2016-2017

Project: Object recognition for coins calculations

XU Sixiang

SUN Yunyun

Professeur : Mr ELAWADY

TSE FISE-2

Parcours 7

télécom saint-étienne
école d'ingénieurs nouvelles technologies

UNIVERSITÉ JEAN MONNET SAINT-ÉTIENNE

École associée INSTITUT Mines-Télécom

Contents :

1. Introduction :

In this project, we are supposed to complete an application Matlab about the detection of coins. the performance of our application should include the detection of the different coins in an image. Consequently, we need a dataset to test the performance of this application.More specifically ,the images to be detected should have a white background. Being limited in the size, these images should be taken with a cellphone or other device whose resolution is appropriate. we chose a smart phone as the image-taking device and obtained a series of images with the same shooting condition. The settings of image-taking are uniform: the same height, he approximately same natural illumination, and same position of the smart phone. the same condition could avoid the “zoom” phenomenon in every image. to have a white ground, we put the coins on a piece of white A4 paper.

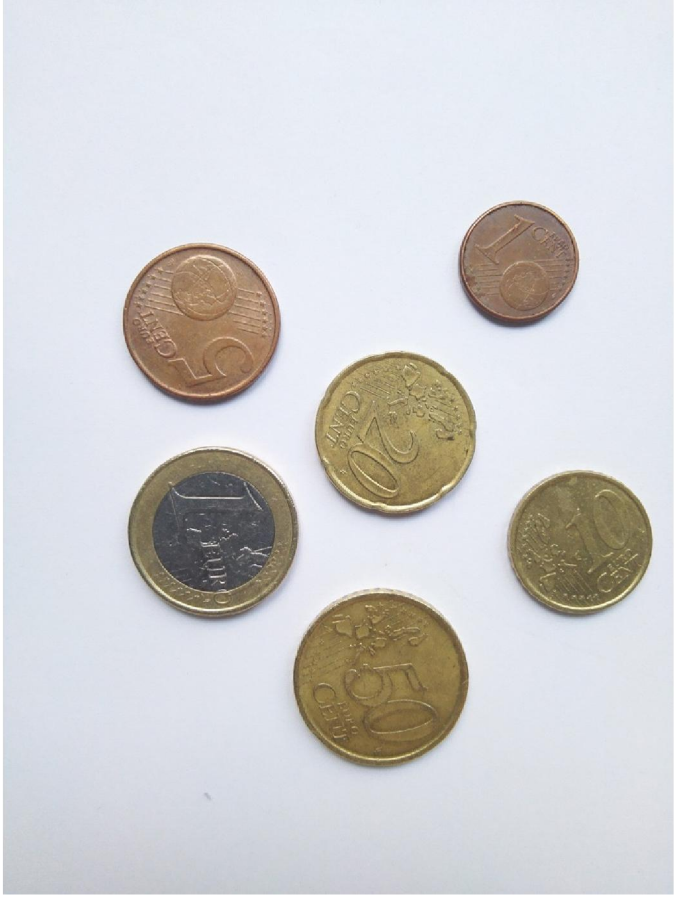


Figure 1a : an image in our dataset

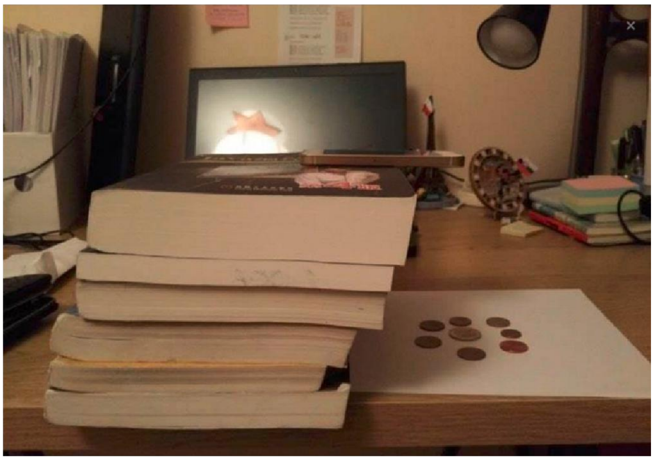
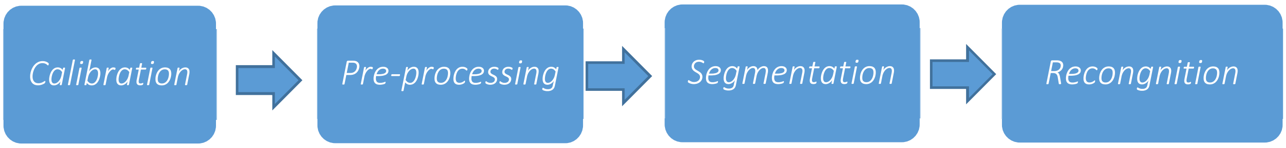


