Francisco Cuervo up202302145 | Silvia Tavares up202204392

Assignment 1 | M&M

The quality assurance tester of a M&M factory retrieves a sample from the production every hour to determine some properties of chocolate peanuts. The factory uses a computer vision application to automatically calculate some properties of the sample, namely:

- the distribution of each color in the sample, e.g., the number of red (or blue or …) chocolates. At the end, if the sample does not have chocolate peanuts of all colors, the system must raise an alarm by printing "Color *XPTO* is missing".
- the average shape area of all chocolate peanuts in millimeters and organized by color.

Considering these requirements:

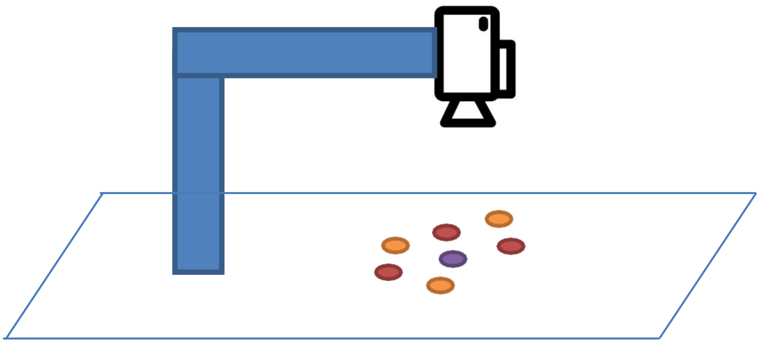

Figure 1 - setup of the imaging system.

a) Calibrate the intrinsic parameters and lens distortion of the camera. Show the intrinsic matrix and the lens distortion coefficients.

Camera calibration is a fundamental task in computer crucial in various applications such as 3D reconstruction, object tracking, augmented reality, and image analysis. Accurate calibration ensures precise measurements and reliable analysis by correcting distortions and estimating intrinsic and extrinsic camera parameters. A camera is a device that converts the 3D world into a 2D image. A camera plays a very important role in capturing three-dimensional images and storing them in two-dimensional images.

Intrinsic or Internal Parameters:

- Calibrating the intrinsic parameters of a camera refers to the process of determining the internal characteristics of the camera that affect the imaging process. Calibrating the intrinsic parameters allows for more accurate measurements and calculations in computer vision applications, such as 3D reconstruction, object tracking, and augmented reality.

The code developed for this step aims to collect image points (2D) and corresponding object points (3D) from images of a chessboard pattern. These points will be used to perform camera calibration.

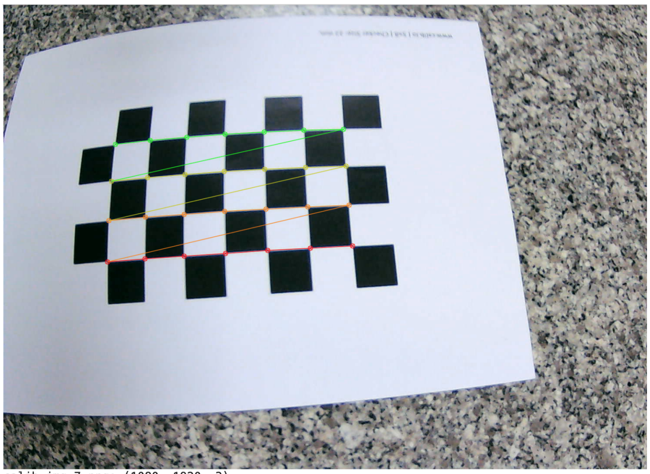

Fig 2: Intrinsic Calibration for image 7

Results:

```
Number of valid images: 36
Intrinsic Matrix:
[[1.32164961e+03 0.00000000e+00 9.88299668e+02]
 [0.00000000e+00 1.32444624e+03 6.42259078e+02]
 [0.00000000e+00 0.00000000e+00 1.00000000e+00]]

Distortion Coefficients:
[[-3.44269790e-01  9.00566070e-02  9.45585473e-05 -3.97940160e-04
  -3.15658830e-03]]
```

