

1. หน้าปก (Title Page)

- ชื่อโครงการ: Smart pH Chart อินดิเคเตอร์ธรรมชาติสู่แถบสีดิจิทัล
- ชื่อรายวิชาและภาคการศึกษา: 01159532 ภาคเรียนที่ 2/2024
- นิสิต: ภูมิ ถิ่นกาญจน์วัฒนา (6714650515)
- อาจารย์ผู้สอน: รศ. ดร. พงศ์ประพันธ์ พงษ์โสภณ
- วันที่: 15 มีนาคม 2025 (ตัวอย่าง)

2. บทนำและความเป็นมา (Introduction & Background)

1. บริบทและเหตุผล

การจัดการเรียนรู้ในประเด็นสิ่งแวดล้อมของหลักสูตรประเทศยังคงมุ่งเน้นไปที่ปัญหา สาเหตุ และผลกระทบของปัญหาสิ่งแวดล้อมในระดับสังคมและโลก ซึ่งข้ามปัญหาสิ่งแวดล้อมในระดับโรงเรียนและชุมชนที่นักเรียนเป็นสมาชิกในพื้นที่นั้น อาจส่งผลให้นักเรียนไม่ได้ตระหนักถึงความสำคัญของปัญหา ดังนั้น การจัดการเรียนรู้ในประเด็นสิ่งแวดล้อมควรเริ่มจากปัญหาใกล้ตัวที่นักเรียนสามารถสืบเสาะได้ด้วยตนเอง เพื่อส่งเสริมการเรียนรู้ที่มีความหมายและเห็นคุณค่าของท้องถิ่นของตนเอง นอกจากนี้ กิจกรรมสำรวจปัญหาสิ่งแวดล้อมที่หนังสือและคู่มือครูแนะนำส่วนใหญ่เป็นเพียงการระบุปัญหาด้วยวิธีการพื้นฐาน ทำให้ไม่สามารถระบุสาเหตุของปัญหาที่ชัดเจนแม่นยำ และนำไปสู่แนวทางการแก้ไขปัญหาที่ไม่มีประสิทธิภาพ

ในกระบวนการสำรวจคุณภาพน้ำ การวัดค่าความเป็นกรด-เบส (pH) เป็นตัวบ่งชี้สำคัญที่ช่วยให้นักเรียนเข้าใจภาวะความเป็นกรดหรือเบสของน้ำในแหล่งน้ำรอบตัว อย่างไรก็ตาม วิธีการวัดค่า pH ในโรงเรียนส่วนใหญ่ยังคงอาศัยเครื่องมือพื้นฐาน เช่น กระดาษลิตมัส และ กระดาษยูนิเวอร์ซัลอินดิเคเตอร์ ซึ่งแม้จะใช้งานง่ายและรวดเร็ว แต่ยังมีข้อจำกัดหลายประการ ได้แก่ 1) ความละเอียดต่ำ คือ กระดาษลิตมัสสามารถบอกได้เพียงว่าสารเป็นกรดหรือเบสเท่านั้น โดยไม่สามารถให้ค่าตัวเลขที่แม่นยำได้ 2) ความแม่นยำต่ำ คือ กระดาษยูนิเวอร์ซัลอินดิเคเตอร์ใช้สีเปรียบเทียบกับแถบสีมาตรฐาน ซึ่งอาจเกิดข้อผิดพลาดจากความแตกต่างของแสงและสายตาของผู้ใช้ 3) ไม่สามารถบันทึกและประมวลผลข้อมูลได้ – เมื่อวัดค่า pH ด้วยวิธีเหล่านี้ ผลลัพธ์จะเป็นค่าที่ต้องจดบันทึกเอง ทำให้ยากต่อการนำไปวิเคราะห์เพิ่มเติม นอกจากนี้ การใช้สารอินดิเคเตอร์ธรรมชาติ แม้จะเป็นแนวทางที่เป็นมิตรต่อสิ่งแวดล้อม แต่ก็ยังมีปัญหาเรื่อง การควบคุมสีของอินดิเคเตอร์ ซึ่งอาจเปลี่ยนแปลงได้ตามแหล่งที่มาและสภาพแวดล้อม

