eExpense: A Smart Approach to Track Everyday Expense

Shahed Anzarus Sabab[1], Sadman Saumik Islam[2], Md. Jewel Rana[3], Monir Hossain[4]
Department of Computer Science and Engineering (CSE)
[1, 3, 4]Northern University Bangladesh, [2]Daffodil International University
Dhaka, Bangladesh
Email: sabab.iutcse@gmail.com[1], ssi.sakib@gmail.com[2], zxdmjr@gmail.com[3], mknishad1@ymail.com[4]

Abstract— **Tracking regular expense is a key factor to maintain a budget. People often track expense using pen and paper method or take notes in a mobile phone or a computer. These processes of storing expense require further computations and processing for these data to be used as a trackable record. In this work, we are proposing an automated system named as eExpense to store and calculate these data. eExpnese is an application that runs on Android smartphones. By using this application, users can save their expense by simply scanning the bills or receipt copies. This application extracts the textual information from the receipts and saves the amount and description for further processing. It also monitors user's income by tracking the received SMS's from the user's saving accounts. By calculating income and expense it produces the user's balance in monthly and yearly basis. Overall, this is a smart automated solution for tracking expense.**

Keywords—smartphone; expenditure; track expense; OCR; SMS tracking.

I. Introduction

From the beginning of human civilization, people have exchanged their fortune with each other for buying or selling goods. It has become a crucial and unchangeable part of our daily life since then. Most of us have a fixed income and we get it in a timely basis (i.e. daily, monthly, yearly etc.). Moreover, everyone follows a strict budget of expense. Generally, the budget is assembled as per category. The categories are distinct, for example, food, entertainment, transportation, education, healthcare, clothing etc. However, the budget of expense is restricted to the income. For that reason, we need to track our expense so that it doesn't exceed our budget. In old days, people used to track their expense manually i.e., using pen and paper system which takes a lot of effort and time.

Nowadays, the availability of electronic devices like smartphones, computers have made our life a lot easier and faster. We can use computers to track our daily expense by using the online and offline software available. But the computer is not accessible all the time. The smart solution to the problem is to use smartphones. Nearly 44% of the world population use smartphones [1]. Smartphones have become an irreplaceable part of our daily lives as they are always accessible on the go. There are some existing applications that can track daily expense [2]–[5]. These applications use a manual input system from the keyboard which is tiresome and time-consuming. To meet the challenge of avoiding manual input, we are proposing a smart method of doing the same work but in a more automated and efficient way which takes less time.

In the proposed approach, users can store expense simply by scanning any bill or receipt paper with a smartphone camera. We have used Optical Character Recognition (OCR) [6] to extract the information from the bills or receipts. This approach is also capable of tracking savings from user's saving accounts by reading the SMS's automatically from the message application of the android device. All in all, we are proposing an automatic tracking system for credit and debit.

The sections are structured as follows: in Section 2, the background analysis is covered. It includes some of the popular applications which are available. Section 3 defines our approach. Section 4 includes the features and functions of our approach. In Section 5, the smart features and functionalities are described which make this app different. Sections 6 includes the system evaluation and Section 7 includes a conclusion.

II. Related Works

There have been a lot of research regarding character recognition from documents and analyzing the acquired data. The development of different OCR engines has been a step forward in serving this purpose.

Chaudhuri et. al. [7] proposed a system which uses the OCR to read scripts written in two different languages i.e. Bangla and Devnagari. As the two-language had a lot of the same features because of both having the same origin, so the system could read the two languages using the same process. A set of algorithms were used for document digitization, skew detection, text line segmentation and zone separation, word and character segmentation, character grouping into basic modifier and compound character category. The system performed best for single font scripts printed on the clear document. Drira et. al. [8] proposed a modification of Weickert coherence enhancing diffusion filter for which new constraints formulated by the addition of the Perona-Malik equation. By doing so, it reinforced character discontinuity and eliminated the inherent problem of corner rounding while smoothing. This lead to the improvement of image quality of old documents. Qadri et. al. [9] proposed an automatic vehicle number plate recognition system based OCR to identify vehicles. The system was developed for highly secured areas and military purposes. The system first detects the vehicle and captures its picture then the number plate region is separated using image segmentation. OCR was applied on the segmented image to recognize the individual character with the help of database stored for each and every alphanumeric character. The acquired data using

978-1-5386-8279-1/18/$31.00 ©2018 IEEE

OCR was then compared with the records on the database to find specific information about the vehicle. Wang et. al. [10] proposed a system for solving the end-to-end problem of word detection and recognition in natural images. OCR was used on the input image for character recognition and thus the word detection. In our system, we have used the tesseract OCR engine which is very efficient in recognizing characters. Smith et. al. [11] gave an overview of the tesseract engine. Tesseract had been released for open source back in 2005. In all of the OCR engines, tesseract was the first to handle the black-on-white text for character recognition.

