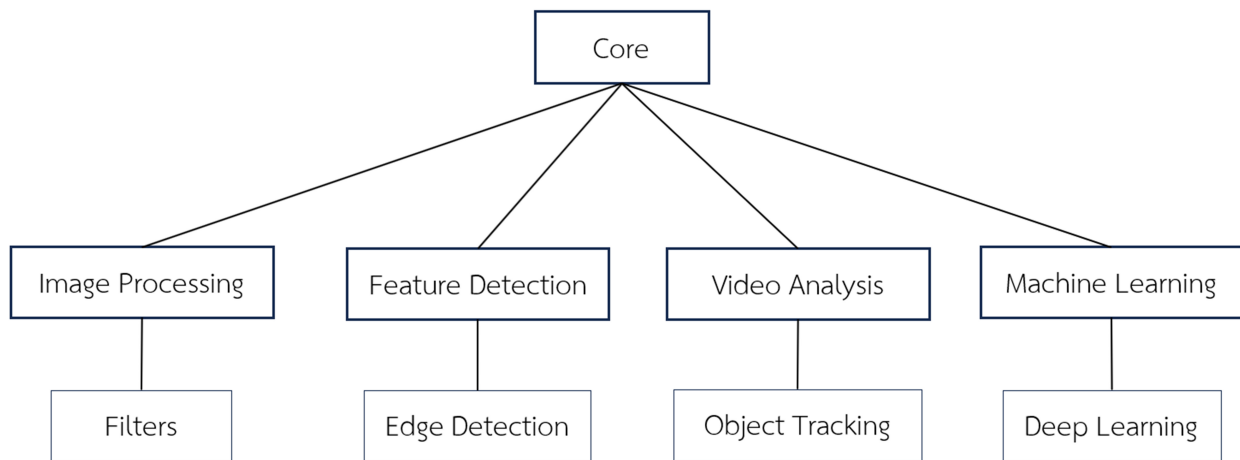
ภาพ 8 การวิเคราะห์สีด้วย OpenCV

## ไลบรารีของ OpenCV

การประยุกต์ใช้การประมวลผลภาพในงานอุตสาหกรรมจำเป็นต้องอาศัยเครื่องมือที่มีประสิทธิภาพและความน่าเชื่อถือสูง OpenCV--Open Source Computer Vision Library เป็นไลบรารีการประมวลผลภาพที่ได้รับความนิยมอย่างแพร่หลายในการประยุกต์ใช้งานด้านการมองเห็นของคอมพิวเตอร์ในอุตสาหกรรม (OpenCV Team, 2024) โดย OpenCV มีโครงสร้างแบบโมดูลาร์ที่ประกอบด้วยไลบรารีย่อยที่ทำงานเชื่อมโยงกัน ดังแสดงในภาพ 9

จากภาพ 9 แสดงให้เห็นว่า OpenCV มีโมดูล Core เป็นศูนย์กลางที่เชื่อมต่อกับโมดูลหลัก 4 ส่วน ได้แก่ Image Processing, Feature Detection, Video Analysis และ Machine Learning โดยแต่ละโมดูลหลักจะมีโมดูลย่อยที่มีความสามารถเฉพาะทาง การทำงานร่วมกันของโมดูลต่าง ๆ เหล่านี้ช่วยให้สามารถพัฒนาระบบการตรวจสอบอัตโนมัติที่มีประสิทธิภาพสูง (Abhigna, 2024) สำหรับรายละเอียดของแต่ละไลบรารีและการประยุกต์ใช้ในงานอุตสาหกรรม ได้แสดงในตาราง 1

จากตาราง 1 ได้โครงสร้างของ OpenCV ที่แบ่งเป็น 5 ไลบรารีหลัก ได้แก่ Core Functionality Image Processing Feature Detection Video Analysis และ Machine Learning โดยแต่ละส่วนมีบทบาทเฉพาะในการประมวลผลภาพ เริ่มจาก Core Functionality ที่รับผิดชอบฟังก์ชันพื้นฐานอย่างการนำเข้า-ส่งออกภาพและการแปลงพื้นที่สี ส่วน Image Processing มุ่งเน้นการปรับปรุงคุณภาพภาพผ่านการกรองสัญญาณรบกวนและการแยกวัตถุ ในขณะที่ Feature Detection ช่วยในการตรวจจับขอบและวิเคราะห์คุณลักษณะของวัตถุ สำหรับการตรวจจับขอบและวิเคราะห์คุณลักษณะของวัตถุ สำหรับ Video Analysis รองรับการวิเคราะห์การเคลื่อนไหวและการติดตามวัตถุ และสุดท้าย Machine Learning เพิ่มความสามารถในการจำแนกประเภทวัตถุและการเรียนรู้เชิงลึก ทั้งหมดนี้ทำให้ OpenCV เป็นเครื่องมือที่ครบครันสำหรับการพัฒนาระบบตรวจสอบคุณภาพในงานอุตสาหกรรม