แนวทางการจัดการเรียนรู้แบบอิงสถานที่ (Place-based Learning: PBL) เป็นแนวทางที่ช่วยให้การเรียนรู้ด้านสิ่งแวดล้อมเกิดความหมายมากขึ้น โดยมุ่งเน้นให้ผู้เรียนได้เรียนรู้ผ่าน ประสบการณ์ตรงจากสถานที่จริง เช่น แหล่งน้ำในโรงเรียน คลองในชุมชน หรือพื้นที่สีเขียวในท้องถิ่นของตนเอง ซึ่งช่วยให้ผู้เรียนรู้สึกเชื่อมโยงกับสิ่งแวดล้อมในพื้นที่ เข้าใจปัญหาสิ่งแวดล้อมที่เกิดขึ้น ทั้งที่เป็นรูปธรรม (เช่น คุณภาพน้ำที่เสื่อมโทรม) และนามธรรม (เช่น ความตระหนักในการอนุรักษ์ทรัพยากรน้ำของคนในชุมชน) การมีส่วนร่วมในการสำรวจและวิเคราะห์ข้อมูลจริงช่วยให้ผู้เรียนเกิดความภาคภูมิใจในท้องถิ่นของตนเอง และพัฒนาเจตคติของการเป็นพลเมืองที่รับผิดชอบต่อสิ่งแวดล้อม

ระบุ **ความสำคัญ** ของการนำโค้ดและเครื่องมือดิจิทัลมาประยุกต์ใช้ในการสอน โค้ด Python สามารถช่วยวิเคราะห์และแปลงสีของอินดิเคเตอร์ให้เป็นค่า pH ดิจิทัลได้อย่างแม่นยำ โดยใช้ การประมวลผลภาพ (Image Processing) และ OpenCV ซึ่งช่วยลดอิทธิพลของแสงและการมองเห็นของมนุษย์ อีกทั้งการใช้ดิจิทัลช่วยให้นักเรียนสามารถ บันทึกและเปรียบเทียบค่า pH ได้อัตโนมัติ ทำให้เกิดการเรียนรู้เชิงลึกเกี่ยวกับการเปลี่ยนแปลงของค่า pH ในแหล่งน้ำต่าง ๆ นอกจากนี้ การใช้เครื่องมือโปรแกรมมิงยังช่วย ฝึกทักษะการคิดวิเคราะห์ การเขียนโค้ด และการใช้เทคโนโลยีในการแก้ปัญหาทางวิทยาศาสตร์ ซึ่งเป็นทักษะสำคัญในยุคดิจิทัลและการเรียนรู้ STEM Education

2. งานวิจัยหรือแหล่งข้อมูลที่เกี่ยวข้อง (ถ้ามี)

การจัดการเรียนรู้แบบอิงสถานที่ (Place-based Learning) คือ การจัดประสบการณ์การเรียนรู้นอกห้องเรียนที่ผู้สอนทำสถานที่ ชุมชนมาสร้างประสบการณ์ตรงให้กับผู้เรียน โดยมีเป้าหมายให้ผู้เรียนเกิดความตระหนัก รู้สึกเป็นส่วนหนึ่งของท้องถิ่น ซึ่งทำให้ผู้เรียนได้พินิจสายสัมพันธ์ระหว่างตนเองกับสิ่งแวดล้อมในท้องถิ่น ทั้งสิ่งแวดล้อมที่เกิดขึ้นตามธรรมชาติและมนุษย์สร้างขึ้น ทั้งที่เป็นนามธรรมและรูปธรรม ส่งผลให้ผู้เรียนเกิดความภูมิใจ ห่วงแทน เห็นคุณค่า และต้องการที่จะอนุรักษ์ฟื้นฟูสิ่งแวดล้อมในถิ่นฐานนั้น ๆ ให้มีคุณภาพดี และเพื่อพัฒนาให้ผู้เรียนมีคุณลักษณะของการเป็นพลเมืองที่ดี (วิจารณ์, 2556)

