VISUAL RECOGNITION
ASSIGNMENT 1

Name
Subbhashit Mukherjee

Roll Number
MT2023065

CONTENT

NAME	PAGE NUMBER
OBJECTIVE	3
TASK 1A	4
TASK 1B	7
RESULTS AND OBSERVATIONS	9

OBJECTIVE

In this report, my first objective is to develop an algorithm for accurately counting the number of books in an image. Leveraging image processing and computer vision techniques, I aim to create a robust solution capable of handling diverse scenarios, including variations in lighting and book arrangements. Through a comprehensive evaluation on a set of test images, I will assess the algorithm's performance in terms of accuracy, speed, and adaptability. Challenges encountered during the development, such as noise reduction and dealing with differing book orientations, will be thoroughly documented. The ultimate goal is to provide a reliable and efficient book counting algorithm that can find applications in areas like library inventory management and automated cataloging.

My second objective is to implement an algorithm for the detection and removal of shadows in images. This algorithm will utilize techniques such as color thresholding, intensity analysis, or machine learning to identify and eliminate shadows effectively. Evaluation of the algorithm's success will be based on the quality of results, preservation of image details, and its adaptability to diverse lighting conditions. Challenges inherent in shadow removal, such as handling dynamic lighting and different shadow characteristics, will be thoroughly investigated and reported. The developed shadow removal algorithm holds potential applications in image enhancement for various fields, including computer vision, photography, and remote sensing, where accurate representation of scenes without shadows is critical.

TASK 1A

Our primary goal is to quantify the number of books in the image. We thoroughly investigated edge detection methods, focusing on Canny and Hough Transform. Canny, chosen for its noise reduction adaptability, and Hough Transform, excelling in line detection, were selected based on their specific strengths. YOLO is strategically incorporated for efficient object detection, significantly enhancing the precision of book localization and counting accuracy.

1. **Canny edge detection:**

   Canny edge detection was selected for book counting due to its superior noise reduction, gradient-based edge detection, and adaptability to varying lighting conditions. Its accurate edge localization and established performance in computer vision tasks make it a reliable choice. The algorithm's robustness to image noise and variations ensures consistent and precise contour identification, crucial for detecting distinct objects like books in diverse images. Steps followed :

   - **Image Loading**:

     The code starts by loading a color image using the OpenCV library.

   - **Grayscale Conversion**:

     The loaded color image is converted to grayscale, simplifying subsequent processing steps.

   - **Gaussian Blur**:

     Gaussian blur is applied to the grayscale image to reduce noise and accentuate edges.

   - **Thresholding**:

     The blurred image undergoes thresholding to create a binary image, distinguishing edges from the background.

   - **Contour Detection**:

     Canny edge detection is employed to identify contours in the binary image, highlighting potential book boundaries.

   - **Filtering Small Contours**:

     Contours with areas below a specified threshold are filtered out, eliminating potential noise.

   - **Drawing Yellow Borders:**

     Yellow borders are drawn around the significant contours on a copy of the original image, representing book boundaries.

- **Book Counting**:

  The algorithm counts the number of books based on the filtered contours.

2. **Yolo Object detection**:

Utilizing YOLO for book counting was driven by its real-time object detection efficiency, allowing simultaneous identification and localization of multiple books. YOLO's adaptability to various environments, rapid processing speed, and precise object localization contribute to enhanced accuracy in book counting tasks. The strategic integration of YOLO in the methodology improves efficiency, providing a reliable solution for accurate and real-time book counting in diverse image scenarios. Steps followed :

- **Loading YOLO Model:**

  The code loads the YOLO (You Only Look Once) object detection model using pre-trained weights ('yolov3.weights') and configuration file ('yolov3.cfg').

- **Loading Classes:**

  The 'coco.names' file is read to obtain the classes that the YOLO model can detect.

- **Image Processing:**

  Image is loaded and processed to create a blob, ensuring compatibility with the YOLO model.