ภาพ 9 แผนภาพแสดงความสัมพันธ์ของโมดูลต่าง ๆ ในไลบรารี OpenCV

## ตาราง 1

สรุปไลบรารีหลักของ OpenCV และการประยุกต์ใช้ในงานอุตสาหกรรม

| ไลบรารีหลัก           | โมดูลย่อย           | ฟังก์ชันที่สำคัญ                   | การประยุกต์ใช้ในงานอุตสาหกรรม       |
|-----------------------|---------------------|------------------------------------|-------------------------------------|
| Core<br>Functionality | - Image I/O         | - imread(), imwrite()              | - การนำเข้าและส่งออกภาพในระบบ       |
|                       | - Color Space       | - cvtColor()                       | - การแปลงพื้นที่สีเพื่อการวิเคราะห์ |
|                       | - Matrix Operations | - resize(), rotate()               | - การปรับแต่งภาพพื้นฐาน             |
| Image Processing      | - Filtering         | - GaussianBlur(), medianBlur()     | - การลดสัญญาณรบกวนในภาพ             |
|                       | - Morphology        | - erode(), dilate()                | - การปรับปรุงคุณภาพภาพ              |
|                       | - Thresholding      | - threshold(), adaptiveThreshold() | - การแยกวัตถุออกจากพื้นหลัง         |
| Feature<br>Detection  | - Contours          | - findContours(), drawContours()   | - การตรวจจับขอบและรูปร่าง           |
|                       | - Edge Detection    | - Canny(), Sobel()                 | - การวิเคราะห์ขอบของวัตถุ           |
|                       | - Feature Matching  | - SIFT, SURF                       | - การเปรียบเทียบคุณลักษณะ           |
| Video Analysis        | - Motion Detection  | - BackgroundSubtractor             | - การตรวจจับการเคลื่อนไหว           |
|                       | - Object Tracking   | - TrackerCSRT                      | - การติดตามวัตถุแบบเรียลไทม์        |
| Machine<br>Learning   | - Classification    | - SVM, KNN                         | - การจำแนกประเภทวัตถุ               |
|                       |                     | - DNN module                       | - การเรียนรู้เชิงลึก                |

## คุณลักษณะสำคัญของ OpenCV ในการประยุกต์ใช้งาน อุตสาหกรรมสมัยใหม่

OpenCV เป็นไลบรารีโอเพนซอร์สการมองเห็นของคอมพิวเตอร์ สำหรับการประมวลผลและวิเคราะห์ภาพ ความหลากหลายและประสิทธิภาพทำให้เป็นตัวเลือกที่เหมาะสมสำหรับงานอุตสาหกรรมต่าง ๆ รวมถึงการควบคุมคุณภาพ การตรวจจับข้อบกพร่อง และระบบอัตโนมัติ ช่วยเพิ่มประสิทธิภาพการผลิต ความปลอดภัย และความแม่นยำ ทำให้เป็นส่วนสำคัญในภาคอุตสาหกรรมการผลิต

### การรับภาพและการประมวลผล

OpenCV ช่วยในการรับสตรีมวิดีโอจากหลายแหล่ง ทำให้สามารถประมวลผลและวิเคราะห์ภาพแบบเรียลไทม์สามารถประยุกต์ใช้งาน เช่น การตรวจจับข้อบกพร่องและการควบคุมคุณภาพในกระบวนการผลิต (Raileanu et al., 2020) และมีไลบรารีสนับสนุนเทคนิคการประมวลผลภาพต่าง ๆ รวมถึงการแบ่งส่วนภาพ การตรวจจับวัตถุ และการวิเคราะห์การเคลื่อนไหว ซึ่งสำคัญในการใช้ติดตามและควบคุมการปฏิบัติงานในอุตสาหกรรม (Suarez et al., 2014)

