AI-Assisted Budget Keeper

First A. Au Ho Kuen1,\*, Second B. Chen Kyle2,\*\*, and Third C. Yang Hip Hing3,\*\*\*

*1CMA Secondary School, Hong Kong*

*2Queen's College, Hong Kong*

*3Tseung Kwan O Government Secondary School, Hong Kong*

*\*Contact:* [*s171025@cmass.edu.hk*](mailto:s171025@cmass.edu.hk)

*\*\*Contact:* [*s15112@qc.edu.hk*](mailto:s15112@qc.edu.hk)

*\*\*\*Contact:* [*tko-186694@tkogss.edu.hk*](mailto:tko-186694@tkogss.edu.hk)

***Abstract*— This document gives a brief overview of our project, which is an AI-assisted budget keeper. This aims to improve existing budget applications by integrating AI into it. We hope this will make young people understand the importance of financial literacy and pick up personal financial management.**

1. Introduction

Personal financial management has always been extremely important. Knowing how to effectively manage your finances enables you to easily and steadily grow your wealth. In this day and age, with financial markets becoming progressively more complex and the options of financial products and services growing in number, knowing how to capably manipulate your money will undoubtedly give you an edge in life. In spite of its significance, financial literacy among the general populace is surprisingly low. With the situation being especially severe among young people. With reference to fig. 1, we can see that only 13% of the people aged between 18-24 in the USA can count as financially literate.