Expense management apps are very common in the application market. Many of them offer exciting features. Different apps have taken different approaches to manage the daily expense.

Daily Expense 3 [2] is a system that can track income and expense and classify them into categories. The application shows reports grouped by periods. Users can also schedule their recurring records. The application also creates a backup of their records to restore information if necessary.

AndroMoney [5] supports multiple accounts to manage expense and income. It uses cloud storage so that the data is safe. Users can set a budget for the expense and the app will notify if they exceed the budget. It provides a number pad to calculate any record. It generates trend, pie and bar charts for cash flow.

Monefy [3] - money manager is an expense management application that has an intuitive interface. It can store a record faster than the others mentioned above. It provides widgets to enhance ease of access. It provides default categories and option to add customized categories. It also provides an on-screen calculator to calculate the expense.

Expense Manager [12] is a feature-rich application, which is interactive and well balanced. Besides tracking user's income and expense it also saves the picture of a receipt. It also tracks tax, mileage, and debts. It provides some convenient tools such as currency converter, regular calculator, tip calculator, loan calculator, credit card payoff calculator etc.

The next application which is being used to track expense is Money Lover [4]. This application manages expense category wise. It also manages income, debts, and loans. Users can set events and save plans on this app. It saves the receipt. It also provides a currency converter and a calculator.

In our work, we have developed a system which has the functionalities of a traditional expense management android applications along with the Tesseract OCR engine's character recognition capability to take automatic input. The system is more interactive and operational than all the other solutions mentioned above.

III. Proposed System

Our proposed solution is a smart assistant for the users to track daily expense. The users simply need to scan receipts of any kind with their smartphone camera and our application is smart enough to detect the necessary words and amount from the receipt. The data of the expenditure is then saved on the database category wise. Our application structure has been divided into two major parts. One is debit and another is credit. All the expenditures are included in the debit part and the incomes are included in the credit part. The incomes are calculated automatically from the messages received in the inbox.

There are some additional features available in our system. The users can set a total budget for the whole month as well as for individual categories. The system will notify the users if they attempt to exceed the set budget. There is a pie chart (see figure 5) representation to summarize the total expense. A calendar view (see figure 6) is also available to show the expense histories of the users.

Our approach solves many issues and limitations of the currently available expense tracking systems in the market. It saves a lot of time and efforts of the users as the major processing is automatically done by the application.

IV. System overview

eExpense is divided into four major sections. Those are Debit, Credit, Balance, and history. This system works as a one-tap solution for tracking everyday expense. It also preserves yearly and monthly records. For the availability of the records, users can check their histories to keep track of their expense so that they do not exceed their pre-allocated budget. The main functions are described below:

A. Debit:

The debit information can be stored using the mechanism from Figure 1:

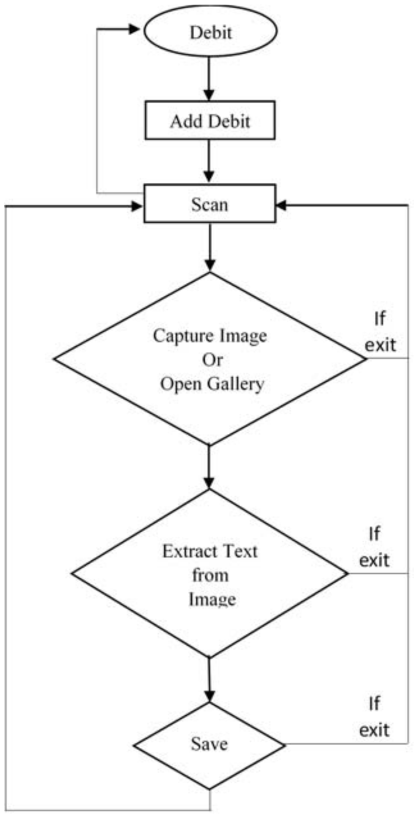


Fig. 1 Debit Information Store

For taking debit information, it is presumed that the users will take input data from money receipt with our automated solution or they will take input manually. The debit interface is designed keeping in mind that the users get the most benefit from our one-tap solution. To take debit information, the users will need to tap on the add button. It will open the phone camera. The image of any money receipt can be captured by the phone's camera or users can also use previous images from the gallery. After that, the intended texts are extracted from the image and provided in the debit interface for further modification. Users will have category information, description and total amount on the debit interface. The description and the total amount are automatically extracted from the receipt image (e.g., see figure 2). Users only need to select the category from the category selections. If the category already exists in the app then the users will select it otherwise users can also create new categories for future usage. Again, they also have the flexibility to change any of the automatically extracted information so that they can remove the wrong information before saving. Finally, users will need to give the approval to save the information. Figure 2 shows the debit input sample interface.

B. *Credit:*

The credit information can be stored using the mechanism from figure 3.

When users open the credit interface, the system reads all the messages from the user's messaging inbox. If the messages are from the bank about any transaction of credit, the app reads it and saves necessary information from the messages (SMS). There is also another option for saving credit. The users can manually give the input of credit in the app. After successfully saving the input, they can view the data on the list view. Figure 4 shows the credit input technique.

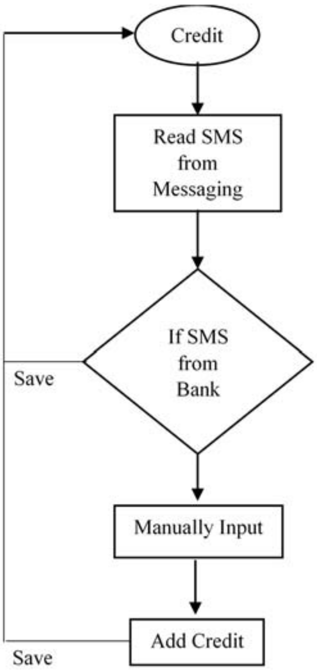
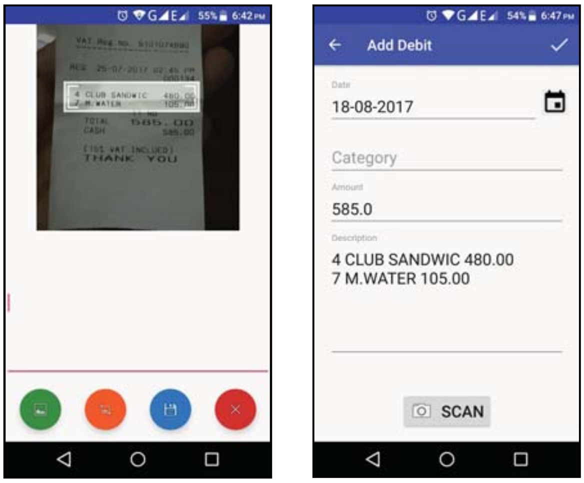


Fig. 3 Credit Information Store

C. *Balance:*

In balance interface, users can see two different types of pie charts, which are yearly and monthly balance. The pie charts represent the total estimation from all the categories (e.g., food, entertainment, transportation, education, healthcare, clothing, bank, groceries). Yearly balance includes the credit and debit information of a specific year. Monthly balance includes all the credit and debit information of 6 months. The balance interface is shown in Figure 5.



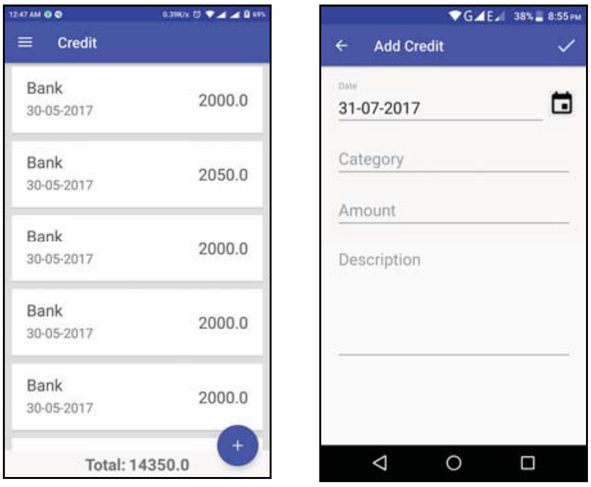
Date
18-08-2017

Category

Amount
585.0

Description
4 CLUB SANDWIC 480.00
7 M.WATER 105.00

Fig. 2 Receipt photo (left), Extracted debit information (right)