### การจัดจำวัตถุและการระบุตำแหน่ง

ความสามารถในการจัดจำวัตถุของ OpenCV ช่วยในการระบุและหาตำแหน่งของชิ้นส่วนอุตสาหกรรม ซึ่งสำคัญต่อระบบอัตโนมัติและหุ่นยนต์ คุณสมบัตินี้มีประโยชน์อย่างยิ่งในกระบวนการจัดการวัสดุและกระบวนการประกอบ (Korta et al., 2014) ความสามารถในการทำงานร่วมกับอุปกรณ์อุตสาหกรรมโดยใช้โปรโตคอลการสื่อสารมาตรฐานเข้ากับกระบวนการผลิตที่มีอยู่ (Raileanu et al., 2020)

### การควบคุมคุณภาพและการตรวจจับข้อบกพร่อง

OpenCV ร่วมกับเทคนิคการเรียนรู้เชิงลึก (deep learning) ช่วยปรับปรุงการตรวจจับข้อบกพร่องและการควบคุมคุณภาพผ่านการทำให้กระบวนการตรวจสอบเป็นอัตโนมัติที่ให้ความแม่นยำและประสิทธิภาพสูง (Islam et al., 2024) ความสามารถในการประมวลผลแบบเรียลไทม์ช่วยลดข้อผิดพลาดและต้นทุนการผลิต (Balakrishna et al., 2024)

## ความคุ้มค่าและความยืดหยุ่น

OpenCV เป็นทางเลือกที่คุ้มค่าแทนระบบสำเร็จรูปที่มีราคาแพงโดยไม่ลดทอนประสิทธิภาพ ลักษณะแบบโอเพนซอร์สช่วยให้สามารถปรับแต่งและปรับตัวตามความต้องการเฉพาะของอุตสาหกรรม ทำให้เป็นเครื่องมือที่ยืดหยุ่นสำหรับการประยุกต์ใช้งานที่หลากหลาย (Korta et al., 2014) ความสามารถในการทำงานร่วมกับแพลตฟอร์มและภาษาโปรแกรมที่แตกต่างกัน ทำให้สามารถทำงานเข้ากับอุตสาหกรรมที่หลากหลาย (Suarez et al., 2014)

## ข้อจำกัดของการใช้ OpenCV ในการประยุกต์ใช้งาน ทางอุตสาหกรรม

OpenCV ยังมีข้อจำกัดสำคัญที่ส่งผลกระทบต่อประสิทธิภาพและการประยุกต์ใช้งาน บทความนี้นำเสนอข้อจำกัดหลักของ OpenCV ใน 3 ด้าน

### ความไวต่อสภาพแวดล้อม

ประสิทธิภาพและความแม่นยำของ OpenCV ขึ้นอยู่กับสภาพแวดล้อมอย่างมาก ในการประยุกต์ใช้งานด้านการตรวจจับเลนถนน ปัจจัยภายนอก เช่น ความเข้มแสงที่แปรผัน พื้นผิวถนน และสิ่งกีดขวางส่งผลกระทบต่อความแม่นยำในการตรวจจับอย่างมีนัยสำคัญ (Vishwakarma et al., 2015) เช่นเดียวความสามารถในการติดตามวัตถุที่มีประสิทธิภาพลดลงในสภาพแวดล้อมที่ซับซ้อนที่มีการบดบังและวัตถุที่เปลี่ยนรูปร่างได้ (Jin & Yang, 2023)

### ข้อจำกัดของอัลกอริทึม

อัลกอริทึมการติดตามใน OpenCV แสดงข้อจำกัดในการจัดการสถานการณ์ที่เปลี่ยนแปลง รวมถึงความท้าทายในการปรับตัวต่อการเปลี่ยนแปลงรูปลักษณ์ ความเข้มแสง และการเคลื่อนที่ของกล้อง โดยเฉพาะอย่างยิ่งในการติดตามคนเดินเท้า (Haggui et al., 2021) เทคนิคการจับคู่แม่แบบ (template matching) ที่ใช้สำหรับค้นหาตำแหน่งของวัตถุในภาพโดยเปรียบเทียบกับภาพต้นแบบ (template) ยังแสดงความอ่อนไหวต่อการเปลี่ยนแปลงของสภาพแวดล้อม แม้จะมีความพยายามในการปรับปรุงแม่แบบแบบไดนามิก (Xiangyu & Weiting, 2013)