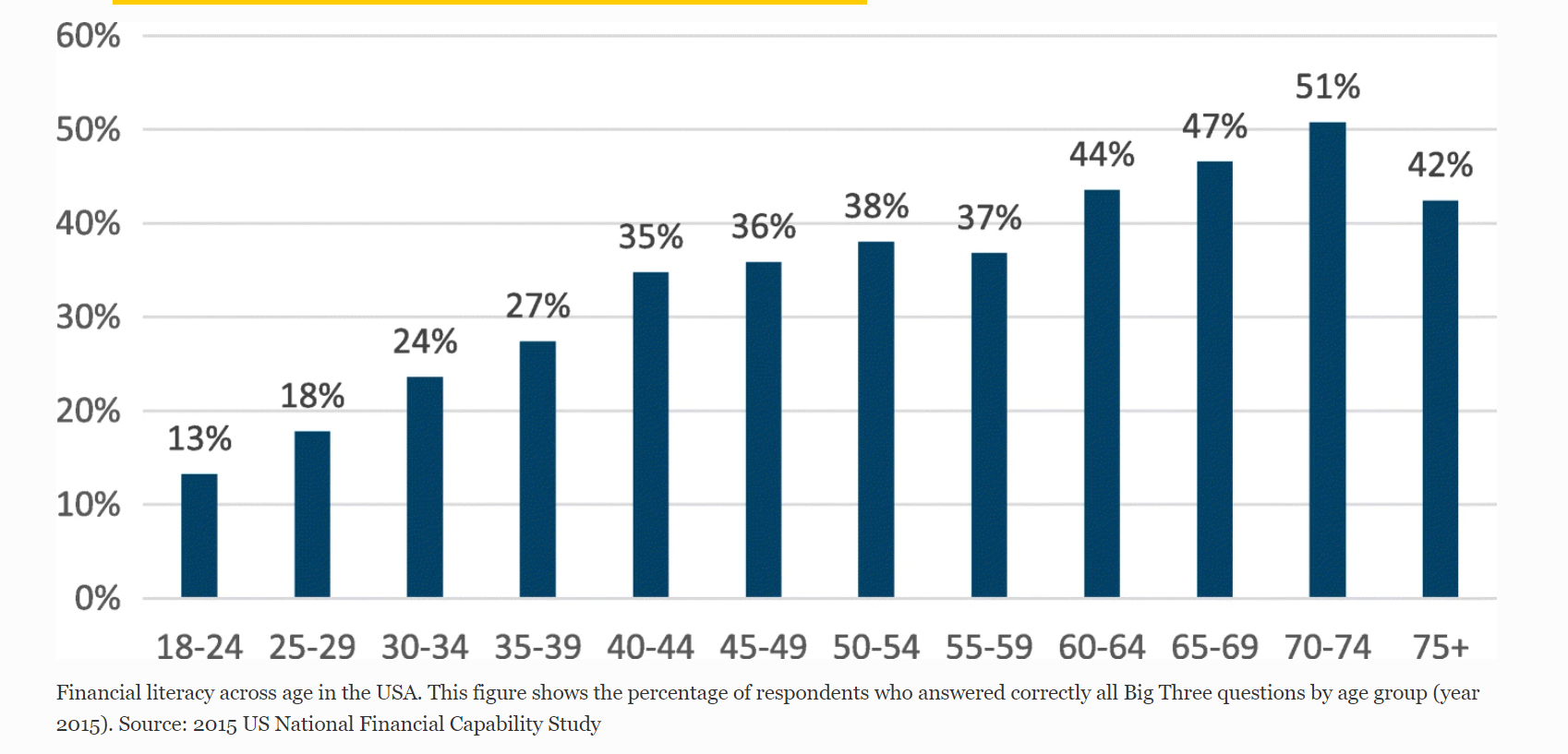


Fig. 1 Financial literacy across age in the USA. This figure shows the percentage of respondents who answered correctly all Big Three questions by age group (year 2015). Source: 2015 US National Financial Capability Study[1]

With this in mind, our group wanted to create something that could alleviate this issue. Budgets are well-known and powerful tools for personal financial management. However, it is not very popular, and a trifling amount of people use it. Our group believes the reason is because the manual input process of each item is overly tedious and boring. Therefore, we decided to use AI to streamline the input process. We decided to develop a receipt scanner as it is both more convenient and interesting. We hope this new input method will encourage more people to start and keep using budgets.