Figure 1b : condition of building dataset

2. Méthodologie

the chain of processing illustrated below indicates the main stapes :


Calibration

Pre-processing

Segmentation

Recongnition

Calibration :

The calibrition permits the application to figure out the colour values(RGB) in different type of the coins which are in our dataset. Similarly, we should be informed about the scalling factor as the multiplication facteur,who can transform the diameter value between pixel and millimeter. Being essential for the recongnition of different coins, the diameter should be well calculated, so should the calibration.

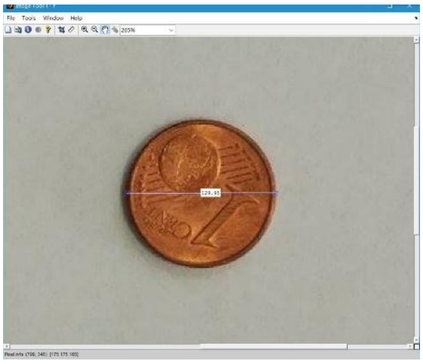


FIGURE 2A : MESURE OF DIAMETER

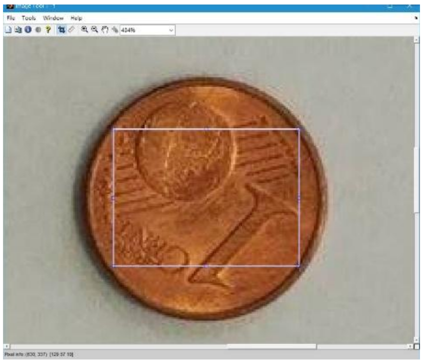


FIGURE 2B : MESURE OF THE COLOR BRONZE

Pré-processing :

In order to obtain a better segmentation, it is elementary for the images to be pre-processed before being segmented directly. It is common that we apply a filter. We found the medfilter very competent in this dataset. We transformed the image from RGB colour space to HSV space. And the case became complicated because the qualities of images transformed varied in different channels in HSV space. When it comes with 2 euro coins, the 3rd channel performed better, meanwhile the saturation channel fonctions better for the other coins. So we seperated the transformation into the 2^{nd} and 3rd channel for the better qualities. In order to being adapted , we reunited the results after.

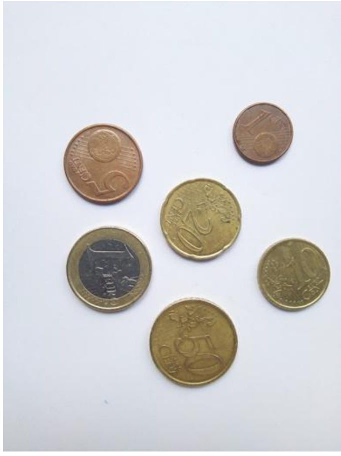


Figure 3a : original Image

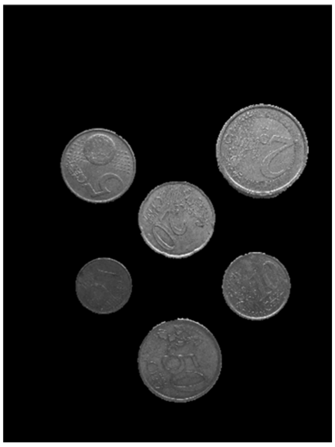


Figure 3b : Image HSV

Segmentation :

We segmented automatically the images without 2 euros into 2 different classes(BW). Segmenting the case containing 2 euros,we adopted a series od process. With images transformed to value sapce, first we detect the noise-ratio in order to substract the unwanted noise. Successively, calculation about the gradients can provide us with images binairy,who has already been segmented in forms. Here comes the post-processing who should optimize the results. With the need of smooth boundary, we apply a set of morphological filters with efficient operation. All the process above gaurentee the quality of our segmentation.