## ความต้องการด้านการประมวลผล

OpenCV มีข้อจำกัดด้านทรัพยากรการประมวลผลที่สำคัญ โดยเฉพาะในแอปพลิเคชันที่ต้องการการประมวลผลแบบเรียลไทม์ โดยเฉพาะในระบบตรวจจับการเคลื่อนไหวที่ต้องรับมือกับความผันผวนของสภาพแวดล้อม ทำให้ต้องใช้ฮาร์ดแวร์ที่ซับซ้อนและต้องการทรัพยากรการคำนวณสูง (Mishra et al., 2022) ได้แก่ หน่วยประมวลผล (CPU) และหน่วยความจำ (RAM) จำนวนมาก

## ทิศทางในอนาคตของ OpenCV ในอุตสาหกรรม

### การบูรณาการโมเดลขั้นสูง

การพัฒนาในอนาคตจะมุ่งเน้นการผสมผสานโมเดลการเรียนรู้เชิงลึกที่ซับซ้อน รวมถึง CNN และ GAN เพื่อเพิ่มขีดความสามารถในการจดจำวัตถุและการแบ่งส่วนภาพ (Majhi & Wao, 2024) เทคนิคการเรียนรู้แบบกำกับตนเองจะมีความสำคัญมากขึ้นในการลดการพึ่งพาข้อมูลที่มีการกำกับ

### การปรับปรุงประสิทธิภาพ

การปรับปรุงประสิทธิภาพของการประมวลผลภาพด้วย OpenCV มุ่งเน้นไปที่การปรับปรุงการประมวลผลแบบเรียลไทม์และความสามารถในการขยายระบบบนแพลตฟอร์มฮาร์ดแวร์ต่าง ๆ (Peddie et al., 2016) การปรับปรุงนี้จะมีผลสำคัญต่อการรองรับการใช้งานบนอุปกรณ์เคลื่อนที่และระบบฝังตัว (Stein, 2012)

## การประยุกต์ใช้แบบสหวิทยาการ

การพัฒนาในอนาคตจะเกี่ยวข้องกับการบูรณาการกับความเป็นจริงเสริม ระบบไซเบอร์-กายภาพ และปัญญาประดิษฐ์เชิงสร้างสรรค์ เพื่อสร้างแอปพลิเคชันที่ครอบคลุมและเชื่อมโยงกันมากขึ้น (Liu et al., 2023) แนวทางสหวิทยาการนี้จะมีความสำคัญต่อการพัฒนาอย่างต่อเนื่องของ OpenCV ในด้านคอมพิวเตอร์วิทัศน์ที่กำลังก้าวหน้าอย่างรวดเร็ว

## บทสรุป

OpenCV เป็นเครื่องมือหนึ่งที่จะช่วยสนับสนุนกระบวนการวิจัยด้านการประมวลผลภาพ เข้ามามีบทบาทสำคัญในการพัฒนางานตรวจสอบในภาคอุตสาหกรรม โดยมีจุดเด่นในด้านการประมวลผลภาพที่ครบครัน ไม่ว่าจะเป็นการวัดขนาด การวิเคราะห์รูปร่าง และการตรวจสอบสี ซึ่งสร้างประโยชน์อย่างเห็นได้ชัดในหลากหลายอุตสาหกรรมการผลิต แม้ว่าการใช้งานในปัจจุบันยังมีข้อจำกัดเรื่องความไวต่อสภาพแวดล้อมและความต้องการในการประมวลผล แต่การพัฒนาอย่างต่อเนื่องและการผสมผสานกับเทคโนโลยีใหม่อย่าง AI และระบบประมวลผลแบบกระจาย ก็สร้างโอกาสที่น่าสนใจ การที่ OpenCV เป็นซอฟต์แวร์เสรีและมีชุมชนนักพัฒนาที่แข็งแกร่ง ทำให้มีรากฐานที่มั่นคงสำหรับการประยุกต์ใช้ในอุตสาหกรรม โดยเฉพาะในระบบควบคุมคุณภาพอัตโนมัติ ในขณะที่กระบวนการผลิตมีความซับซ้อนมากขึ้น บทบาทของ OpenCV ในการสร้างระบบตรวจสอบที่มีประสิทธิภาพและแม่นยำจะยังคงมีความสำคัญต่อความก้าวหน้าของภาคอุตสาหกรรมต่อไป