3. วัตถุประสงค์และผลลัพธ์การเรียนรู้ (Objectives and Learning Outcomes)

1. วัตถุประสงค์ของโครงการ

- พัฒนาโปรแกรม Python เพื่อให้นักเรียนสร้าง pH Chart และระบุค่า pH ของแหล่งน้ำเพื่อนำไปวิเคราะห์คุณภาพของแหล่งน้ำ

2. ผลลัพธ์การเรียนรู้ (Learning Outcomes)

- ทักษะทางวิทยาศาสตร์และสิ่งแวดล้อม → เข้าใจความสัมพันธ์ระหว่างค่า pH กับสีของอินดิเคเตอร์ธรรมชาติเพื่อใช้ระบุคุณภาพของแหล่งน้ำ
- ทักษะทางคอมพิวเตอร์และการเขียนโปรแกรม → ฝึกใช้ Python, OpenCV และ Machine Learning
- ทักษะทางคณิตศาสตร์และการวิเคราะห์ข้อมูล → คำนวณสี, วิเคราะห์ข้อมูล, และใช้ตรรกศาสตร์
- ทักษะทางทักษะทางปัญญาและการแก้ปัญหา → คิดวิเคราะห์, แก้ปัญหา, และประยุกต์ใช้เทคโนโลยี

4. กลุ่มเป้าหมายและการบูรณาการกับการสอนวิทยาศาสตร์ (Target Learners and Integration with Science Teaching)

1. ระดับชั้นหรือกลุ่มผู้เรียน

- นักเรียนชั้นมัธยมศึกษาปีที่ 3 ที่เรียน เรื่อง ระบบนิเวศ, ปฏิกิริยาเคมี
- นักเรียนชั้นมัธยมศึกษาปีที่ 5 ที่เรียน เรื่อง อินดิเคเตอร์

2. หัวข้อทางวิทยาศาสตร์/สิ่งแวดล้อมที่ครอบคลุม

- ปัญหาคุณภาพของแหล่งน้ำที่ส่งผลต่อสิ่งมีชีวิตและสิ่งแวดล้อม

3. แนวทางทางวิชาการ/แนวปฏิบัติ

การบูรณาการโค้ดวิเคราะห์ pH กับบทเรียน “ปฏิกิริยาเคมีที่ส่งผลต่อสิ่งมีชีวิตและสิ่งแวดล้อม” ผ่านการสอนแบบ Project-Based Learning (PBL)

5. การออกแบบโครงงานและอัลกอริทึม (Project Design and Algorithm)

1. ภาพรวมของโปรแกรม/เครื่องมือ

โปรแกรมนี้ใช้สำหรับ วิเคราะห์ค่า pH ของแหล่งน้ำโดยใช้อินดิเคเตอร์ธรรมชาติ ผ่านการประมวลผลภาพ โปรแกรมสามารถ อ่านภาพตัวอย่างจากแหล่งน้ำ, วิเคราะห์สีของอินดิเคเตอร์, เปรียบเทียบกับ pH Chart ที่สร้างขึ้น, และ แสดงค่า pH โดยอัตโนมัติ โปรแกรมช่วยให้การวิเคราะห์ค่า pH มีความแม่นยำมากขึ้น และลดข้อผิดพลาดที่อาจเกิดจากการประเมินสีด้วยสายตา

2. อัลกอริทึมหรือผังงาน (Flowchart)

ขั้นตอนการทำงานของโปรแกรม:

