Computer Vision & Image Processing project of Visual Inspection of Motorcycle Connecting Rods

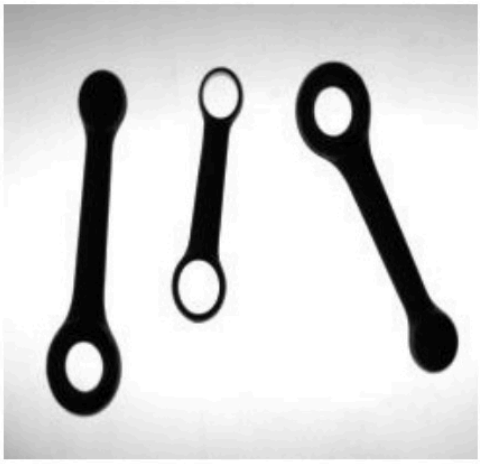
Prof: Di Stefano luigi

Xiaotian Fan
Matriculation Number: 0000973864

Introduction

In this project, I've developed a software system aimed at visual inspection of motorcycle connecting rods. The system is able to analyse the dimensions of two different types of connecting rod to allow a vision-guided robot to pick and for rods based on their type and dimensions. The two rod types are characterised by a different number of holes: Type A and Type B. Type A rods have one hole whilst type B rods have two holes.

The exemplar working rods in the figures are presented as below:



To be more precisely, I'm asked to tell which one is type A, and which is type B. And also in task two, there are some other objects showed in the image, thus, I'm asked to filter the blobs.

Notions Used

At this project, here are the notions are used:

1. Gray scale
2. Binary Image (OTSH)
3. Blob Analysis
4. Morphology Operation

Libraries used:

At this project, here are the libraries used:

1. Opencv2: the main library uses to work with images

2. numpy: the library used to take images as array

3. Matplotlib: the library used to plot images.

4. Skimage: a.k.a scikit-image, a collection of algorithms for image processing and computer vision.

5. Os: module provides a portable way of using operating system dependent functionality.

6.pandas: the library used to manipulate the properties of blobs.

Stages:

This project is divided into two different parts, part one is made to solve the task first, and the part two is made to solve the task second:

For the first task :

1. Importing the libraries and methods.
2. Load the images
3. Defining the functions:

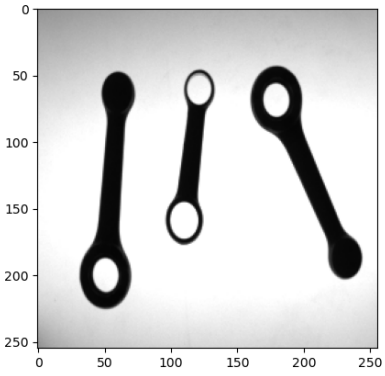
a. distance(p1,p2): get the euclidean distance of the center and contour point.

b. width_at_barycenter(center, contour): calculate the width at a given centroid.

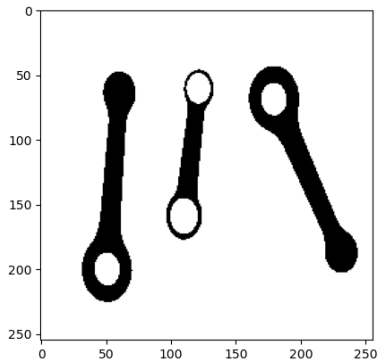
c. task_1(image): Given an image, perform grayscale processing and binary processing on the image to facilitate labeling of connected regions. Use the obtained Euler number to determine the type of rods.

