Flag Classification using YOLOv5

Rushikesh Musale

June 9, 2025

Objective

To detect whether an input image contains the Indonesian flag (red on top, white below) or the Polish flag (white on top, red below) using object detection and color-based classification.

Methodology

1. **Object Detection**: Use a YOLOv5 model to locate the flag region in the input image.
2. **Cropping**: Extract the largest bounding box assumed to contain the flag.
3. **Color Analysis**: Convert the cropped flag image to HSV color space and scan it vertically.
4. **Classification Rule**: If red appears before white from the top, it's Indonesian. If white appears first, it's Polish.

Libraries Used

- `opencv-python` (cv2)
- `numpy`
- `ultralytics` (YOLOv5)

Code Overview

Flag Cropping with YOLOv5

```
def crop_flag(image_path):
    model = YOLO('yolov5s.pt')
    img = cv2.imread(image_path)
    results = model(img)
    for result in results:
        boxes = result.boxes
        if boxes is not None and len(boxes) > 0:
            areas = (boxes.xyxy[:, 2] - boxes.xyxy[:, 0]) * (boxes.xyxy
                [:, 3] - boxes.xyxy[:, 1])
            idx = int(areas.argmax())
            x1, y1, x2, y2 = boxes.xyxy[idx].cpu().numpy().astype(int)
            cropped_flag = img[y1:y2, x1:x2]
            cv2.imwrite("cropped_flag.jpg", cropped_flag)
            return cropped_flag
    cv2.imwrite("cropped_flag.jpg", img)
    return img
```

Listing 1: Crop the largest detected flag using YOLOv5

Color-Based Classification

```
def identify_flag(flag_image):
    hsv = cv2.cvtColor(flag_image, cv2.COLOR_BGR2HSV)
    red_lower1 = np.array([0, 50, 50])
    red_upper1 = np.array([10, 255, 255])
    red_lower2 = np.array([170, 50, 50])
    red_upper2 = np.array([180, 255, 255])
    white_lower = np.array([0, 0, 200])
    white_upper = np.array([180, 50, 255])

    height, _ = hsv.shape[:2]
    stripe_height = max(5, height // 20)
    for y in range(0, height, stripe_height):
        stripe = hsv[y:y + stripe_height]
        red_mask1 = cv2.inRange(stripe, red_lower1, red_upper1)
        red_mask2 = cv2.inRange(stripe, red_lower2, red_upper2)
        red_pixels = cv2.countNonZero(red_mask1 | red_mask2)
        white_mask = cv2.inRange(stripe, white_lower, white_upper)
        white_pixels = cv2.countNonZero(white_mask)
        if red_pixels > white_pixels and red_pixels > 50:
            return "Indonesian Flag             "
        elif white_pixels > red_pixels and white_pixels > 50:
            return "Polish Flag              "
    return "Flag type could not be determined"
```

Listing 2: Scan from top and detect red/white pixels

Main Function

```
def Execute():
    image_path = input("Enter the path to the flag image: ").strip()
    if not os.path.exists(image_path):
        print("Error: File not found.")
        return
    cropped_flag = crop_flag(image_path)
    result = identify_flag(cropped_flag)
    print(f"[RESULT] Detected flag: {result}")
```

Execution

Run the program using:

`python Task3.py`

When prompted, provide the image path. The script will save the cropped flag as `cropped_flag.jpg` and print the classification result.

Conclusion

This project demonstrates combining object detection (YOLOv5) with simple color-based heuristics to differentiate between the Indonesian and Polish flags.