Interpretation:

The intrinsic matrix is represented as a 3x3 matrix and contains the following parameters:

- Focal Length (fx, fy): The first two diagonal elements of the matrix (top-left and middle element) represent the focal length of the camera. In this case, the focal length in the x-direction (horizontal) is approximately 1321.65, and in the y-direction (vertical), it's approximately 1324.45.
- Principal Point (cx, cy): The values at the third column and third row (bottom-left and bottom-middle) are the principal point coordinates. This point represents the center of the image. In this case, the principal point is approximately (988.30, 642.26).
- Skew (0): The intrinsic matrix usually has a skew factor (0 in this case), which

indicates that the axes are perpendicular.

- Aspect Ratio (1): The aspect ratio (1 in this case) is the ratio of the horizontal and vertical pixel sizes. An aspect ratio of 1 means that the pixels are square.

Distortion Coefficients: The distortion coefficients are represented as a 1x5 vector. The coefficients correspond to radial and tangential lens distortion effects:

- k1, k2, and k3: These are the radial distortion coefficients. In this case, k1 is approximately -0.344, k2 is approximately 0.0901, and k3 is a very small value (close to zero).
- p1 and p2: These are the tangential distortion coefficients. In this case, p1 is approximately -0.0004, and p2 is approximately -0.0032.

b) Calibrate the extrinsic parameters of the camera setup (or instead, use the extrinsic image to manually measure the dimensions of a black square of the chessboard in pixels. In this way, you will know the conversion between pixel to millimeter). *Show the extrinsic matrix (rotation, translation, etc) and/or the conversion ratio between pixel to millimeter that was used*.

Extrinsic or External Parameters describe the orientation and location of the camera. describe the spatial relationship between the camera and the 3D world. These parameters define the camera's position and orientation in the real world, allowing us to transform 3D world coordinates into 2D image coordinates and vice versa. This refers to the rotation and translation of the camera with respect to some world coordinate system.

For this step we used “extrinsic.png” image located in the folder “WhiteBackground” to calculate the size of individual squares in the pattern, estimate a conversion rate from pixels to square millimeters, and compare it to a measurement made in GIMP a conversion rate measured in GIMP (GNU Image Manipulation Program for converting pixel measurements to square millimeters).

Before performing extrinsic calibration for a camera, we undistort the image. This step is important because undistorting the image helps improve the accuracy and reliability of the calibration process.

After, we estimate a conversion rate from pixels to square millimeters per pixel by dividing the total area of the chessboard pattern (8x21 squares in the x-direction and 5x21 squares in the y-direction) by the total number of pixels in the bounding box.

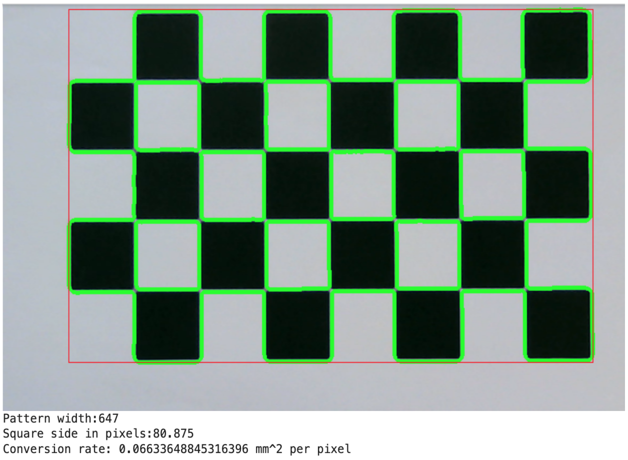

Fig 3: Extrinsic Calibration

Results:

```
# measured in GIMP

Pattern width:647
Square side in pixels:80.875
Conversion rate: 0.06633648845316396 mm^2 per pixel
Conversion rate (GIMP): 0.07066175292421087 mm^2 per pixel
```