Bank 30-05-2017	2000.0
Bank 30-05-2017	2050.0
Bank 30-05-2017	2000.0
Bank 30-05-2017	2000.0
Bank 30-05-2017	2000.0

Total: 14350.0

Date
31-07-2017

Category

Amount

Description

Fig. 4 Automatic credit input (left), Manual credit input (right)

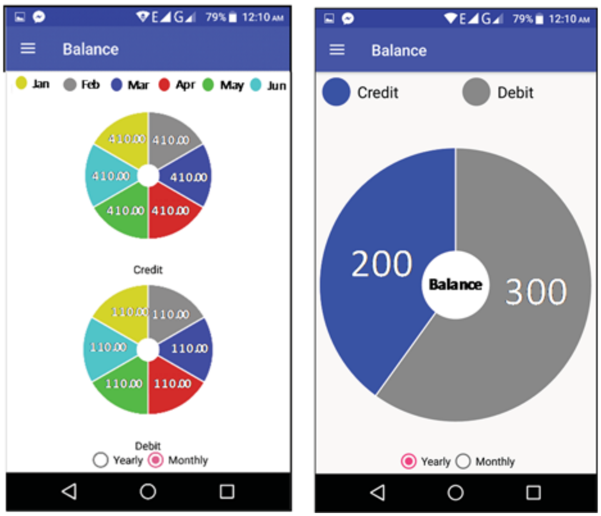
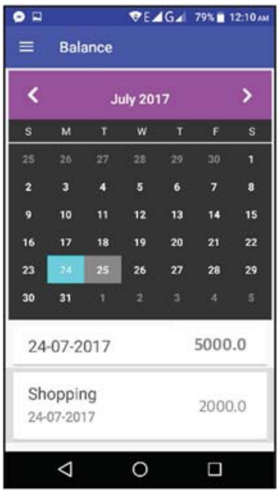


Fig. 5 Monthly balance (left), Yearly balance (Right)

D. History:

In the history section, users can see a calendar interface. In the calendar interface, users can tap and select a date and the total amount of expense on that date is shown. A user can also view category wise expense of a particular date. If a user wants, he can also see the whole history of inputs of a particular day. Figure 6 shows the history interface where the user can search for any specific day's expense. In the figure, the date 24 July 2017 is selected (i.e., tap to select any date) and the total expense amount is 5000.0 on that particular date. The information is shown category wise right below the total amount.



24-07-2017 5000.0

Shopping
24-07-2017
2000.0

Fig. 6 History in Calendar View

V. System Functionalities

The system has been designed following standard principles. Moreover, there are several specific contributions which make this app a novel approach while eliminating the issues with the traditional time-consuming approaches. The innovation of this system can be described as the following basis:

A. Interface Design:

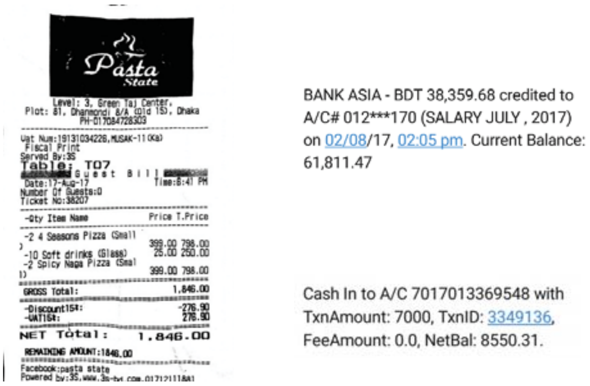
Our application is designed by following the Google Design Pattern rules [13, 14]. So that the users can use it with ease. The interface is simple and user-friendly. The contents of our application are divided into four major sections. Those are Debit, Credit, Balance, and History. The debits and credits are shown in different lists. The last entry is always on top of the list and the previous one is just below the last entry and so on. It allows users to access their latest entries first. Also, there is a search option for searching debit or credit entries. There is a floating action button on debit and credit interface (left section of figure 4) which is used to add new records. Therefore, new debit or credit entries can simply be added by tapping the floating add button. In the balance section, there are different pie charts showing the proportion of debits and credits (figure 5). The history section is represented by a calendar view (figure 6), where debit and credit entries of a particular date can be seen by tapping on a date.