## References

- Abhigna, K. (2024). Integrating OpenCV and pandas for enhanced image filtering and color detection. *International Journal of Scientific Research in Engineering and Management (IJSREM)*, 8(7), 1-5. <https://doi.org/10.55041/ijssrem36213>
- Balakrishna, K., Tiwari, V., Deshpande, A. M., Patil, S., Garg, A. K., & Geetha, B. T. (2024). Computer vision for automated quality inspection in manufacturing. *Conference: 2024 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)* (pp. 1-6). Chennai: IEEE

- Bono, F. M., Radicioni, L., & Cinquemani, S. (2023). A novel approach for quality control of automated production lines working under highly inconsistent conditions. *Engineering Applications of Artificial Intelligence*, 128, 106149. <https://doi.org/10.1016/j.engappai.2023.106149>
- Cervera, E. (2020). GPU-accelerated vision for robots: Improving system throughput using OpenCV and CUDA. *IEEE Robotics & Automation Magazine*, 27(2), 15-27. <https://doi.org/10.1109/MRA.2020.2977601>
- Ciora, R. A., & Simion, C. M. (2014). Industrial applications of image processing. *Acta Universitatis Cibiniensis. Technical Series*, 64(1), 17-21. <https://doi.org/10.2478/aucts-2014-0004>
- Guillen, G. (2019). *Digital image processing with Python and OpenCV*. New York: Springer.
- Haggui, O., Tchalim, M. A., & Magnier, B. (2021). A comparison of OpenCV algorithms for human tracking with a moving perspective camera. *In European Workshop on Visual Information Processing* (pp. 1-6). Paris: IEEE.
- Honzátko, D., Türetken, E., Bigdeli, S. A., Dunbar, L. A., & Fua, P. (2021). Defect segmentation for multi-illumination quality control systems. *Machine Vision and Applications*, 32(4), 1-15. <https://doi.org/10.1007/s00138-021-01244-z>
- Islam, M. R., Zamil, M. Z. H., Rayed, M. E., Kabir, M. M., Mridha, M. F., Nishimura, S., & Shin, J. (2024). Deep learning and computer vision techniques for enhanced quality control in manufacturing processes. *IEEE Access*, 12, 121449-121479. <https://doi.org/10.1109/ACCESS.2024.3453664>
- Ivanchenkova, L., & Stasiukova, K. (2019). Quality of products in the factory system of providing competitiveness of modern production. *Ukrainian Journal of Applied Economics*, 2, 67-75. <https://doi.org/10.36887/2415-8453-2019-2-8>
- Jain, H., Harisinghani, K., Gangar, S., & Kambli, M., (2022). DoodSearch - OpenCV with image recognition. *IEEE Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)* (pp. 1-4). Gwalior: IEEE.
- Jin, Z., & Yang, H. (2024). Real time object tracking using deep learning and OpenCV. *Applied and Computational Engineering*, 35, 272-279. <https://doi.org/10.54254/2755-2721/35/20230406>
- Khan, G., & Zabeeulla, Z., (2023). The smart application development in real time parallel applications with industrial automation. *International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)* (pp. 1-7). Ballar: IEEE.
- Korta, J., Kohut, J., & Uhl, T. (2014). OpenCV based vision system for industrial robot-based assembly station: calibration and testing. *Pomiary Automatyka Kontrola*, 60, 35-38. <https://bibliotekanauki.pl/articles/156732>