4. Image pre-processing: in this process, the images got processed as below:

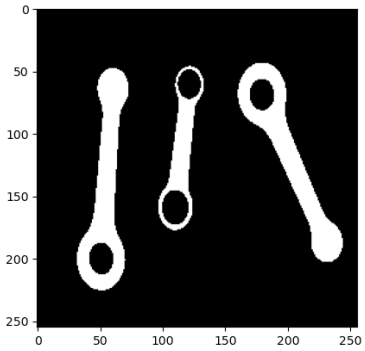
The original image



The binarized image with the method cv2.cvtColor() and thresh_otsu

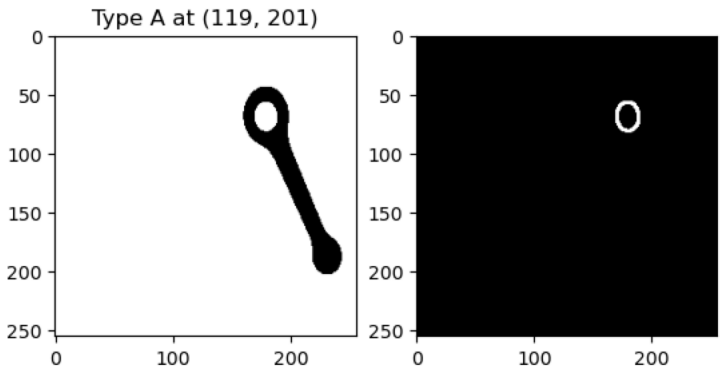


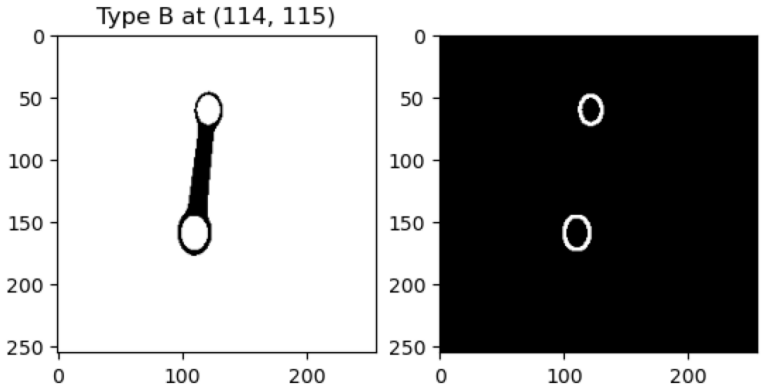
Inverted_binarized_image with the method cv2.bitwise_not()

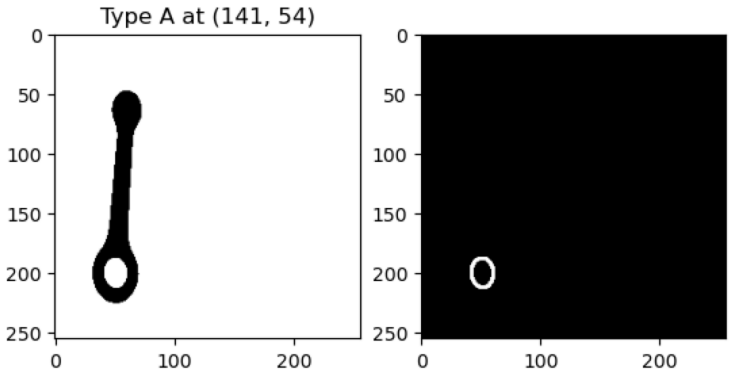


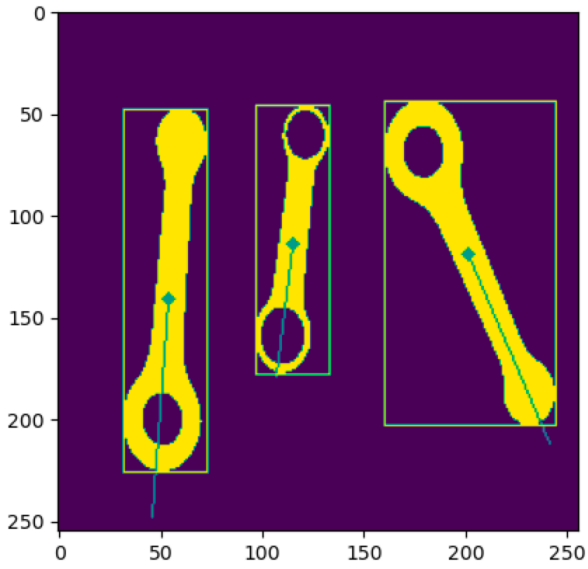
In this step, labelling connected region on the inverted binary image and extract features from the connected region.

5.After the image processed, for each connected region, place it in a separate image and calculate the width at the centroid of the connected region. Also, obtain the position of the deteriorated diameter of the hole. Draw the major axis and centroid position of the connected region, and then draw a bounding box around it. And also get the hole position by using the mehhod cv2.fitEllipse().



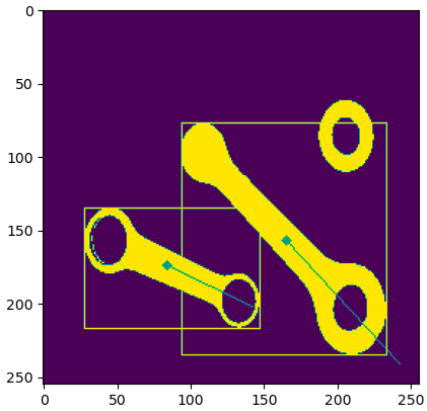






For the second task, followed the stages from the first task with one change:

I filtered out the interfering connected regions by processing ‘area’. Only the connected regions with an area larger than 1100 are reserved (even 800 or 1000 do not satisfied the need, which depends on the area of distractors). After applying the method from Task1 to process the image, got the classification on the rods.



Final discussion

In this project, for task one, the codes work well. It first return the type of each rod, then by separating them into different image plate in order to get the values that we need easily. For the second task, it's kinda tricky to use area as the way to flirt the distractors. This way probably won't work well on images other than the given, cause once the area of distractors are close to the rods, then it is hard to do the distinguish.

Reference:

https://scikit-image.org/docs/stable/api/skimage.measure.html

https://opencv24-python-tutorials.readthedocs.io/en/latest/py_tutorials/py_tutorials.html