1. รับข้อมูล – โปรแกรมรับ ภาพตัวอย่างน้ำ ที่ผ่านการทดสอบด้วยอินดิเคเตอร์ธรรมชาติ
2. ปรับปรุงคุณภาพภาพ – ใช้ OpenCV เพื่อ ปรับ White Balance และ Normalize สี เพื่อลดอิทธิพลของแสง
3. แยกค่าเฉลี่ยสี – คำนวณค่า RGB หรือ HSV ของอินดิเคเตอร์
4. เปรียบเทียบกับ pH Chart – ใช้อัลกอริทึม Euclidean Distance เพื่อหาระยะห่างระหว่างสีตัวอย่างกับ pH Chart
5. แสดงผลค่า pH – แสดงค่า pH ที่ใกล้เคียงที่สุด พร้อมแสดงภาพกราฟและข้อมูลวิเคราะห์

3. ฟังก์ชันสำคัญ (Key Functions)

ตัวอย่างฟังก์ชันที่ใช้ในโปรแกรม

```
import cv2
```

```
import numpy as np
```

ฟังก์ชันปรับ White Balance

```
def white_balance(img):
    result = cv2.cvtColor(img, cv2.COLOR_RGB2LAB).astype(np.float32)
    avg_a = np.mean(result[:, :, 1])
    avg_b = np.mean(result[:, :, 2])
    result[:, :, 1] -= (avg_a - 128)
    result[:, :, 2] -= (avg_b - 128)
    result = np.clip(result, 0, 255).astype(np.uint8)
    return cv2.cvtColor(result, cv2.COLOR_LAB2RGB)
```

ฟังก์ชันวิเคราะห์ค่า pH จากภาพ

```
def analyze_ph(image, ph_chart):
    image_color = np.mean(image, axis=(0,1)) # หาค่าเฉลี่ยสี
    closest_ph = min(ph_chart, key=lambda ph: np.linalg.norm(image_color - ph_chart[ph]))
    return closest_ph
```

4. โลบารี่ที่ใช้ (Libraries Used)

- OpenCV (cv2) → ใช้ในการประมวลผลภาพ (White Balance, คำนวณสี RGB/HSV)
- NumPy → ใช้สำหรับคำนวณค่าเฉลี่ยสี และเปรียบเทียบระยะห่างสี
- Matplotlib → ใช้แสดงผลแถบสี pH Chart
- Gradio → ใช้สร้าง UI แบบโต้ตอบ เพื่อให้ผู้ใช้สามารถอัปโหลดภาพและรับค่าพีเอช

6. การพัฒนาโค้ด (Coding Implementation)

1. โค้ด (Code Snippets)

โค้ดตัวอย่างสำหรับ เปรียบเทียบสีของอินดิเคเตอร์กับ pH Chart

```
import cv2
import numpy as np
import gradio as gr

# ตัวอย่างค่า RGB ของ pH Chart (ค่าโดยประมาณ)
ph_chart = {
    1: np.array([186, 0, 125]), # pH 1 สีม่วงแดง
    3: np.array([255, 0, 0]),   # pH 3 สีแดง
    5: np.array([255, 165, 0]), # pH 5 สีส้ม
    7: np.array([0, 255, 0]),   # pH 7 สีเขียว
    9: np.array([0, 255, 255]), # pH 9 สีฟ้า
    11: np.array([0, 0, 255]),  # pH 11 สีน้ำเงิน
    12: np.array([128, 0, 128]) # pH 12 สีม่วง
}

# ฟังก์ชันวิเคราะห์ค่า pH
```

```
def predict_ph(image):
    img = cv2.cvtColor(image, cv2.COLOR_BGR2RGB) # แปลงเป็น RGB
    avg_color = np.mean(img, axis=(0,1)) # คำนวณค่าเฉลี่ยสี
    closest_ph = min(ph_chart, key=lambda ph: np.linalg.norm(avg_color - ph_chart[ph]))
    return f"ค่า pH ที่คาดการณ์คือ: {closest_ph}"

# สร้างอินเทอร์เฟซแบบโต้ตอบด้วย Gradio
demo = gr.Interface(fn=predict_ph, inputs="image", outputs="text")
demo.launch(share=False)
```

