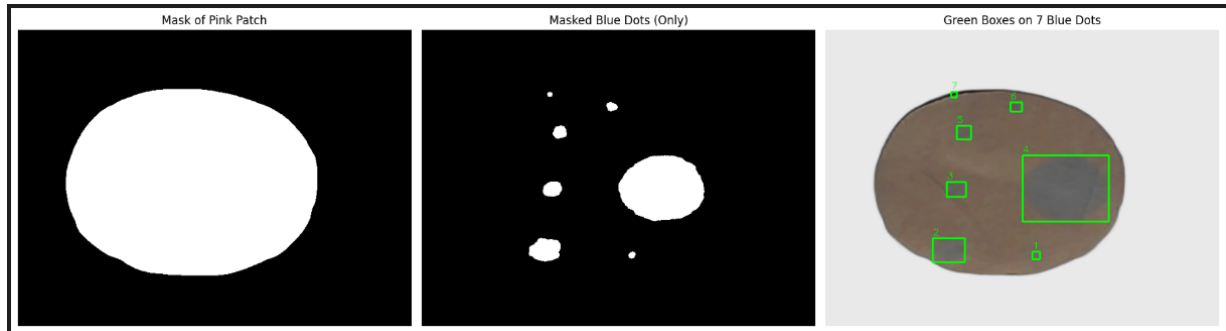# Experiment1

It processes the image to detect blue color changes on a pink paper. It first improves the color contrast using LAB color space and removes the background to focus only on the paper. Then, it finds the pink area using color difference (ΔE) in LAB space. Inside this pink region, it uses KMeans clustering to separate blue spots from the background. The blue spot mask is cleaned using basic image operations, and finally, each spot is highlighted with a green box. The output includes a clean binary mask and an image with all detected spots marked. HSV color space is not used in this method.

Output:

Step	Module	Purpose
1	clean_color_highlight()	Enhances pink-blue contrast using LAB + gamma correction
2	Background Removal	Isolates the pink paper from image background using rembg
3	Pink Patch Detection	Detects main pink area using ΔE thresholding (LAB color distance)
4	Blue Dot Segmentation	Segments non-pink regions (potential blue dots) using KMeans
5	Mask Refinement	Cleans dot mask via morphological ops and flood-fill
6	Dot Localization	Detects and counts blue dots with green boxes on final image

Reasoning behind Each Step

1. Image Enhancement

- The image is adjusted to make pink and blue colours more noticeable.
- Brightness is slightly reduced and contrast is improved.
- **Why?** This helps to clearly see the difference between the pink paper and blue dots.

2. Background Removal

- The background (page lines and handwritten characters) is removed, keeping only the paper.
- Empty areas are filled with white.
- **Why?** So it focuses only on the paper and avoids being confused by things in the background.

3. Pink Patch Detection

- It looks for the main pink area by checking how different each part of the image is from the background color.
- It keeps only the largest pink area and ignores small, unwanted parts.
- **Why?** To make sure it works only on the pink paper and not on shadows or corners.

4. Blue Dot Segmentation

- It looks at the colors inside the pink area.
- It groups colors into 3 clusters: pink background, blue dots, and minor variations.(KMeans Clustering)
- The pink color group is ignored, and the remaining colors (like blue) are kept.
- **Why?** This allows to find blue dots without needing fixed color rules.

5. Mask Cleaning

- The blue dot area is cleaned.
- Any rough edges or extra noise are removed.
- **Why?** To make sure the detected blue areas are smooth, clear, and accurate.

6. Contour Detection and Output

- It finds the edges of each blue spot.
- It removes very tiny dots (which might be noise).
- Boxes are drawn around each blue spot and numbered.
- **Why?** So the result is easy to understand, with each blue dot clearly marked.
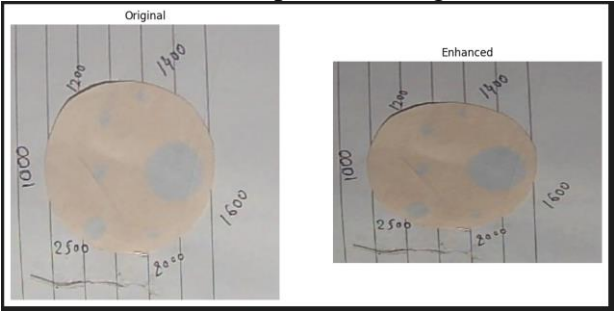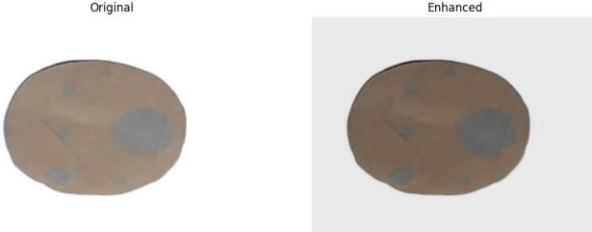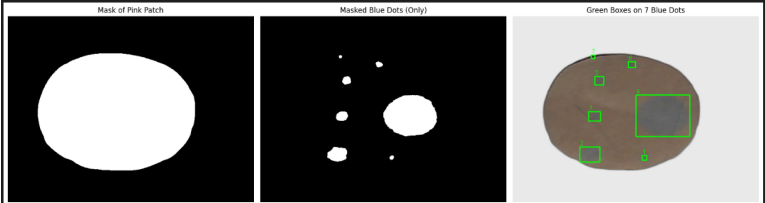
Implementation Details

Tools and Libraries Used

- **OpenCV**: For handling and processing images.
- **rembg**: To remove the background automatically.
- **scikit-learn**: To group colors using KMeans.
- **skimage**: To work with LAB color format.
- **PIL / Matplotlib**: For displaying and saving images.

Main Functions

- clean_color_highlight(): Makes pink and blue easier to tell apart.
- KMeans: Finds different color groups in the pink area.
- cv2.findContours(): Finds the shape and edges of blue dots.
- cv2.rectangle(): Draws green boxes around detected dots.

- **Visual Outputs**

Output File	Description
enhanced_output.png	Contrast-enhanced input (LAB + gamma)
clean_peach_patch_whitebg.png	Foreground-only image with white background
enhanced_output1.png	Image used for blue dot segmentation
Plot 1	Pink Mask (region of interest)
Plot 2	Mask of segmented blue dots
Plot 3	Final annotated image with green boxes and dot counts

Limitation:

It doesn't work well on other similar images because they have uneven lighting, shadows, different paper positions, or faint blue dots. These variations confuse the color clustering and mask generation, leading to poor or incomplete detections. The method relies heavily on consistent color and structure, which limits its performance on varied inputs.