

TPR2251: PATTERN RECOGNITION

*MELAKA CAFÉ RECOGNITION SYSTEM*

\*\*\*\*\*\*\*\*

Trimester 2, 2017/2018

LECTURER: DR. TEE CONNIE

GROUP: SE1A

|  |  |  |
| --- | --- | --- |
| NAME | ID | SIGNATURE |
| Phuah Chee Woei | 1141128570 |  |
| Sia Chun Wai | 1141128429 |  |
| Do Chen Hao | 1141128675 |  |

# Table of Contents

[Introduction 1](#_Toc506170566)

[Overview 1](#_Toc506170567)

[Mori Café 1](#_Toc506170568)

[Ice Town 1](#_Toc506170569)

[The Alley 1](#_Toc506170570)

[Design and Implementation 2](#_Toc506170571)

[Flow chart 2](#_Toc506170572)

[Methods used for segmentation 3](#_Toc506170573)

[Feature extraction and Classification 3](#_Toc506170574)

[Highlights of special features 3](#_Toc506170575)

[Input Data 4](#_Toc506170576)

[Mori Café 4](#_Toc506170577)

[Ice Town 4](#_Toc506170578)

[The Alley 4](#_Toc506170579)

[Tyj](#_Toc506170580) **[Error! Bookmark not defined.](#_Toc506170580)**

[Experiment Setup 5](#_Toc506170581)

[Performance Evaluation 6](#_Toc506170582)

[Conclusion 6](#_Toc506170583)

# Introduction

## Overview

The Melaka Café Recognition System is built to recognize a café when given an image as input.

The purpose of this project is to expose us to pattern recognition systems gain knowledge on solving practical problems by applying the knowledge and tools learned in this course. We are required to choose at least 3 cafés to be our subjects.

## Mori Café

* Location: Bukit Beruang



## Ice Town

* Location: Bukit Beruang

****

## The Alley

* Location: Kota Laksamana

# C:\Users\Chun Wai\Downloads\IMG_20180107_102422.jpg

# 

# Design and Implementation

## Flow chart

## Methods used for segmentation

For this part we wrote several scripts to ease the process of making changes to our data sets. We wrote a script to change the names of our text files, check the content of our text files, convert our images to black and white, appending the data from every image into a single matrix text file, convert all of our images into text file, convert the resolution of our photos if the resolution is different, and resize every photo into a predetermined size.

**Change name of text file**

1. **import** numpy as np
3. **for** i **in** range(1, 4):
4. data = np.loadtxt(('AlleyW1p{}NRes.txt').format(i))
5. np.savetxt('Test{}.txt'.format(i), data)
6. **print**('Test{}.txt'.format(i), data.shape)

**Check content of text file**

1. **import** numpy as np
3. data = np.loadtxt('AlleyM.txt')
4. **print**(data)
5. **print**(data.shape)

**Convert images to black and white**

1. **from** PIL **import** Image, ImageEnhance, ImageFilter
2. **import** numpy as np
3. **import** matplotlib.pyplot as plt
4. **from** scipy.misc **import** imsave
6. **for** i **in** range(1,34):
7. im = 'MoriW3p{}Resz.jpg'
8. img = Image.open(im.format(i)).convert('LA')
9. imgg = 'MoriW3p{}BW.png'
10. img.save(imgg.format(i))

**Appending data to another text file**

1. **import** numpy as np
3. matrix = []
5. **for** i **in** range(43, 57):
6. data = np.loadtxt(('Mori{}.txt').format(i))
7. matrix.append(data)
9. matrix = np.array(matrix)
10. np.savetxt('MoriTest.txt'.format(i), matrix)
11. **print**(matrix)
12. **print**(matrix.shape)

**Convert image into text**

1. **from** PIL **import** Image
2. **import** numpy as np
4. arry = np.zeros(shape=(2, 14400))
6. **for** i **in** range(3, 4):
7. img = Image.open(('AlleyW1p{}BWR.png').format(i))
8. img = np.array(img)
9. X = img[:, :, 0]
10. X = np.reshape(X, (-1, 1))
11. X = X.T
12. np.savetxt(('AlleyW1p{}BWR.txt').format(i), X)
13. **print**(X)
14. **print**(X.shape)

**Convert resolution if matches condition**

1. **import** numpy as np
3. matrix = []
5. **for** i **in** range(1, 34):
6. data = np.loadtxt(('MoriW3p{}BWR.txt').format(i))
7. shape = data.shape
8. shape = int(shape[0])
10. **if** shape == 14400:
11. data = data[:-160]
12. np.savetxt('MoriW3p{}NRes.txt'.format(i), data)
13. **print**('MoriW3p{}NRes.txt'.format(i), data.shape)

**Resize image**

1. **from** PIL **import** Image
3. basewidth = 160
5. **for** i **in** range(1,34):
6. im = 'MoriW3p{}BW.png'
7. img = Image.open(im.format(i))
8. wpercent = (basewidth/float(img.size[0]))
9. hsize = int((float(img.size[1])\*float(wpercent)))
10. img = img.resize((basewidth,hsize), Image.ANTIALIAS)
11. imgg = 'MoriW3p{}BWR.png'
12. img.save(imgg.format(i))

## Feature extraction and Classification

For feature extraction we used both the Principal Component Analysis (PCA) and Fisher’s Linear Distribution (FLD) method to further increase the accuracy of the system.

**PCA**

We used our own training and testing data sets

We also changed the height and width values to 160 and 89 respectively.

## Highlights of special features

# Input Data

## Mori Café

## Ice Town

## The Alley

# Experiment Setup

# Performance Evaluation

## Conclusion

Throughout the entire project many discussions and changes took place. There were problems in preparing the train and test data. Writing the code to convert an image into a text was not easy and there were problems like missing pixel. We have managed to solve it only after few days of struggle and experimenting. Once we have those data, we used PCA to find the principal components of each café (mean image).

# References