2. คำอธิบายขั้นตอนหลัก

1. โปรแกรมรับภาพอินพุตจากผู้ใช้งาน ผ่าน Gradio
2. ใช้ OpenCV แปลงภาพเป็น RGB และคำนวณค่าเฉลี่ยสี
3. เปรียบเทียบกับฐานข้อมูล pH Chart โดยใช้ Euclidean Distance
4. แสดงค่า pH ที่ใกล้เคียงที่สุด

3. การทดสอบและแก้ไข (Testing and Debugging)

1. การทดสอบความถูกต้องของการวิเคราะห์สี
 - ทดลองกับภาพตัวอย่างหลายภาพที่รู้ค่าพีเอชล่วงหน้า
 - ใช้ Matplotlib แสดงแถบสีเพื่อดูว่าสีที่ได้ตรงกับที่คาดไว้หรือไม่
2. ปัญหาที่พบและวิธีแก้ไข
 - ปัญหา: pH Chart มีความคลาดเคลื่อนจากการเลือกรูปที่ใช้เป็นตัวแทนของสีใน pH chart ไม่ดี

วิธีแก้: ใช้รูปของสียินดิเคเตอร์ที่ใช้ระบุค่า pH หลายรูป/ 1 ค่า pH

- ปัญหา: ความแตกต่างของแสงทำให้สียินดิเคเตอร์เปลี่ยนไป

วิธีแก้: ใช้เทคนิค White Balance และ Histogram Matching

- ปัญหา: อินดิเคเตอร์บางตัวมีสีใกล้เคียงกัน ทำให้โปรแกรมแยกไม่ออก

วิธีแก้: เพิ่มการเปรียบเทียบใน Color Space แบบ HSV แทน RGB

7. แผนการนำไปใช้ในชั้นเรียน (Classroom Implementation Plan)

1. การตั้งคำถามและระบุปัญหา

นักเรียนเริ่มต้นด้วยการสำรวจปัญหาสิ่งแวดล้อมในโรงเรียน โดยเน้นที่คุณภาพของแหล่งน้ำผ่านค่า pH ซึ่งเป็นตัวบ่งชี้ภาวะความเป็นกรดหรือเบสของน้ำ นักเรียนจะตั้งคำถาม เช่น น้ำในแหล่งน้ำในโรงเรียนมีค่า pH เท่าไหร่? ค่า pH ส่งผลกระทบท่อสิ่งมีชีวิตในระบบนิเวศอย่างไร? และเราจะใช้เทคโนโลยีช่วยระบุปัญหานี้ได้อย่างไร? เพื่อให้การเรียนรู้มีความหมายและเกี่ยวข้องกับชีวิตจริง

2. การวางแผนและออกแบบการทดลอง

นักเรียนวางแผนการสำรวจและเก็บตัวอย่างน้ำจากแหล่งน้ำในโรงเรียน โดยใช้ อินดิเคเตอร์ธรรมชาติ เช่น ดอกอัญชันหรือกะหล่ำปลีม่วงในการตรวจสอบค่า pH จากนั้น นักเรียนจะวางแผนเปรียบเทียบค่าพีเอชที่ได้จากกระดาษยูนิเวอร์ซัลอินดิเคเตอร์ และการวิเคราะห์ด้วยโค้ด Python และ OpenCV เพื่อลดข้อผิดพลาดจากการมองเห็นของมนุษย์

3. การทดลองและรวบรวมข้อมูล

นักเรียนทำการทดลองจริงโดยหยดอินดิเคเตอร์ธรรมชาติลงในตัวอย่างน้ำและบันทึกการเปลี่ยนแปลงของสี จากนั้นใช้ กล้องถ่ายภาพสีของสารละลาย (นักเรียนต้องใช้กล้องหรือเครื่องมือที่ควบคุมแสงสำหรับถ่ายภาพ) และใช้โค้ด Python วิเคราะห์ค่าเฉลี่ยสีของอินดิเคเตอร์ เพื่อแปลงเป็นค่าพีเอช ข้อมูลที่ได้จะถูกเปรียบเทียบกับค่า pH ที่วัดจาก กระดาษยูนิเวอร์ซัลอินดิเคเตอร์ และกระดาษลิตมัส เพื่อให้ได้ผลลัพธ์ที่แม่นยำยิ่งขึ้น