- **YOLO Forward Pass:**

  The processed image is passed through the YOLO model using the forward method to obtain object detection results.

- **Book Counting Algorithm:**

  The code iterates through the detected objects, determining their class, confidence, and position within the image.

- **Distinguishing Horizontal and Vertical Books:**

  The aspect ratio of each detected object is calculated, and based on a predefined threshold, objects are classified as horizontal or vertical books.

- **Drawing Borders and Counting:**

  Detected books are outlined with green borders for horizontal and red borders for vertical, respectively. Counts for both types of books are updated.

3. **Hough Transform:**

Hough Transform is employed for book counting due to its ability to detect straight lines in images, crucial for identifying book boundaries. This method excels in scenarios where books are aligned horizontally or vertically, enhancing accuracy. Hough Transform's adaptability to different line orientations makes it suitable for diverse book arrangements, contributing to a robust and effective book counting solution. Steps followed :

- **Image Preprocessing:**

  The input image is converted to grayscale to simplify subsequent processing steps.

- **Edge Detection:**

  The Canny edge detection algorithm is applied to identify edges in the grayscale image.

- **Hough Transform for Line Detection:**

  The Hough Transform is utilized to detect lines in the edge-detected image. This step helps identify potential book boundaries.

- **Green Line Visualization:**

  Detected lines are visualized on a black canvas by drawing green lines, aiding in the visualization of the Hough Transform results.

- **Rectangle Identification:**

  Pairs of lines with similar angles are considered potential sides of rectangles. This logic is based on the assumption that books have perpendicular sides.

- **Red Rectangle Visualization:**

  Identified rectangles are drawn with red borders on a copy of the original image, helping visualize the regions where books are detected.

- **Results Display:**

  The images with green lines and red rectangles are displayed to illustrate the outcomes of the Hough Transform-based book detection.

- **Book Counting:**

  The count of identified rectangles corresponds to the estimated number of books in the image. This count is returned as the output of the algorithm.

TASK 1B

Our primary goal here is to enhance image quality by effectively removing shadows through a methodology based on the LAB color space. Extensive exploration and analysis led to the integration of LAB color space for brightness adjustments and shadow removal. The LAB color space was chosen for its ability to separately represent luminance and chromatic information, providing a robust foundation for addressing shadow-related challenges. The algorithm involves brightness modification using LAB space and neighborhood averaging for shadow pixel replacement, ensuring an improved image appearance by mitigating the impact of shadows.

Steps followed :

1. **Enhancing Contrast:**

    - **LAB Color Space Conversion:**

      The input image is converted from the BGR color space to the LAB color space using cv2.cvtColor().

    - **Luminance Channel Extraction:**

      The L channel (luminance) is extracted from the LAB color space.

    - **CLAHE Application:**

      Contrast Limited Adaptive Histogram Equalization (CLAHE) is applied to the L channel using cv2.createCLAHE(). This step enhances the contrast while limiting the effect of extreme values.

    - **L Channel Replacement:**

      The original L channel is replaced with the enhanced one in the LAB color space.

    - **Back to BGR Conversion:**

      The LAB color space with the enhanced L channel is converted back to the BGR color space using cv2.cvtColor().

    - **Displaying Results:**

      The original and enhanced images are displayed using cv2_imshow() in colab.

2. **Removing Shadows Using LAB Color Space:**

    - **LAB Color Space Conversion:**

      The input image is converted from the BGR color space to the LAB color space using cv2.cvtColor().

    - **Luminance Channel Extraction:**

      The L channel (luminance) is extracted from the LAB color space.

    - **Shadow Mask Creation:**

A binary mask is created based on predefined luminance thresholds, identifying regions likely to be shadows.

- **Inverting Shadow Mask:**

  The shadow mask is inverted to obtain a mask for non-shadow regions.