- Lin, C. Y., Rahman, M. A. A., & Maropoulos, P. (2022). Enhancing the quality inspection process in food manufacturing through automation. *2022 8th International Conference on Control, Decision and Information Technologies (CoDIT)* (pp. 379-384). Istanbul: IEEE
- Liu, K., Dai, P., Lee, V. C. S., Ng, J. K. Y., & Son, S. H. (2023). *Future directions in industrial computer vision*. London: Springer.
- Majhi, R. K., & Wao, A. A. (2024). Advances in computer vision: New horizons and ongoing challenges. *Visual and Performing Arts Journal*, 5(5), 45-52. <https://doi.org/10.29121/shodhkosh.v5.i5.2024.1893>
- Marques, G., & Pitarma, R. (2021). *Industrial informatics for quality assurance and real-time defect detection through computer vision*. London: Springer.
- Mishra, S., Verma, V., Akhtar, N., Chaturvedi, S., & Perwej, Y. (2022). An intelligent motion detection using OpenCV. *International Journal of Scientific Research*, 9(2), 156-163. <https://doi.org/10.32628/IJSRSET22925>
- Mohan, C. G., Saber, T., & Nallathambi, P. J. (2024). Recognition of industrial spare parts using an optimized convolutional neural network model. *Information*, 15(12), 793. <https://doi.org/10.3390/info15120793>
- OpenCV Team. (2024). *About OpenCV*. Retrieved from <https://opencv.org/about/>
- Peddie, J., Akeley, K., Debevec, P., Fonseca, E., Mangan, M., & Raphael, M. (2016). A vision for computer vision: emerging technologies. *SIGGRAPH '16: Special Interest Group on Computer Graphics and Interactive Techniques Conference* (pp. 1-2). California: Association for Computing Machinery
- Pineda-Perdomo, G. A., & Villatoro-Flores, H. F. (2023). Implementation of a computer vision system for fault and component analysis of computer PCBs. *2023 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT)* (pp. 1-6). San Salvador: IEEE
- Raileanu, S., Borangiu, T., Silisteanu, A., Anton, S., & Anton, F. D. (2020). Open source platform for vision guided robotic systems integrated in manufacturing. *Journal of Control Engineering and Applied Informatics*, 22(4), 3-14. <https://bit.ly/4j2bQ8A>
- Rathod, J. M., & Salehi, H. S. (2020). Vision system with deep learning classifiers for automatic quality inspection. In *Pattern Recognition and Tracking XXXI Vol. 11400* (pp. 22-29) (Online). USA: SPIE.
- Reyna, M., Delgado, G., Akundi, A., Luna, S., & Chumacero, E. M. (2022). Product digital quality inspection using machine vision systems: A categorical review. *2022 17th Annual System of Systems Engineering Conference (SOSE)* (pp. 37-42). New York: IEEE.
- Sachan, V., Deb, P. S., Sharma, S., Thakur, H., & Dubey, G. K. (2020). Product Quality Inspector. *2020 9th International Conference System Modeling and Advancement in Research Trends (SMART)* (pp. 212-215). Moradabad: IEEE

- Saif, Y., Yusof, Y., Ahmed, M. I., Pathan, Z. K., Latif, K., & Kadir, A. Z. A. (2020). A framework to develop intelligent system for capturing product features using OpenCV. *Journal of Technology and Operations Management*, 15(2), 76-85. <https://doi.org/10.32890/jtom2020.15.2.5>
- Shaikh, S., Hujare, D., & Yadav, S. (2022). Surface defect detection using convolutional neural net work model architecture. *Journal of Engineering Research and Sciences*, 1(5), 134–144. <https://doi.org/10.55708/js0105014>
- Stein, F. (2012). The challenge of putting vision algorithms into a car. *Computer Vision and Pattern Recognition, 2012*, 89–94. <https://doi.org/10.1109/CVPRW.2012.6238900>
- Suarez, O. D., Carrobbles, M. F., Enano, N. V., García, G. B., & Gracia, I. S. (2014). *OpenCV essentials*. Birmingham: Packt Publishing.
- Tiufiakov, N. Y., Kalinichev, A. V., Pokhvishcheva, N. V., & Peshkova, M. A. (2021). Digital color analysis for colorimetric signal processing. *Sensors and Actuators B: Chemical*, 343, 130274. <https://doi.org/10.1016/j.snb.2021.130274>
- Varjovi, M. H., Talu, M. F., & Hanbay, K. (2022). Fabric defect detection using customized deep convolutional neural network for circular knitting fabrics. *Türk Doğa ve Fen Dergisi*, 11(3), 160–165. <https://doi.org/10.46810/tdfd.1108264>
- Vishwakarma, S., Akash, C. & Yadav, D. (2015). Analysis of lane detection techniques using OpenCV. *2015 Annual IEEE India Conference (INDICON)* (pp. 1-4). New Delhi: IEEE
- Xiangyu, Z., & Weiting, C. (2013). OpenCV-based video target tracking algorithm. *Journal of Image Processing*, 27(3), 1245-1252. <https://patents.google.com/patent/CN102915545A/en>
- Wei, Y., Zhao, Z., & Gu, K. (2024). The defect detection method for automotive parts based on improved single shot multibox betector using feature enhancement. *2024 2nd International Conference on Signal Processing and Intelligent Computing (SPIC)* (pp. 891-895). Guangzhou, China: IEEE