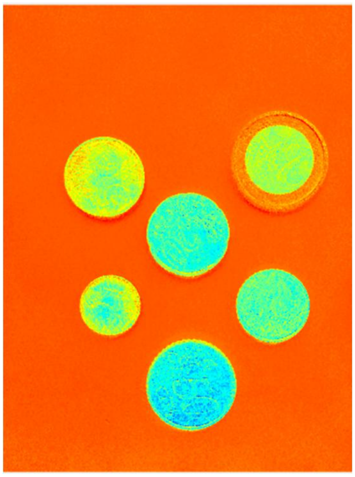


Figure 4a : Inoise-ratio

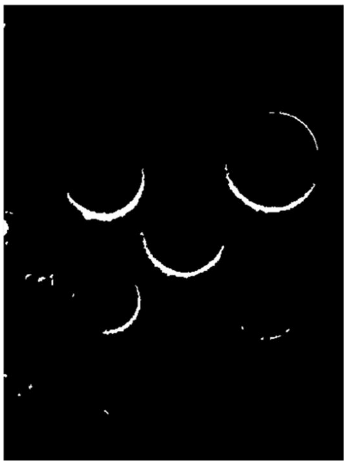


Figure 4b : noise detected

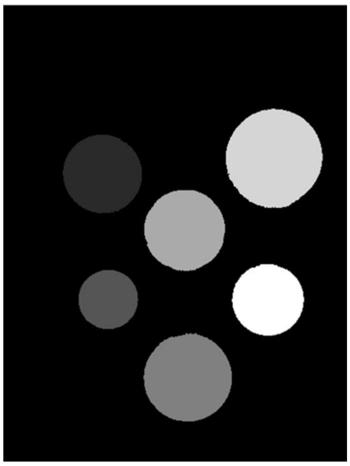


Figure 4c : mask with noise substracted

- **Recongnition :**

We distributed a label for every object segmented. With the help of Matlab functions, we can get the information of equivelent diameter,pixel list,centroide about every object in te image. Analysing the information gives us the crutial values which chould be used for the classification. Everytime we came acrossed a failed case, we updated the classification conditions after analysing the critical values. And we could also use the colour values when the values above are not adequate. We have alse created a vector variable named 'wallet' to verify the accuracy of our segmentation.

