IVR Coursework 1
Report

Weiting Goh (S1450710) and Tomas Markevicius(S1452595)

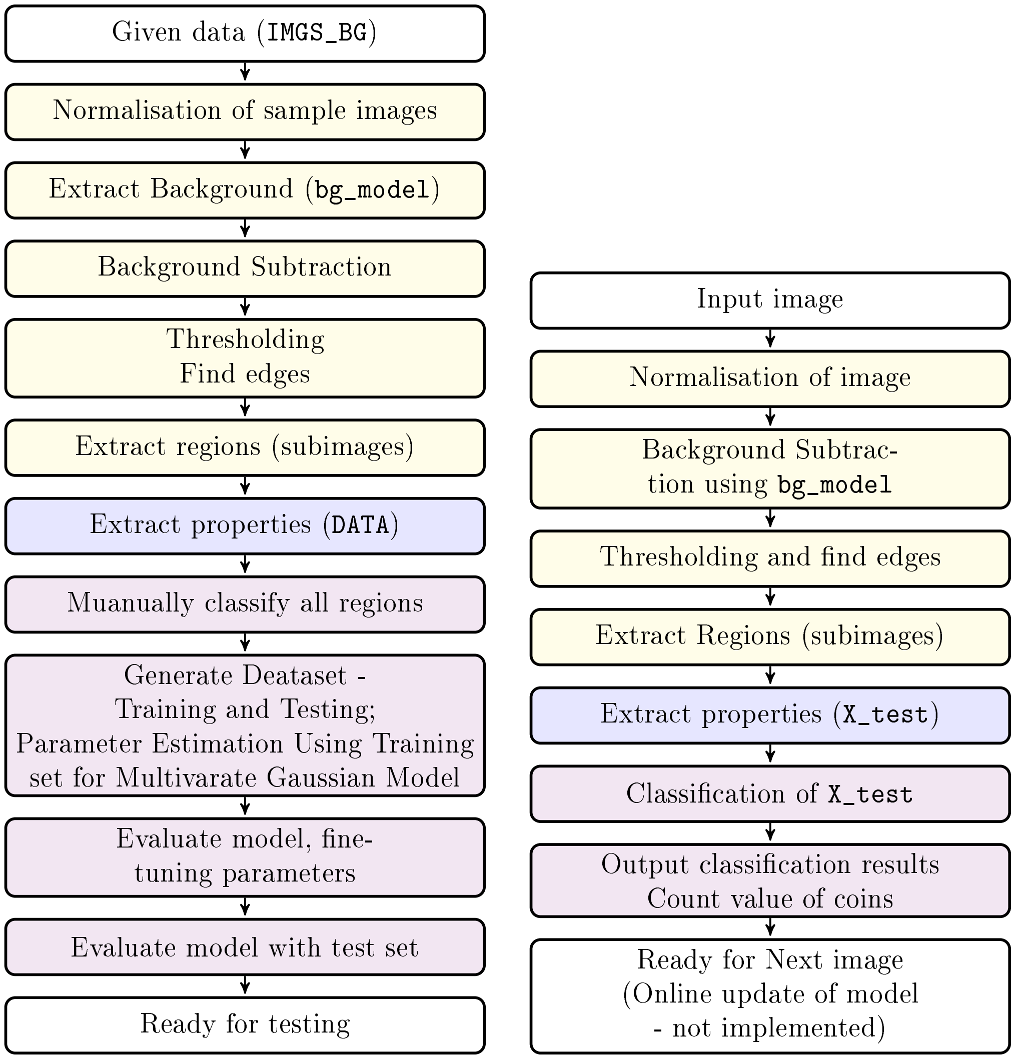
October 26, 2016

1 Introduction

To recognise and count the coins in an image, Coinsy, have three subtasks: 1) Image processing (includes image segmentation), 2) Features extraction, and 3) Classifiation (and then counting the coins).

For Coinsy to be a proficient detector, classifier and counter, we have to first train it. The training processes differs from evaluation as described in figure Figure 1 - the operation pipeline for training and evaluation. We will describe the methods that we use for each of these subtasks in the next section - methodology, and her evaluation results after. Lastly, we conclude with a discussion on Coinsy performance. The code for the project is listed in the appendix.

In the following subsections, we give an brief description of the operation pipeline for each subtasks from training to testing. The approach here is an abstract idea of what we did for our base model. Additional techniques were explored and will be discussed in later section. We start by understanding the data.



(a) Operation pipeline to train Coinsy

(b) Pipeline for (trained) Coinsy to count coins

Figure 1: Differences in Pipeline; Yellow boxes indicates the image processing procedures; Blue indicates the feature extraction procedures; Violet indicates the classification procedures.

Data

The data we were given are 14 images (consisting of 5 `harder` and 9 `simpler images`). Each consists of objects in the foreground for us to classify and calculate the values. The class labels and its respective object and value are as follow:

Class	Object	Value
1	1 Pound coin	1 pound
2	2 Pound coin	2 pound
3	50 Pence coin	50 pence
4	20 Pence coin	20 pence
5	5 Pence coin	5 pence
6	Washer with small hole	75 pence
7	Washer with large hole	25 pence
8	Angle bracket	2 pence
9	AAA Battery	-
10	Nut	-
11	unknown	-

The objects differ in colors, size and shapes; some are very similar to the background - such as 1 pound coins (see Figure 2). Hence, the features extracted must be invariant to rotation, and importantly to detect the objects will the images to be processed, such that the objects are salient to the computer vision. On top of the images itself output from `imshow` or `imagesc`, we can understand an image from the distribution of the pixel intensities (such as a histogram) and its gradient magnitude in each channel.