4. การวิเคราะห์ข้อมูลและสรุปผล

เมื่อนักเรียนได้ค่า pH จากการทดลองแล้ว พวกเขาจะนำข้อมูลมา วิเคราะห์ความสัมพันธ์ระหว่างค่า pH และ ปัญหาสิ่งแวดล้อมในแหล่งน้ำนั้น เช่น น้ำที่มี pH ต่ำกว่า 5 อาจเป็นผลมาจากกิจกรรมของคนในโรงเรียน หรือน้ำที่มี pH สูงอาจเกิดจากการไม่ได้รับการดูแล ซึ่งนักเรียนสามารถใช้ Google sheet และ Pandas เพื่อสร้าง กราฟเปรียบเทียบค่า pH ของแต่ละแหล่งน้ำ เพื่อให้เห็นแนวโน้มของคุณภาพน้ำในโรงเรียน

5. การนำเสนอและสะท้อนคิด

สุดท้าย นักเรียนจะนำเสนอผลการวิจัยของตนเองผ่าน pH Chart ที่สร้างจากโค้ด Python หรือพัฒนาเป็น เว็บไซต์ด้วย Gradio เพื่อให้ผู้อื่นสามารถอัปโหลดภาพของอินดิเคเตอร์และให้โค้ดวิเคราะห์ค่า pH ได้ นักเรียนจะสะท้อนถึง วิธีที่เทคโนโลยีช่วยให้การวิเคราะห์น้ำมีประสิทธิภาพขึ้น และอภิปรายแนวทางการแก้ไขปัญหาที่เป็น กรดหรือด่างในชุมชน ซึ่งช่วยให้พวกเขาพัฒนาทักษะการเป็นพลเมืองที่ใส่ใจสิ่งแวดล้อม

8. ผลลัพธ์และข้อสังเกต (Results and Observations)

1. ความสนใจของนักเรียน (Student Engagement)

การใช้โค้ด Python ในการวิเคราะห์ค่า pH ผ่านการประมวลผลภาพช่วยเพิ่มความสนใจของนักเรียนในการเรียนรู้เกี่ยวกับ ปฏิกิริยาเคมีและสิ่งแวดล้อม นักเรียนพบว่าการใช้ อินดิเคเตอร์ธรรมชาติ เช่น ดอกอัญชันและกะหล่ำปลีม่วง เป็นเรื่องที่น่าสนใจและเกี่ยวข้องกับชีวิตประจำวันมากขึ้น นอกจากนี้ การใช้ Gradio สร้างอินเทอร์เฟซแบบโต้ตอบที่สามารถอัปโหลดภาพเพื่อทำนายค่า pH ได้ทันที ทำให้นักเรียนรู้สึกตื่นเต้นกับการทดลองมากขึ้น อย่างไรก็ตาม นักเรียนบางส่วนอาจพบอุปสรรคในการทำความเข้าใจหลักการทำงานของโค้ด เช่น White Balance, การเปรียบเทียบค่า RGB, และอัลกอริทึม Euclidean Distance ซึ่งต้องใช้เวลาในการอธิบายและฝึกฝน

2. พัฒนาการด้านการเรียน (Learning Gains)