The pattern width is calculated to be 647 pixels. This represents the width of the of the chessboard in pixels. The size of each square in the pattern to be approximately 80.875 pixels. This value indicates that, on average, each side of a square in the chessboard pattern is about 80.875 pixels long. This measurement helps to understand the scale of the pattern in terms of pixel units. The conversion rate is calculated as approximately 0.06633648845316396 square millimeters per pixel. This rate represents how many square millimeters in the real world correspond to each pixel in the image. It's calculated by dividing the total area of the chessboard pattern (in square millimeters) by the total number of pixels in the pattern. The result means that for each pixel in the image, it corresponds to roughly 0.06634 square millimeters of the real-world object being measured

The conversion rate (GIMP) is approximately 0.07066175292421087 square millimeters per pixel. This means that for every pixel in the image, it corresponds to roughly 0.07066 square millimeters of the physical object being measured. This conversion rate can be used to relate pixel measurements to real-world dimensions in applications where you need to perform measurements or calculations based on images taken in GIMP with known physical dimensions.

c) Consider only the images retrieved from a **White Background**. Implement the functions described above.

I. Define a ROI (region of interest) for the image;

```
Original Image Shape: (135, 240, 3) ROI Shape: (50, 50, 3)
```

ROI ("Region of Interest") refers to a specific area or region within an image that is of particular interest for analysis or manipulation. This region is typically a subregion of the overall image and is defined based on specific criteria or requirements for a particular task.

Results Interpretation:
"Original Image Shape: (135, 240, 3)" means that the original image has the following characteristics:

- Width: 240 pixels
- Height: 135 pixels
- Color Channels: 3 (RGB, where each channel represents a different color component)

"ROI Shape: (50, 50, 3)" means that a region of interest (ROI) within the original image has the following characteristics:

- Width: 50 pixels
- Height: 50 pixels
- Color Channels: 3 (again, representing the colors in the ROI)

II. Calculate the number of M&M per color for all images provided;

Image:	Number of red M&Ms:	Number of yellow M&Ms:	Number of blue M&Ms:	Number of green M&Ms:
calib_img 2 .png	5	5	5	0
calib_img 3 .png	5	5	5	3
calib_img 1 .png	0	5	5	0
calib_img 0 .png	0	0	0	0
extrincict.png	0	0	0	0

Table 1. Count of M&M per color

Fig 4. Result for *"calib_img 3.png"*

III. Determine the **average area** for peanuts in pixels, considering the image "*calib_img 3*";

   i. Show all peanuts that were detected.

To perform this, we developed a code that has a function "count_peanuts_color" that counts peanuts of a specific color in an image. It takes the input image and the HSV color range for a specific peanut color and performs the following steps:

- Converts the image to the HSV color space.
- Creates a binary mask by applying the specified color range.
- Finds contours in the mask, which correspond to regions of the specified color.
- Filters the detected contours by minimum area.
- Returns the filtered peanut contours.

It proceeds to detect peanuts in the image for each of the specified colors. The results are stored in the detected_peanuts dictionary, where each color corresponds to a list of detected peanut contours.

The code goes on to visualize the detected peanuts. For each color, it iterates over the detected peanut contours and performs the following steps:

- Fits an ellipse around each peanut contour and draws it on a copy of the original image (image_with_ovals).
- Calculates the area of the peanut contour and displays it above the peanut with green text.
- Determines the shape of the peanut (either a circle, triangle, or rectangle) based on the number of corners and displays it with green text.
- Prints information about the detected peanuts, including their area and shape.

Finally, it displays the image with detected peanuts and contour markings that include shape and area using green ovals and text.