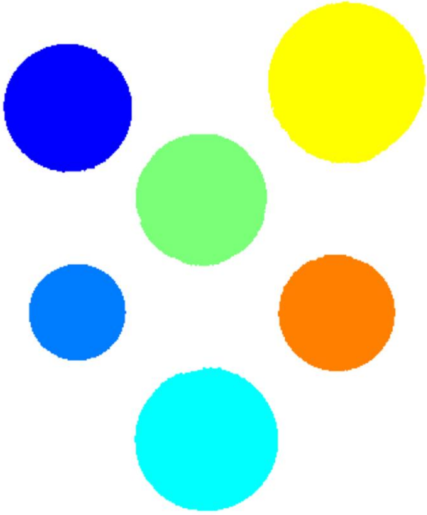


Figure 5a : Image with labels

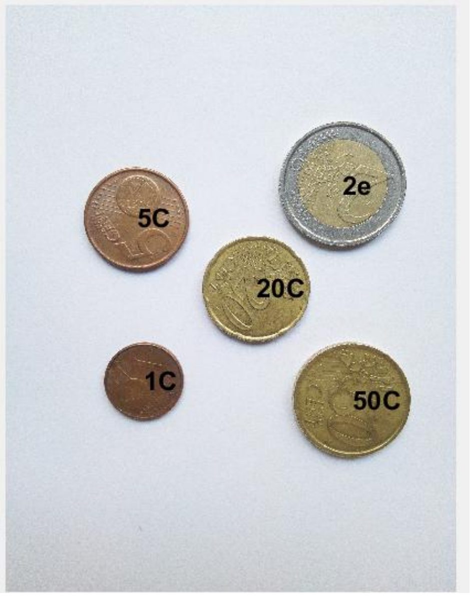


Figure 5b : Image après reconnaissance

3. Project Analysis

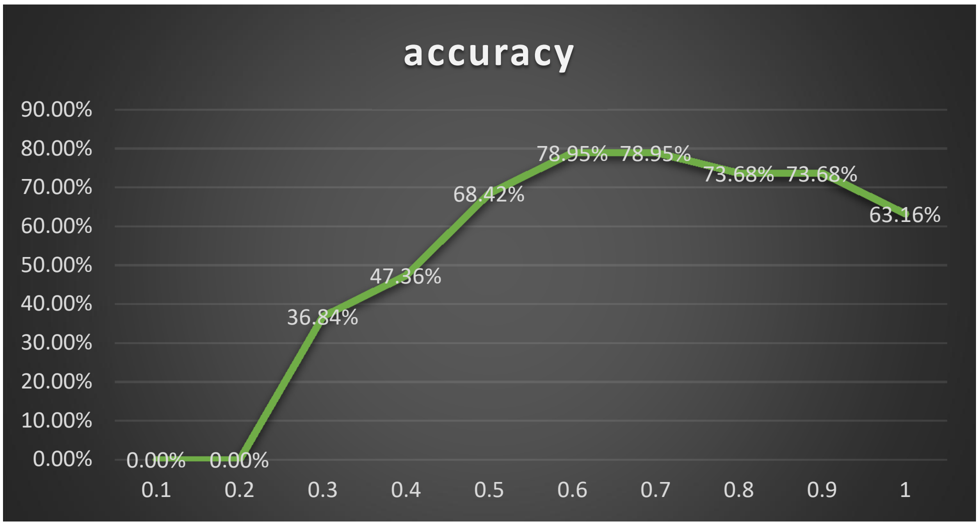
- **Quantitative :**

Nom en lettres	N° Success	N° Fail	Accuracy
Our dataset	61	0	100 %
Boisard_May_A1	38	16	70.4 %

- **Error analysis :**

1. We use the different diameters to distinguish different types of coins.However,in the dataset given by teacher,we find that the diameters of 50 cent,2euro aren’t stable.In some pictures,50 cent is bigger than 2 euro but the sizes of 50 cent and 2 euro become normal in other pictures.Therefore,our programme sometimes makes mistakes when recognizing coins containing 50 cent or 2 euro.
2. The scale factor given in the dataset isn’t correct.By our mesure,it is 0.019.

- **Analyse about bruit**



We add gaussian noise and change the intensity of light which is controlled by the value ‘r’ to simulate the effect of a picture taken in the negative circumstance to study the durability of our programme.

From the graph,we find that it is impossible to recognize the coins when r is low (0.1 or 0.2)because the image is too ’bright’.The maximum of accuaracy happens when r is 0.6 or 0.7.And if r=1(it means image just influenced by Gaussian noise) ,the image will be relatively difficult to recognize too.

