UTM
UNIVERSITI TEKNOLOGI MALAYSIA

Malaysia-Japan International Institute of Technology (MJIIT)

SMJE4263 COMPUTER INTEGRATED MANUFACTURING

Individual Assignment: Extracting Information from Receipt and Invoice

Name: Bhrnitharan A/L Nadaraja

Matric No: A19MJ0018

Lecturer: Prof. Madya Ir. Dr. Zool Hilmi bin Ismail

Introduction

Text recognition is another name for optical character recognition (OCR). Data is extracted and reused from scanned documents, camera photos, and image-only PDFs by an OCR programme. The original material can be accessed and edited by using OCR software, which isolates letters on the image, turns them into words, and then turns the words into sentences. Furthermore, it does away with the requirement for human data entry.

OCR systems transform physical, printed documents into machine-readable text by combining hardware and software. Text is copied or read using hardware, such as an optical scanner or specialized circuit board; the sophisticated processing is then usually handled by software.

OCR software can use artificial intelligence (AI) to create more sophisticated intelligent character recognition (ICR) techniques, such as recognising languages or handwriting styles. OCR is most frequently used to convert paper-based legal or historical documents into pdf files that can then be edited, formatted, and searched just like word processor-created documents.

Methodology

The software that was used for the OCR is Python. Python libraries were used such as tesseract, Pytesseract and PIL. Pytesseract is a useful tool for activities that need working with text embedded in photographs or scanned documents since it allows you to extract text from images. The Tesseract OCR engine, an open-source OCR engine created by Google, is wrapped in a programme called Pytesseract. The library is simple to use and can be smoothly included into Python programmes. The tesseract engine was downloaded from github. The loaded image was processed using Pytesseract's 'image_to_string() function. The extracted text is returned as a string after being entered as an image. Figure 1 shows the images of tesseract and Python

Figure 1: Tesseract and Python

Codes

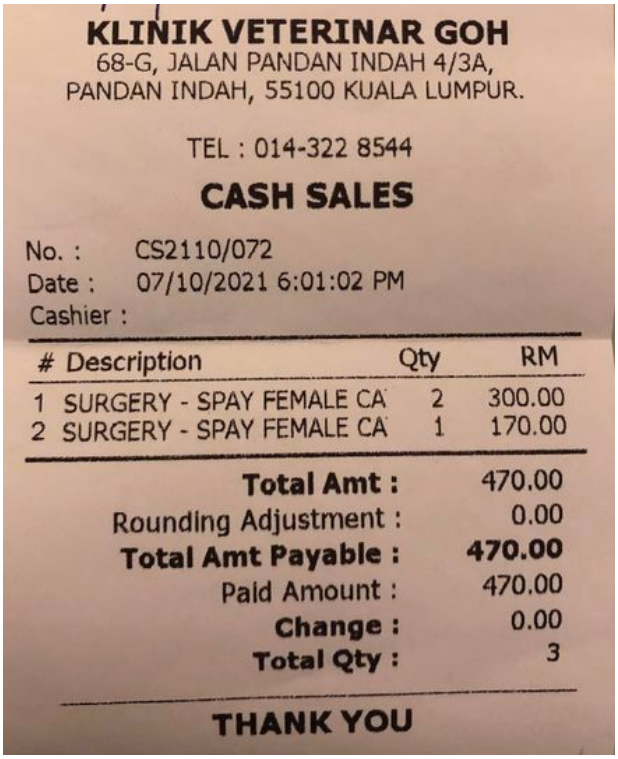
```
from PIL import Image

import pytesseract


# Load the receipt image

image = Image.open('receipt_image.png')


# Preprocess the image (example: resizing and grayscale conversion)

image = image.resize((800, 600))  # Resize the image for better OCR performance

gray_image = image.convert('L')   # Convert the image to grayscale


# Apply any other preprocessing techniques as needed (e.g., thresholding, denoising)

# ...


# Perform OCR on the preprocessed image and extract text

extracted_text = pytesseract.image_to_string(gray_image)


# Print the extracted text

print(extracted_text)
```

```
import re

# Regular expressions to match store name and total amount patterns
store_name_pattern = r'Store Name: (.+)'
total_amount_pattern = r'Total: (\d+\.\d+)'

# Find matches for the patterns in the extracted text
store_name_match = re.search(store_name_pattern, extracted_text)
total_amount_match = re.search(total_amount_pattern, extracted_text)

# Extract the matched values
if store_name_match:
    store_name = store_name_match.group(1)
else:
    store_name = "Not found"

if total_amount_match:
    total_amount = total_amount_match.group(1)
else:
    total_amount = "Not found"
```