B. Smart Features:

We have introduced some smart features in our application to save time and reduce the effort of the users. We used Optical Character Recognition (OCR) technique to take the input of a debit from the picture of receipts. OCR is the mechanical or electronic conversion of images of typed, handwritten or printed text into machine-encoded text [6]. It allows users to perform a debit entry by simply scanning a printed receipt. Our application will take the items and the amounts from the receipt and save them to the database. The text filtering is done based on the following keywords: Total amount, Net amount, Net total, Total. If any of the keywords is found, then the amount field of the interface is filled by the amount next to the found keyword. Otherwise, the system sums all the item's cost and fills the amount field of the interface by the sum total (i.e., amount field in figure 2). We have used Tesseract library [15] (Open Source OCR library) to convert the image into text.

The debit interface includes category information (figure 2). To ensure the minimal input, if any category is previously stored, the user will have suggestions while taking input. If the category is not previously stored, the new category information will be added to assist the user in the future [16].

Another automation is done in the credit section. The credit entries are imported automatically from the message inbox. Our system scans through the messages in the inbox and searches for messages from banks. The filtering is done with the following keywords: credit, credited, received, cash in. If any of the keywords exists in the message, the message is from the users' bank account and it represents a credit event [17]. Then the message is scanned to find the amount by looking at every character to determine if it is a numeric digit or not. After finding the digits, all the consecutive digits are converted to a decimal number. The decimal number is the amount credited to the user's bank account. The amount is then saved to the database along with the category 'Bank' and the date when the message was received. There's also an option available to manually input a credit (i.e., floating add button to add from figure 4). The sample debit and credit information are given in figure 7:



Pasta State
Level: 3, Green Taj Center,
Plot: 81, Dhanmondi 8/A (Old 15), Dhaka
PH-01708472830З

Vat Num:19131034226,MUSAK-11(Ka)
Fiscal Print
Served By:3S
Table: T07
[redacted] Guest Bill [redacted]
Date:17-Aug-17 Time:6:41 PM
Number Of Guests:0
Ticket No:38207

-Qty Item Name	Price	T.Price
-2 4 Seasons Pizza (Small)	399.00	798.00
-10 Soft drinks (Glass)	25.00	250.00
-2 Spicy Naga Pizza (Small)	399.00	798.00
GROSS Total:		1,846.00
-Discount15%:		-276.90
-VAT15%:		276.90
NET Total:		1.846.00

REMAINING AMOUNT:1846.00
Facebook:pasta state
Powered by:3S,www.3s-bd.com,01712111881

BANK ASIA - BDT 38,359.68 credited to A/C# 012***170 (SALARY JULY , 2017) on 02/08/17, 02:05 pm. Current Balance: 61,811.47

Cash In to A/C 7017013369548 with TxnAmount: 7000, TxnID: 3349136, FeeAmount: 0.0, NetBal: 8550.31.

Sample Debit Receipt Sample Credit Messages

Fig. 7 Sample Inputs

There is another smart feature in our application which is called target allocation. The users can set their target of expense for an upcoming month. The target can be set category wise. After setting a target, the system always shows the status of the target amount vs expense of a particular category. The system notifies the users if they attempt to cross the allocated target for a particular category.

The search option of our application is very smart. Users can search debit or credit items by date, category, and amount. It is also possible to search for the combination of date, category and amount.

We have introduced a calendar view to represent the expense history. We added on tap event listener to all the calendar date items. Debit entries of a particular date are shown after tapping a particular date of the calendar (figure 6).

VI. System Evaluation

Usability testing [18] has been used to evaluate our system. The essence of usability testing is to monitor users while they are using a system. This testing ensures the workable features and non-workable features from the users' feedback. The feedback proves to be a crucial part of the evaluation, which helps to design an efficient system.

We took the empirical approach [19] while we tried to examine and compare the usability of multiple conceptual designs. In the app market, we found several traditional apps which also calculate the day to day expenditure. By comparing with the investigated applications, we propose a better solution. To evaluate the usability of our system, we designed some of the experiments based on the following metrics:

- Time to complete a certain task.
- Success rate of task completion.
- Number of errors made.
- User rating.

We set the experiments in a controlled environmental setting with proper lighting conditions. We had a group of people of different ages. In total 18 participants contributed to this test. 8 of them were in the age range of 18-29, 6 of them were in the age range of 30-39 and 4 of them were in the age range of 40 and over. They were given a few tasks to complete. We tried to determine the performance of the users while they were performing the tasks. Their satisfaction level was also evaluated using questionnaires and interviews.