จากการทดลองใช้โค้ดในบทเรียน พบว่านักเรียนสามารถ เข้าใจแนวคิดเรื่องค่า pH, อินดิเคเตอร์ และผลกระทบของความเป็นกรด-เบสต่อสิ่งแวดล้อมได้ดียิ่งขึ้น นอกจากนี้ นักเรียนยังได้พัฒนาทักษะด้าน การคิดเชิงตรรกะ (Logical Thinking) และการเขียนโค้ดเบื้องต้น ผ่านการทำงานกับ Python, OpenCV และ NumPy นักเรียนสามารถ วิเคราะห์และเปรียบเทียบผลการทดลองจากอินดิเคเตอร์ธรรมชาติ กับผลลัพธ์จากการประมวลผลภาพ ทำให้เข้าใจข้อจำกัดของวิธีการวัดค่า pH แบบดั้งเดิมมากขึ้น

ในแง่ของ การพัฒนาทักษะการเขียนโค้ด นักเรียนที่มีพื้นฐานด้านการเขียนโปรแกรมสามารถเข้าใจและปรับแต่งโค้ดได้รวดเร็ว ขณะที่นักเรียนที่ไม่มีพื้นฐานอาจต้องการคำแนะนำเพิ่มเติม อย่างไรก็ตาม เมื่อนักเรียนได้ลองรันโค้ดและเห็นผลลัพธ์จากภาพถ่ายของอินดิเคเตอร์จริง พวกเขาจะเข้าใจหลักการทำงานของโค้ดได้ดีขึ้น

3. อุปสรรคหรือปัญหา (Challenges)

แม้ว่าโค้ดนี้สามารถช่วยเพิ่มความแม่นยำในการวิเคราะห์ค่า pH ได้ แต่ยังมีข้อจำกัดที่อาจพบในการใช้งานจริง เช่น ปัญหาเกี่ยวกับแสงและเงาของภาพ ซึ่งอาจส่งผลให้การตรวจจับสีคลาดเคลื่อนได้ หากไม่มีการควบคุมสภาพแวดล้อมของการถ่ายภาพ ปัญหานี้สามารถแก้ไขได้โดยใช้ White Balance และ Histogram Matching อย่างไรก็ตาม นักเรียนอาจต้องฝึกฝนเพิ่มเติมเกี่ยวกับการถ่ายภาพตัวอย่างน้ำในสภาพแสงที่เหมาะสม

นอกจากนี้ ข้อจำกัดด้านอุปกรณ์และการเชื่อมต่ออินเทอร์เน็ต เป็นอีกปัจจัยที่อาจส่งผลต่อการใช้งานโค้ด นักเรียนบางกลุ่มอาจไม่มีคอมพิวเตอร์ที่รองรับ OpenCV ได้ดี หรือบางโรงเรียนอาจมีอินเทอร์เน็ตที่ไม่เสถียร ทำให้การรันโค้ดผ่าน Google Colab อาจล่าช้า การเตรียมอุปกรณ์และแผนสำรอง เช่น การใช้คอมพิวเตอร์ของโรงเรียน หรือรันโค้ดแบบออฟไลน์ผ่าน Jupyter Notebook จึงเป็นสิ่งที่ควรคำนึงถึงในการนำไปใช้ในห้องเรียน

9. สรุปผลและแนวทางในอนาคต (Conclusion and Future Directions)

แนวทางในการขยายผลและปรับปรุงในอนาคต

1. การพัฒนาเพิ่มเติม

- เพิ่มฟีเจอร์หรือองค์ประกอบใหม่ เช่น การประยุกต์ใช้เทคโนโลยีเสริม (เช่น แบบจำลองเสมือนจริง หรือการจำลองสถานการณ์) เพื่อเพิ่มประสิทธิภาพการเรียนรู้
- ปรับปรุงสื่อการสอนให้ครอบคลุมหัวข้อที่ลึกซึ้งขึ้น หรือเพิ่มกิจกรรมที่เน้นการปฏิบัติจริง เพื่อเสริมสร้างประสบการณ์การเรียนรู้