Print the extracted information

print("Store Name:", store_name)

print("Total Amount:", total_amount)



KLINIK VETERINAR GOH
68-G, JALAN PANDAN INDAH 4/3A,
PANDAN INDAH, 55100 KUALA LUMPUR.

TEL : 014-322 8544

CASH SALES

No. : CS2110/072
Date : 07/10/2021 6:01:02 PM
Cashier :

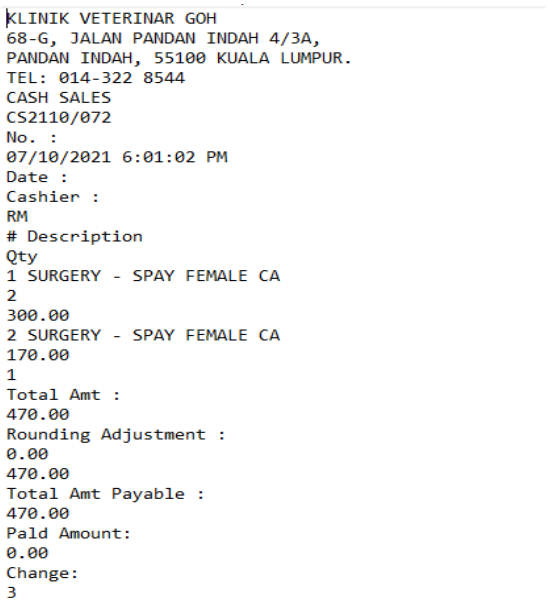
#	Description	Qty	RM
1	SURGERY - SPAY FEMALE CA	2	300.00
2	SURGERY - SPAY FEMALE CA	1	170.00

Total Amt :	470.00
Rounding Adjustment :	0.00
Total Amt Payable :	**470.00**
Paid Amount :	470.00
Change :	0.00
Total Qty :	3

THANK YOU

Figure 2: Receipt

Results



```
KLINIK VETERINAR GOH
68-G, JALAN PANDAN INDAH 4/3A,
PANDAN INDAH, 55100 KUALA LUMPUR.
TEL: 014-322 8544
CASH SALES
CS2110/072
No. :
07/10/2021 6:01:02 PM
Date :
Cashier :
RM
# Description
Qty
1 SURGERY - SPAY FEMALE CA
2
300.00
2 SURGERY - SPAY FEMALE CA
170.00
1
Total Amt :
470.00
Rounding Adjustment :
0.00
470.00
Total Amt Payable :
470.00
Pald Amount:
0.00
Change:
3
```

Figure 3: Text extracted from the receipt

Discussion

The accuracy of OCR can be impacted by the intricacy of the layout, font styles, and picture quality. Therefore, testing with preprocessing methods and OCR settings may be required to get correct results for the particular receipt types that we are working with. Therefore, the Python code has successfully extracted the desired information from the receipt such as the store name, date and the total amount.

Conclusion

Hence, using OCR for receipt processing has significant benefits for both people and companies. It is a useful tool for financial and accounting operations since it speeds up data capture, enhances accuracy, and simplifies cost management. OCR technology is expected to improve receipt processing efficiency even more as it develops, changing how businesses handle and maintain receipts in the process. Python demonstrates to be a flexible and useful option for integrating OCR in processes for processing receipts. The influence on receipt processing efficiency is anticipated to increase as OCR technology and Python continue to advance, providing even more cutting-edge solutions to many sectors.