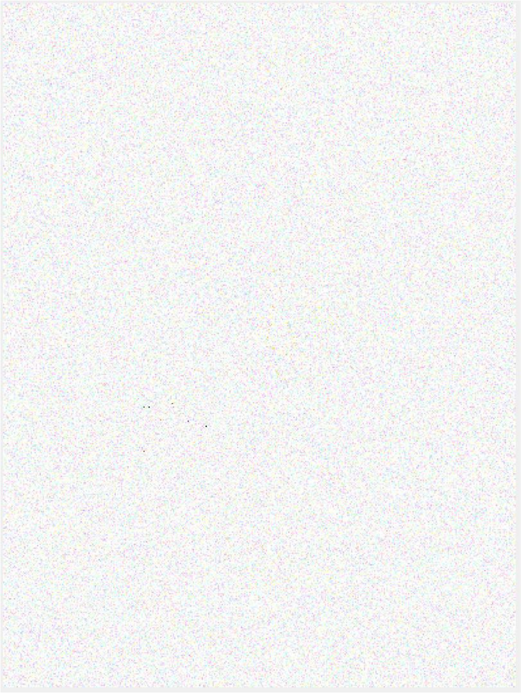


Image bruitée avec r=0.2

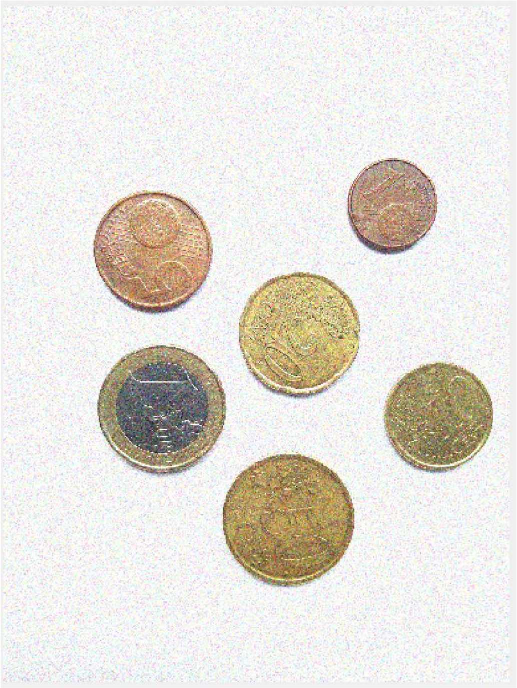
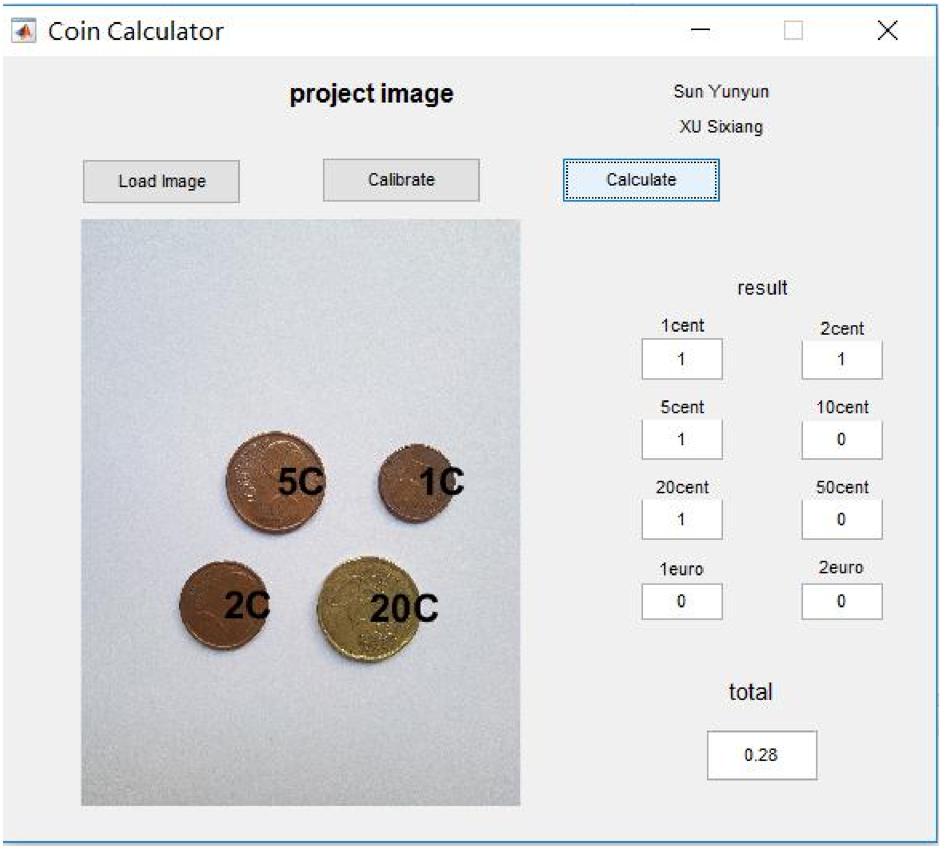


Figure7b : Image bruitée avec r=0.8

- **<u>Improvement :</u>**
  - We should have taken more images because several images in the dataset aren't in good quality
  - It would be better if we also use the method 'contour color' to recognize coins

- **interface (GUI) :**



Coin Calculator

project image

Sun Yunyun

XU Sixiang

result

1cent	2cent
1	1
5cent	10cent
1	0
20cent	50cent
1	0
1euro	2euro
0	0

total

0.28

GUI

4. Code explanation

- **Calibration.m** : calibration window to input SF
- **segmentCoins.m** : function to segment the coins from the image.
- **DemoGui.m** : the main interface of users
- **distinguishCoins.m** : function to detect the price of each coins
- **tse_imdetectmaxgrad.m,tse_imgrad.m,tse_imhysthreshold.m**: auxiliary functions used in segmentCoins.m to segment coins by the method gradient.

For more details about codes,please see comments in the Matlab codes.

5. Reference :

- Aide Matlab
- https://fr.mathworks.com/matlabcentral/fileexchange/54647-shadow-detection?s_tid=srchtitle Shadow detection proposed by matlab.com
- Documents of presentations of the 4 sessions.