2. การขยายผล

- ขยายการใช้งานกิจกรรมนี้สู่ระดับชั้นอื่น ๆ หรือวิชาอื่น ๆ ที่มีความเกี่ยวข้อง เช่น นำไปประยุกต์ใช้ในวิชาเคมี, ฟิสิกส์ หรือวิทยาศาสตร์สิ่งแวดล้อม
- นำโครงการไปประยุกต์ใช้ในสถานการณ์จริง (Real-world Data) เช่น การทดลองในชุมชน หรือการเก็บข้อมูลจากสถานที่จริง เพื่อเสริมความเชื่อมโยงระหว่างทฤษฎีกับการปฏิบัติ

3. แนวทางการวิจัยหรือสอนในอนาคต

- พัฒนากิจกรรมให้สอดคล้องกับนโยบายการศึกษาสมัยใหม่ เช่น การเรียนรู้แบบ Personalized Learning หรือการเรียนรู้เชิงปัญหา (Problem-based Learning)
- ศึกษาผลกระทบระยะยาวของกิจกรรมที่มีต่อการพัฒนาทักษะทางวิทยาศาสตร์ เช่น การคิดเชิงระบบ (Systems Thinking) หรือทักษะการแก้ปัญหา (Problem-Solving Skills)
- สำรวจความคิดเห็นของผู้เรียนและครูผู้สอนเพิ่มเติม เพื่อปรับปรุงกิจกรรมให้เหมาะสมกับบริบทการเรียนรู้ที่แตกต่างกัน

10. บรรณานุกรม (References)

วิจารณ์ พานิช. (2556, 26 ตุลาคม). นวัตกรรมสู่การเป็นประเทศแห่งการศึกษา.

<https://www.gotoknow.org/posts/565909>

12. ภาคผนวก (Appendices) (ถ้ามี)

- โค้ดฉบับเต็ม (กรณีไม่ได้ใส่ทั้งหมดใน Section 6)

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
from google.colab import files
import re
from collections import defaultdict

# 📌 กำหนดขนาดภาพมาตรฐาน
IMAGE_SIZE = (200, 200)

# 📌 อัปโหลดภาพ
uploaded = files.upload()
ph_data = defaultdict(list)

# 📌 1. โหลดและแสดงภาพตัวอย่างเพื่อตรวจสอบสี
for filename in uploaded.keys():
    match = re.search(r'pH(\d+)', filename)
    if match:
        ph = int(match.group(1))

        img = cv2.imread(filename)
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB) # **แปลง BGR → RGB**
        img = cv2.resize(img, IMAGE_SIZE) # **ปรับขนาดให้เท่ากัน**

        # 📌 แสดงตัวอย่างภาพ
        plt.figure()
        plt.imshow(img)
        plt.title(f"pH {ph}")
        plt.axis("off")
        plt.show()

        ph_data[ph].append(img)

# 📌 2. คำนวณค่าเฉลี่ยของสี
ph_values = sorted(ph_data.keys())
colors = []

for ph in ph_values:
    img_stack = np.stack(ph_data[ph], axis=0) # **รวมอาร์เรย์ให้มีขนาดเดียวกัน**
    avg_img = np.mean(img_stack, axis=0).astype(np.uint8) # คำนวณค่าเฉลี่ยของภาพทั้งหมด
    avg_color = np.mean(avg_img.reshape(-1, 3), axis=0).astype(int) # คำนวณเฉลี่ยของสีทั้ง
    colors.append(avg_color)
```

```

# 📌 3. แปลงสีเป็น 0-1 สำหรับ Matplotlib
colors = np.array(colors) / 255.0

# 📌 4. สร้างแถบสี pH Chart
fig, ax = plt.subplots(figsize=(10, 2))
for i in range(len(ph_values) - 1):
    ax.barh(0, width=ph_values[i+1] - ph_values[i], left=ph_values[i],
           color=colors[i], edgecolor="black", height=1)

ax.set_xticks(ph_values)
ax.set_xticklabels([f"pH {pH}" for pH in ph_values])
ax.set_yticks([])
ax.set_title("Natural pH Indicator Chart (Corrected Colors)")
ax.set_xlim(min(ph_values), max(ph_values))

plt.show()

```