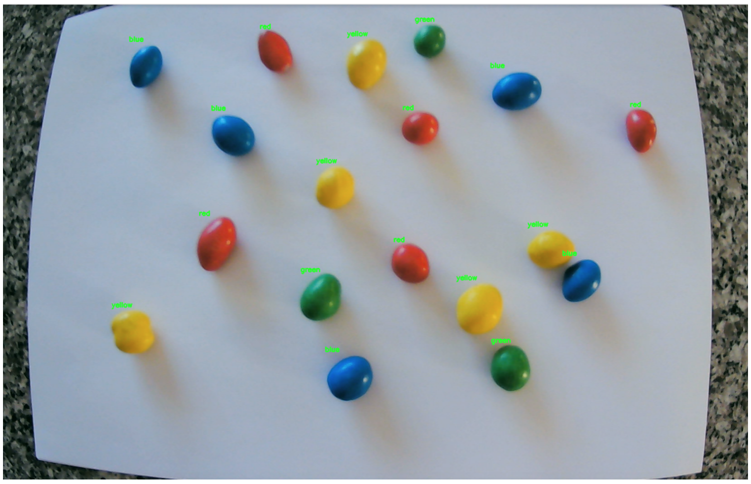

Fig 5. Result for peanuts detected *"calib_img 3.png"*

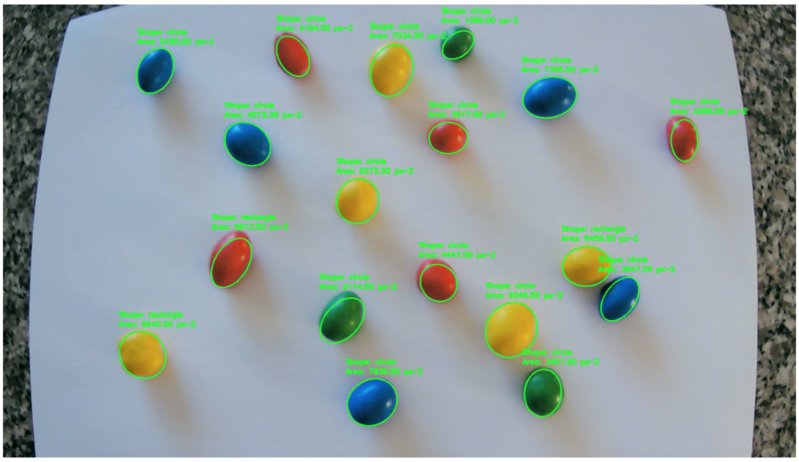

Fig 6. Result for average area for peanuts in pixels for "calib_img 3.png"

ii. Discuss the limitations of the peanut detection method used (in 2 sentences).

The peanut detection method used in this code is based on color and shape has some limitations:

- Color Variation: It heavily relies on color information in the HSV color space. If the peanuts have subtle color variations or are affected by lighting conditions, it may lead to false positives or missed detections.
- Minimum Area Filtering and Real-Time Processing: The code filters contours based on a minimum area (min_area = 100) to remove small contours. This threshold may not be suitable for all images and peanut sizes. Large peanuts might be incorrectly filtered out, or small objects unrelated to peanuts could be detected. Also, it designed for offline image processing. If we intend to use it in a real-time system, it may not be efficient enough.

IV. Determine the **average area** (and **standard deviation**) for peanuts in millimeters and grouped by color, considering images “*calib_img 2*” and “*calib_img 3*”.

i. Show the result in a table color vs area and standard deviation.

	Color	Average Area (mm^2) - Calib 2	Standard Deviation (mm^2) - Calib 2
0	red	47.04875	7.657905
1	yellow	73.46400	10.520580
2	blue	58.43700	11.111377
3	green	NaN	NaN

	Color	Average Area (mm^2) - Calib 3	Standard Deviation (mm^2) - Calib 3
0	red	45.30625	7.906305
1	yellow	71.90600	10.668934
2	blue	58.42000	15.234593
3	green	25.55000	10.462534

Fig 7. Result for average area and standard deviation for peanuts

ii. Discuss the limitation of the method implemented (in 2 sentences).