The participants were given two tasks: one for debit and one for credit. For debit, five receipts of different categories (receipt from the grocery shop, cinema ticket, gas bill, restaurant bill and bus ticket) were provided. Participants had to take the input of date, category, description, and total amount. For credit, two different transaction information (mobile banking and cash credit) were provided where the percipients had to take the information of total amount, category type and description. They had to complete these tasks using traditional application (manual input) and eExpense**.** They were given 3 minutes (i.e., 180 seconds) to complete these two tasks. The results were analyzed based on the total time taken to complete the tasks and average success rate.

The basis is the followings:

Total time = Average time to complete given tasks.

$$\text{Success rate} = \frac{\mathit{Completed\ tasks\ in\ given\ time}}{\mathit{Total\ tasks}} \times 100$$

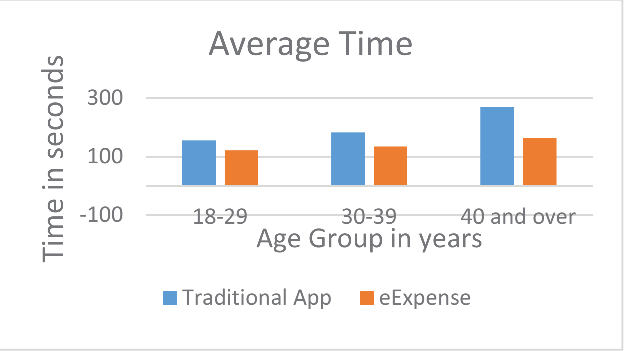


Fig. 8 Average Time to Complete Task

In figure 8 the average time taken to complete the tasks by the participants is shown. The participants of age group 18-29 took just over 2 and a half minutes (i.e., 155 seconds) on average to complete the given tasks by using traditional application where is it only took them 2 minutes and 2 seconds (i.e., 122 seconds) in average to complete the given tasks by using eExpense. The participants of age group 30-39 and age group of 40 and over took just over 3 minutes (i.e., over 183 seconds) and 4 and a half minutes (i.e., 271 seconds) on average accordingly to complete the given tasks by using traditional applications but it only took them 2 minutes and 15 seconds (i.e., 135 seconds) and 2 minutes and 44 seconds (i.e., 164 seconds) on average to complete the given tasks by using eExpense**.** We can observe from the figure 8 that most of the participants of age group 18-29 and 30-39 could complete the tasks in the given time by using both the traditional apps and eExpense. It took the participants of age group 18-29 almost 35 fewer seconds and almost 50 fewer seconds for the participants of age group 30-39 to complete the tasks by using eExpense. Surprisingly most the participants of age group 40 and over could not complete tasks in given time (i.e., 3 minutes) using the traditional application but almost all of them were successful to complete the tasks in given time by using eExpense.

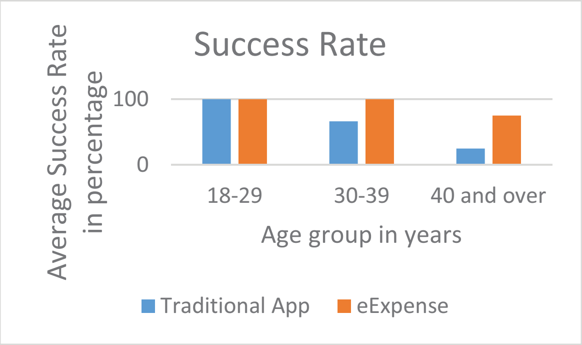


Fig. 9 Average Time to Complete Task

From the figure 9 we can observe that the success rate of the participants of age group 18-29 is 100% for both the traditional application and eExpense. The success rate of the participants of age group 30-39 for traditional application is 66.6% but the success rate using eExpense is 100%. Again, surprisingly the success rate of the participants of age group 40 and over for traditional application is 25% but the success rate using eExpense is 75%. Our proposed system proved to be effective for people of age group 40 and over.

After completion of the tasks, we interviewed the participants to get their feedback. They were also asked to rate eExpense based on their user experience. We found that most of them were satisfied and gave us positive feedback though they had some complaints on the accuracy rate. Later on, it was discovered that due to the lower image quality of receipts, the accuracy rate was low. Though those errors are solvable during the input taking. One limitation we discovered that our system could not detect image in lower lighting condition. Our system completely depends on optical character recognition (OCR) during taking Debit input which performs poor in lower quality images.