- **Neighborhood Extraction:**

  A neighborhood around each non-shadow pixel is extracted using cv2.boxFilter().

- **Average Luminance Calculation:**

  The average luminance of the neighborhood for each non-shadow pixel is calculated.

- **Shadow Removal:**

  The shadow regions in the LAB color space are replaced with the calculated average luminance.

- **Back to BGR Conversion:**

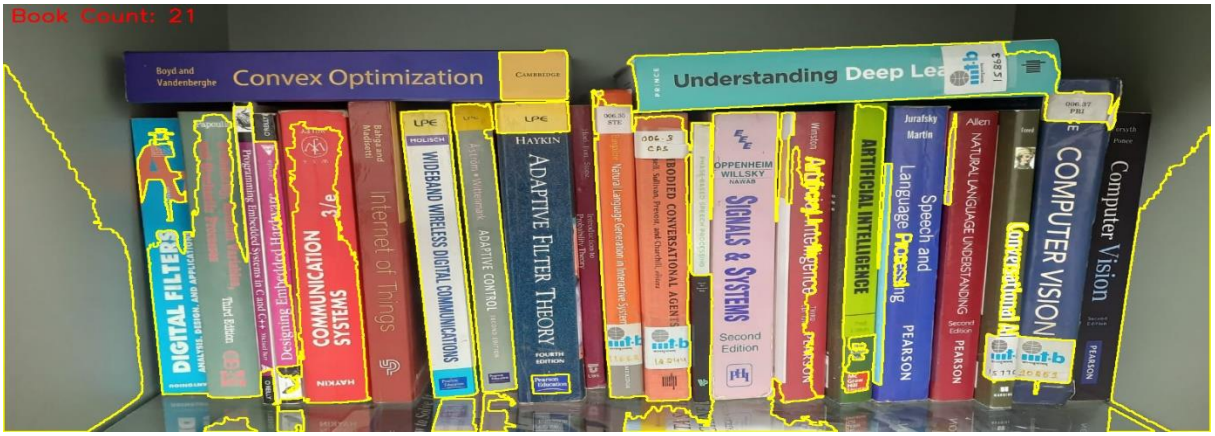
  The resulting LAB color space is converted back to the BGR color space using cv2.cvtColor().

RESULTS AND OBSERVATIONS

Task 1A - Counting Books:

1. **Canny Edge Detection:**

   Canny method showed limitations in accurately detecting books, often including elements below the glass and misinterpreting text as books due to contour irregularities.