The implemented method relies on contour detection to calculate peanut areas, which might be sensitive to varying lighting conditions and peanut orientations, leading to inconsistent results. Additionally, the method does not account for overlapping peanuts, potentially underestimating the actual peanut count and area.

V. Provide some recommendations that the M&M factory should take into consideration to improve the performance of the quality assurance process based on image processing (e.g., imaging setup, calibration process and photometric effects), in 4 sentences.

To enhance the quality assurance process, the M&M factory should invest in a consistent and controlled imaging setup, ensuring uniform lighting conditions and minimal shadows. Implementing a rigorous calibration process, accounting for camera intrinsic parameters and lens distortion, is crucial for accurate measurements. Additionally, considering photometric effects such as reflections and surface textures during color detection can refine the algorithm's precision. Regular maintenance of the imaging equipment and continuous monitoring of image quality will ensure sustained accuracy in peanut inspection, allowing for timely adjustments and improvements in the production line.

d) Consider the images retrieved from a **Grey Background** and repeat the previous point without defining a region of interest.

II. Calculate the number of M&M per color for all images provided;

Results:

	red	yellow	blue	green	brown
calib_img0.png	0	0	0	0	0
calib_img1.png	0	5	7	0	0
calib_img2.png	0	5	7	3	0
calib_img3.png	5	5	7	3	5
calib_img4.png	5	5	7	3	5
calib_img5.png	5	4	5	2	5
extrincict.png	0	0	0	0	0

The obtained results were not so accurate for some images, although we try a lot of combination for the HSV range, we couldn't achieve the results that we wanted. By adjusting the HSV range and also the minimum area if needed, better results can be achieved.

III. Determine the average area for peanuts in pixels, considering the image "calib_img 3";

i. Show all peanuts that were detected.

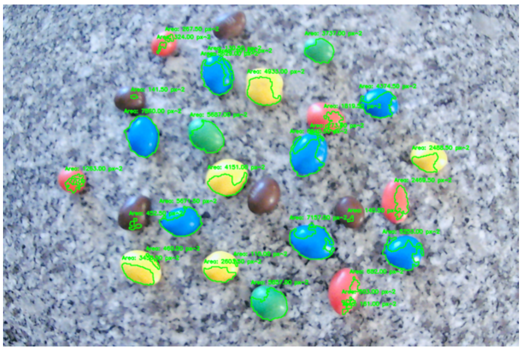

Fig 8. Result of area detected for peanuts

ii. Discuss the limitations of the peanut detection method used (in 2 sentences).

The detection method relies on the color information in the image to segment and identify the peanuts based on their color. The detected contours are used to determine the number of peanuts of each color and other relevant information, such as their areas.

This method is useful for scenarios where the color of objects is a distinguishing feature, but it did not work well because the lighting conditions or object appearances vary significantly.

IV. Determine the average area (and standard deviation) for peanuts in millimeters and grouped by color, considering images "calib_img 2" and "calib_img 3".

i. Show the result in a table color vs area and standard deviation.

File	red_avg_area	red_std	yellow_avg_area	yellow_std
calib_img 4.png	406.8125	305.5845	2599.4285	1663.1782
calib_img 3.png	2627.9166	1177.1999	6358.6	1055.1181

File	green_avg	green_std	blue_avg	blue_std	brown_avg	brown_std
calib_img 4.png	5093.6666	961.8154	4269.0	2835.2434	452.5	0.0
calib_img 3.png	6139.1666	1405.489	4429.2	2912.1470	217.5	96.5

ii. Discuss the limitation of the method implemented (in 2 sentences).

In scenarios with complex lighting conditions or variations in peanut appearance we faced some limitations, as it solely depends on color information and doesn't consider shape or texture characteristics. Additionally, it requires manual adjustment of color ranges for different lighting conditions.