VII. Conclusion

In today's world, time is the most valuable asset because people lack ample of it. People are obsessed with completing tasks in lesser time and our system is an approach serving this purpose. eExpense can manage daily expense much faster than any other traditional app in the market which takes manual input. Our system proves to be most effective for the people aged 40 and over and an efficient solution comparing to any of the traditional applications. Nowadays, the world is leaning towards the one tap solution and our system is one of a kind. After all, automation is the way of future and eExpense can be a step towards it. The application still has a lot of aspects that need to be improved. The app performs poorly if the input image from the receipt includes a lot of noises. The app is unable to detect the region of interest automatically. On that note, the user has to set the region of the receipt to have a better result. Again, the performance of the character recognition from the receipt declines in low lights. Therefore, this system triggers few research scopes which can be a starting point for the improvement of the proposed approach.

VIII. Reference

[1] "44% of World Population will Own Smartphones in 2017." [Online]. Available: https://www.strategyanalytics.com/strategy-analytics/blogs/smart-phones/2016/12/21/44-of-world-population-will-own-smartphones-in-2017#.Wq1BcehuY2y. [Accessed: 17-Mar-2018].

[2] "Daily Expense 3 - Apps on Google Play." [Online]. Available: https://play.google.com/store/apps/details?id=mic.app.gastosdiarios. [Accessed: 17-Mar-2018].

[3] "Monefy - Money Manager - Apps on Google Play." [Online]. Available: https://play.google.com/store/apps/details?id=com.monefy.app.lite. [Accessed: 17-Mar-2018].

[4] "Money Lover: Budget Planner, Expense Tracker - Apps on Google Play." [Online]. Available: https://play.google.com/store/apps/details?id=com.bookmark.money . [Accessed: 17-Mar-2018].

[5] "AndroMoney (Expense Track) - Apps on Google Play." [Online]. Available: https://play.google.com/store/apps/details?id=com.kpmoney.android . [Accessed: 17-Mar-2018].

[6] "Optical Character Recognition (OCR) – How it works." [Online]. Available: https://www.nicomsoft.com/optical-character-recognition-ocr-how-it-works/.

[7] B. B. Chaudhuri and U. Pal, "An OCR system to read two Indian language scripts: Bangla and Devnagari (Hindi)," in *Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on*, 1997, vol. 2, pp. 1011–1015.

[8] F. Drira, F. LeBourgeois, and H. Emptoz, "Document images restoration by a new tensor-based diffusion process: Application to the recognition of old printed documents," in *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on*, 2009, pp. 321–325.

[9] M. T. Qadri and M. Asif, "Automatic number plate recognition system for vehicle identification using optical character recognition," in *Education Technology and Computer, 2009. ICETC'09. International Conference on*, 2009, pp. 335–338.

[10] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, 2011, pp. 1457–1464.

[11] R. Smith, *An Overview of the Tesseract OCR Engine*. 2007.

[12] "Expense Manager - Apps on Google Play." [Online]. Available: https://play.google.com/store/apps/details?id=com.expensemanager. [Accessed: 17-Mar-2018].

[13] "Introduction - Material Design." [Online]. Available: https://material.io/guidelines/material-design/introduction.html#. [Accessed: 17-Mar-2018].

[14] "Resources - Google Design." [Online]. Available: https://design.google/resources/. [Accessed: 26-Mar-2018].

[15] "tesseract-ocr ·GitHub." [Online]. Available: https://github.com/tesseract-ocr/. [Accessed: 26-Mar-2018].

[16] U. Gargi and R. C. Gossweiler III, "Predictive Text Entry for Input Devices." Google Patents, 2011.

[17] N. Wu, M. Wu, and S. Chen, "Real-time monitoring and filtering system for mobile SMS," in *Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on*, 2008, pp. 1319–1324.

[18] O. Sohaib and K. Khan, "Integrating usability engineering and agile software development: A literature review," in *Computer design and applications (ICCDA), 2010 international conference on*, 2010, vol. 2, pp. V2----32.

[19] N. Condori-Fernandez, M. Daneva, K. Sikkel, R. Wieringa, O. Dieste, and O. Pastor, "A systematic mapping study on empirical evaluation of software requirements specifications techniques," in *Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement*, 2009, pp. 502–505.