1. Methodology

Optical character recognition is used in our project for image recognition for scanning the receipt and turning it into text. Optical character recognition (OCR) is a technology that enables the conversion of scanned or digital images of text into machine-encoded text. Tesseract is the OCR engine developed by Google and used for developing our project.

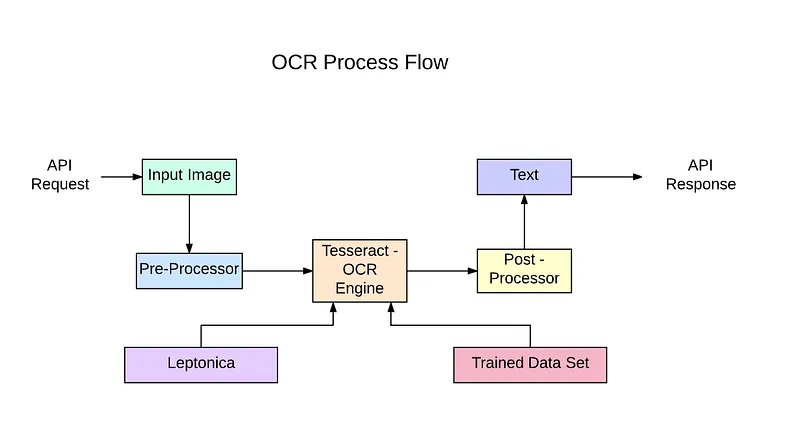


Fig. 2 Process Flow of using Tesseract OCR engine.[2]

The methodology of OCR can be broken down into several stages:

1. *Image Pre-processing*

During this stage, the input image is processed to remove any artifacts or distortions that could affect the accuracy of the OCR results. This can include techniques such as thresholding, dilation, erosion, and deskewing. The goal is to obtain a clean, clear image that is easy for the OCR engine to analyze.