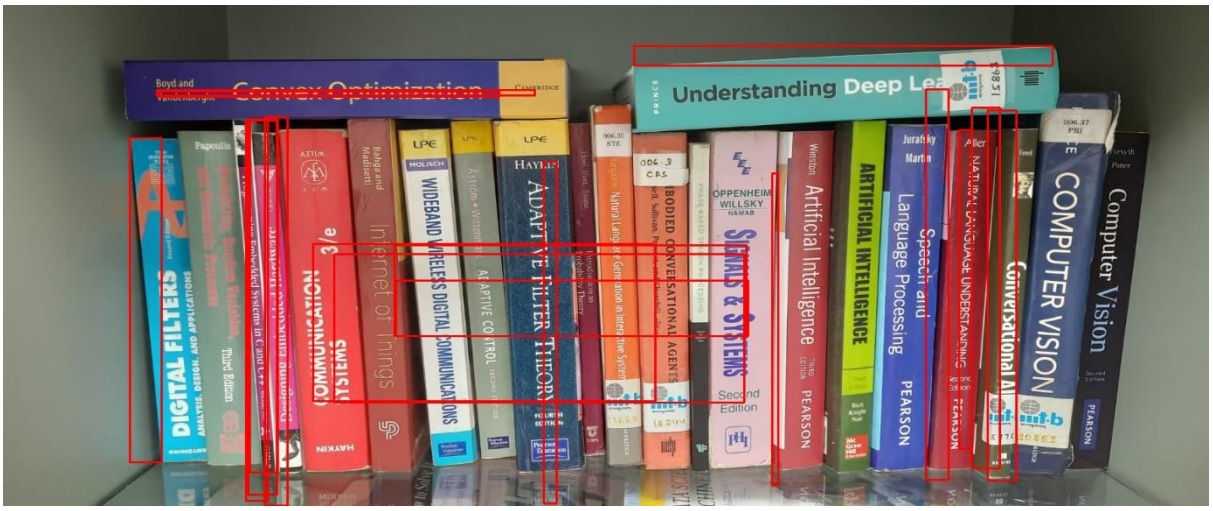


2. **YOLO (You Only Look Once):**

   YOLO demonstrated better performance, although it struggled with horizontally positioned books and tended to count thicker books as multiple entities. Overall, it outperformed Canny in book identification.

3. **Hough Transform:**

   Hough Transform excelled in identifying horizontally positioned books but encountered challenges in detecting all vertically positioned books accurately.

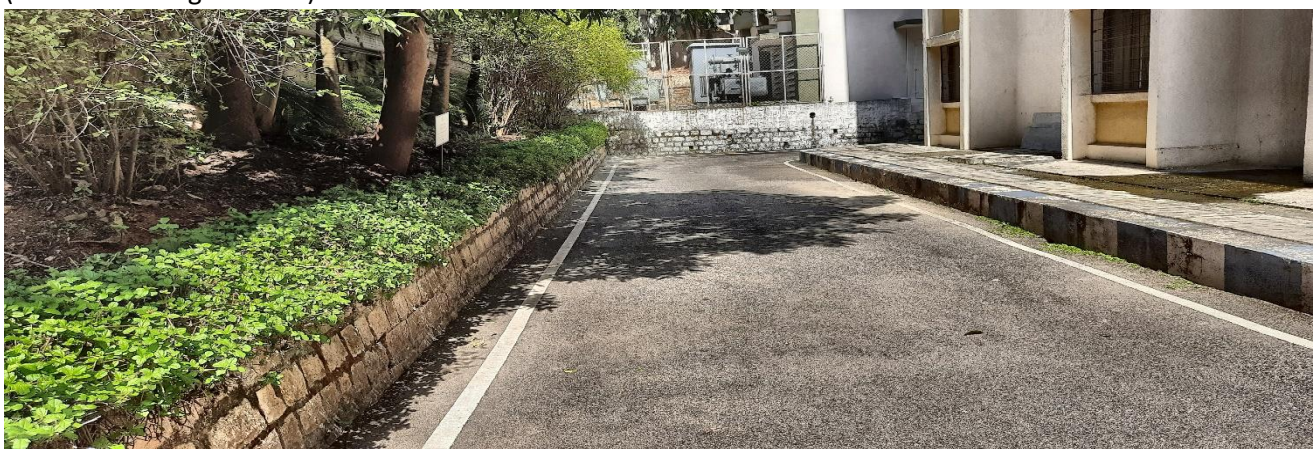


YOLO stands out as the most effective method, but further fine-tuning is required to address issues related to book thickness and horizontal book detection. Combining the strengths of Hough Transform for horizontal books and YOLO for vertical books may yield an improved overall book counting algorithm.

Task 1B - Shadow Removal:

Shadow Mask Generation:

The initial phase produced a robust shadow mask; however, certain regions persisted as shadows even after contrast enhancement and averaging neighboring pixel intensities.
(After increasing contrast)



(inverted shadow mask)

Contrast Enhancement and Averaging:

While increasing image contrast and averaging pixel intensities in shadow regions, some color distortion was observed, affecting the overall visual quality of certain pixels.

Despite challenges, the shadow mask generation proved successful. Future enhancements could focus on refining contrast enhancement and pixel averaging techniques to mitigate color distortions and ensure a more visually appealing outcome.