Figure 2: Sample images from given data.

Figure 3: Sample images after their background is subtracted.

Image Processing

The image processing step aims to 1) make all images (for training the model or for evaluation) comparable, 2) extract objects in the foreground (also known as image segmentation). The outcome of this stage is an array of subimages ready for feature extraction.

The central idea is to, first, model the background before subtracting it from all images; second - make the edges of the objects 'obvious' by finding a suitable threshold to binarise the image; third - crop out the objects to obtain the subimages, which will be our data points.

Feaure Extraction

Given the subimages, this stage aims to represent each subimages with a feature vector that contains properties to adequately describe the class it belongs to (or shape). Notaby, we have many circular objects varying in size. This renders global descriptors such as convexity and elongation less useful for these classes.

Classification

The task here is to label an unknown (sub)image given its set of feature vector. The model we consider is a multivarate gaussian classifier that classify an image based on the parameters of each class (class mean and standard deviation (or variance)). This multivarate gaussian classifier outputs the posterior probability of a given feature vector x ($P(C_k|X)$) for each class k, with the class giving the highest probability being the label. That is $x.class = argmax_k P(C_k|x)$.

A slight tweak for Coinsy is her ability to reject the class label output by the classifier if the highest probability falls out of her confidence interval. In such cases, Coinsy will intervene and change the class to unknown (class 11).

The last step of Coinsy is to sum the value of all the objects she managed to identify from a given image.

2 Methodology

In this section we describe the techniques we considered and brainstormed during the project and those that are implemented in the base model.

Table 1: Summary of techniques used for each tasks

Task	Subtask	Techniques	Base Model
Image Processing			
Feature Extraction	Global Descriptors	asd	✓
		Hu's Invariant moments	✓
		Complex moments	✓
Classification		Multivarate Gaussian Model Linear Discriminant	✓

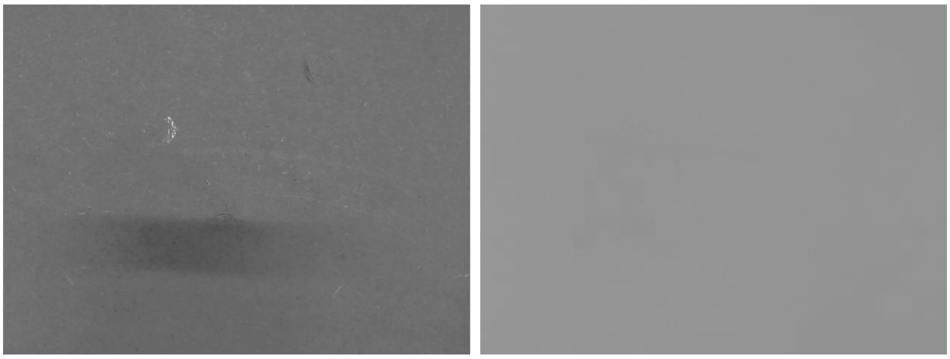
Background Subtraction

In this coursework, since a background image is not readily available, we have to model it. Noting that images varies in illumination, we have to make the images comparable by normalising it first, using the following formula.

$$P_{r,c}(R', G', B') = \left(\frac{R}{\sqrt{(R^2 + G^2 + B^2)}}, \frac{G}{\sqrt{(R^2 + G^2 + B^2)}}, \frac{B}{\sqrt{(R^2 + G^2 + B^2)}}\right)$$

With objects scattered around randomly in the images, we find the median of all image pixels for each channel separately in order to reconstruct the background.

The outcome of the background with and without normalisation is shown in Figure 4.



(a) Background model without normalisation (b) Background model after normalisation

Figure 4: Background model generated from all 14 images

The sample images with their background removed is shown in Figure 3. It is evident that the background removal process removed the background - making the images appearing black. However, it also inevitably reduce the intensity for the bottom half of each images, such that the objects are no longer salient to our eyes. This is because the background we modelled have a lower intensity at the bottom, possibly due to presence of shadow in all 14 images.

Nevertheless, the historgram is still bimodal, which is essential for thresholding to be effective.

Segmentation

2.1 Classification

3 Result

A For training -

References

[1] E.R. Davies. *Computer and Machine Vision (Fourth Edition)*. Academic Press, Boston, fourth edition edition, 2012. ISBN 978-0-12-386908-1. doi: http://dx.doi.org/10.1016/B978-0-12-386908-1.00001-X. URL `http://www.sciencedirect.com/science/article/pii/B978012386908100001X`.

[2] Noah Snavely. Lecture 2: Image filtering. URL `http://www.cs.cornell.edu/courses/cs6670/2011sp/lectures/lec02_filter.pdf`.

[3] A. Walker R. Fisher, S. Perkins and E. Wolfart. Hipr2 - image processing learning resources. URL `http://homepages.inf.ed.ac.uk/rbf/HIPR2/index.htm`.

[4] Andrew Ng. Stanford machine learning class notes. URL `http://www.holehouse.org/mlclass/index.html`.