1. *Segmentation*

In this stage, the OCR engine separates the image into individual characters and words. This is done by analyzing the spatial relationships between the characters and identifying areas of whitespace. This stage is critical for the accuracy of the OCR results, as it determines how the characters will be grouped and processed.

1. *Feature extraction*

In this stage, the OCR engine extracts a set of features from each character and word. These features can include information about the shape, size, and orientation of the character, as well as information about the texture and intensity of the pixels. The extracted features are used as inputs to the machine learning classifier, which will be used to recognize the characters.

1. *Character recognition*

In this stage, the OCR engine uses the extracted features to recognize each character and assign it a corresponding machine-encoded value, such as an ASCII or Unicode character. This is typically done using a machine learning classifier, such as a neural network or support vector machine. The classifier is trained on a large corpus of labeled text images, which allows it to learn the patterns and relationships between the features and the characters.

Fig. 1 shows an example of a processed image with recognized and outlined texts using Tesseract, and OpenCV, an open-source computer vision and machine learning software library. Whereas some text may be recognized wrongly or be skipped.

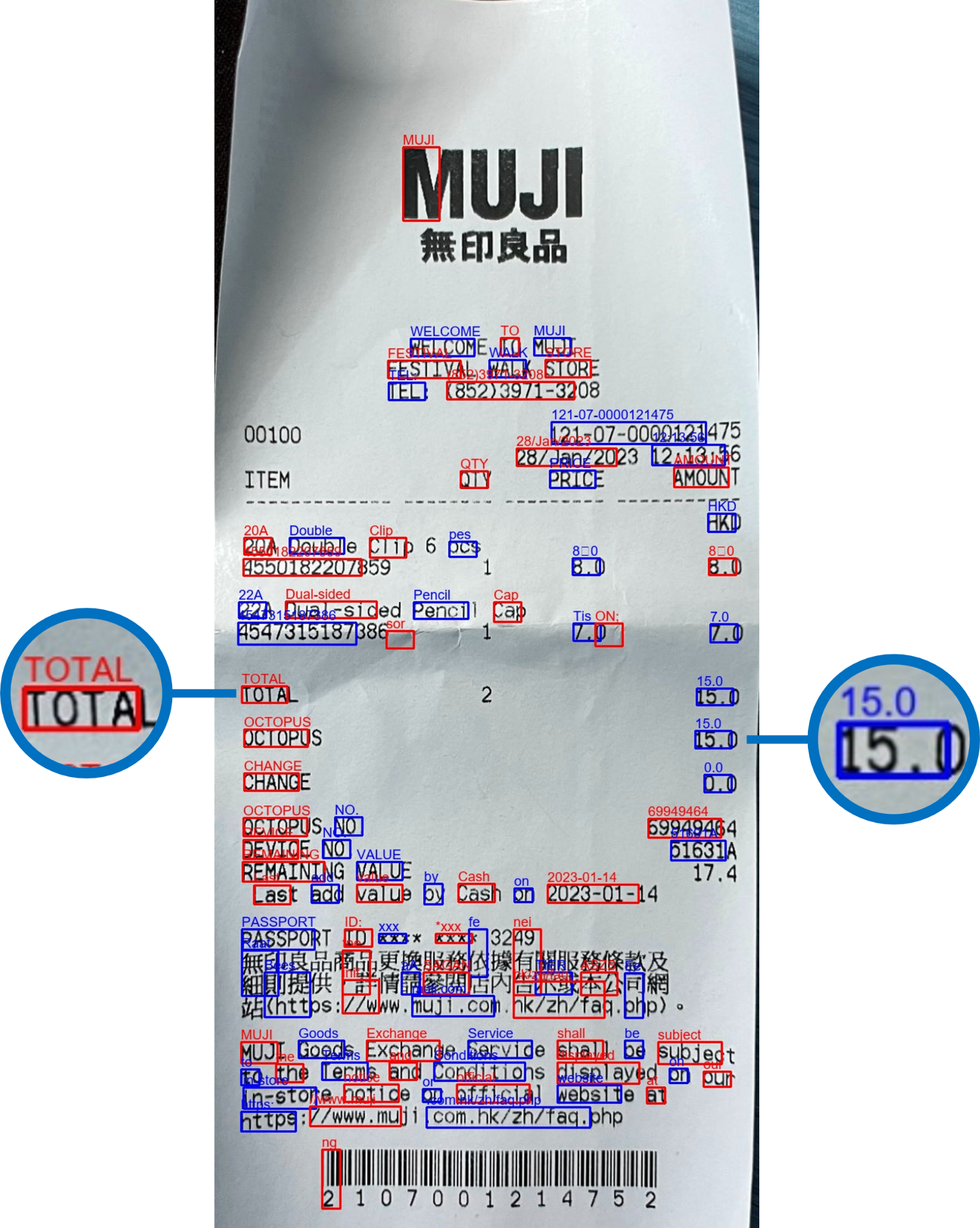


Fig. 3 A receipt after character recognition and outlined the texts with coloured boxes processed by Tesseract and OpenCV

1. *Context analysis*

In this stage, the OCR engine uses the recognized characters to analyze the context of the text, such as the words, sentences, and paragraphs. This information is used to improve the accuracy of the OCR results, by correcting any errors or inconsistencies in the recognized text. For example, if the OCR engine recognizes a sequence of characters as a misspelled word, it can use the surrounding context to correct the misspelling.

1. *Output processing*

In this final stage, the OCR engine converts the recognized characters into a machine-readable format, such as plain text, HTML, or XML. This step may also include post-processing, such as spell checking, language modeling, and error correction algorithms, to improve the overall accuracy of the OCR results.

These stages are performed in an iterative manner, with the OCR engine using the output of one stage as input for the next stage. Many factors, including the quality of the input image, the complexity of the text, the accuracy of the machine learning classifier, and the effectiveness of the pre-processing and post-processing steps influence the accuracy of the OCR results.

Overall, OCR is a complex process involving image processing, machine learning, and language modeling techniques.

1. Results

The result is shown in our YouTube video. (<https://youtu.be/yKiqNuqOK1I>). Applying this receipt scanner, we were able to scan receipts with relatively high accuracy.

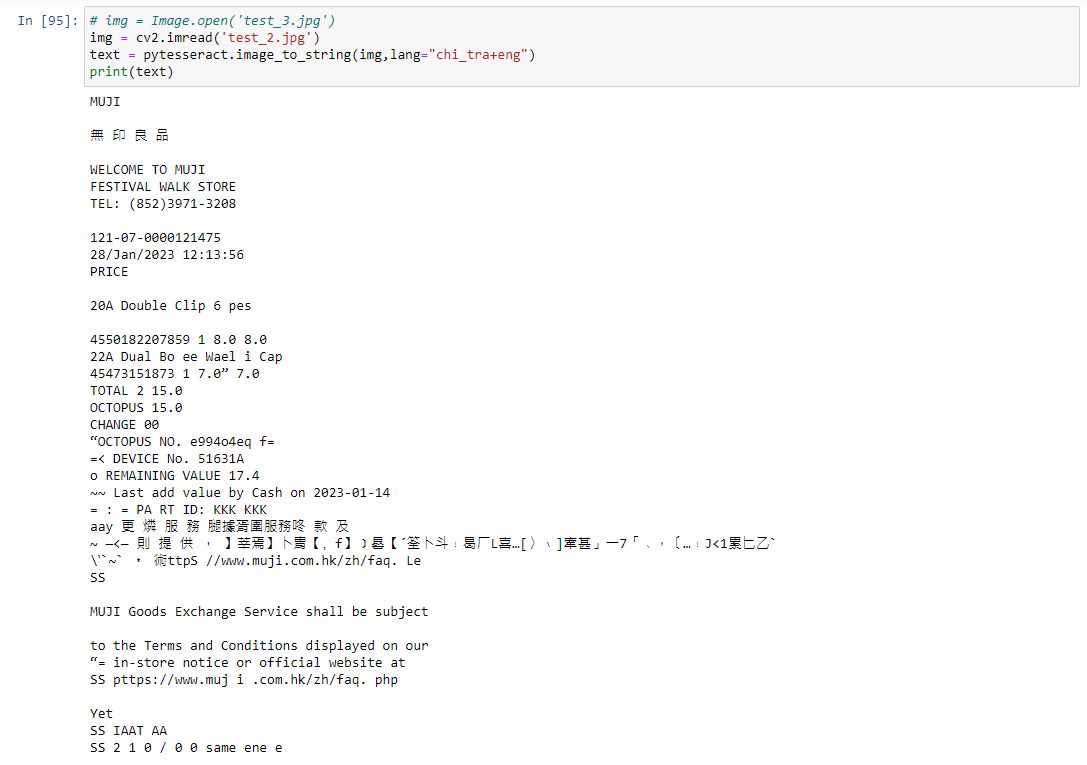
 

Fig.4 The inputted receipt Fig. 5 The output after running the OCR engine

However, as not all the scanned words were useful, we had to simplify the output to extract the useful information which we put in the budget.

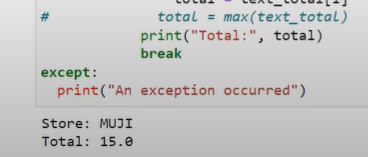


Fig. 6 The simplified output with key information

Here is another receipt we scanned with our receipt scanner.

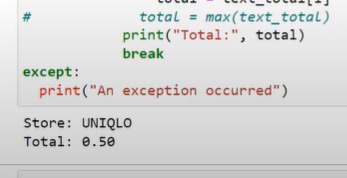
 

Fig. 7 The inputted receipt Fig. 8 The simplified output

Here is our GitHub source code: <https://github.com/s171025-TerryAU/Budget-Keeper/>

1. Conclusion/Reflection

In this project, we were able to develop a working receipt scanner by applying the knowledge we learnt from the course and from our own research. This receipt scanner is then used to make a prototype of our AI-Assisted Budget Keeper. This may seem like a small achievement, but it is a huge step forward for everyone in our group. Having not done anything like this before in the past, we all gained valuable experience in problem identification, solution brainstorming and product development. Key skills required to establish a start-up. We will surely benefit greatly from this.

Looking forward, it goes without saying that our prototype has plenty of room and potential for growth. Through collecting mentors’ feedback and brainstorming ourselves, we have already thought of many notable features we would like to add if given more time. Examples include adding a verbal input option using voice recognition, introducing a handy way to input transactions that do not involve receipts, and using AI to analyse and predict user spending patterns, which will help our users better plan and execute financial goals. Our group hopes that through our continued effort, we will be able to make a genuine impact on this issue.

Acknowledgment

To end, we would like to express our heartfelt gratitude to Dr Ray, John, Tiger, Clarice, all TAs, mentors, guest speakers, all staff members involved, and our schools for giving us this amazing opportunity to learn about AIoT, supporting us along the journey, and to give us such an eye-opening experience that will undoubtedly broaden our horizons. We would also like to thank the EDB for funding this program. Last but not least, I’d also like to thank both of my teammates as without either of them this project would never have gone so smoothly. We look forward to applying the skills we acquired in this course for the betterment of society.

References

[1] Annamaria Lusardi. (2019). Financial literacy and the need for financial education: evidence and implications. Retrieved from: <https://sjes.springeropen.com/articles/10.1186/s41937-019-0027-5>

[2] Suresh Thiyagaraj. (2020). A comprehensive guide to OCR with Tesseract, OpenCV and Python. Retrieved from: <https://medium.com/nanonets/a-comprehensive-guide-to-ocr-with-tesseract-opencv-and-python-fd42f69e8ca8>