

Smart Expense Tracker with OCR

Prof. Harshita C K¹, Bhakti S I², Soumya M Patil³, Varsha M Kori⁴

¹Professor, Dept. of Computer Science Engineering, Jain College of Engineering and Technology Karnataka, India.

^{2,3,4}Student, Dept. of Computer Science Engineering, Jain College of Engineering and Technology Karnataka, India

Email id: harshithajcetcse@gmail.com, bhaktiibrandi@gmail.com, soumyapatil736@gmail.com, varsha.kori.2904@gmail.com

Abstract - Effective personal finance management has become increasingly important in the modern digital environment. However, many existing expense tracking systems still depend on manual data entry, which can be time-consuming and prone to human errors. To overcome these challenges, this work proposes a Smart Expense Tracker that integrates Optical Character Recognition (OCR), voice input, and intelligent data analysis to automate the expense tracking process. The proposed system allows users to upload receipt images, from which the OCR module automatically extracts important information such as the transaction date, amount, and merchant details. In addition, voice commands provide an alternative method for quickly recording expenses without manual typing. The application presents financial insights through interactive visualizations including pie charts and trend graphs, enabling users to better understand their spending behaviour. Additional features such as budget alerts, data export in CSV or PDF format, and AI-based spending insights further enhance financial awareness and decision-making. By combining automation with intelligent analysis, the proposed system simplifies the process of managing personal expenses while improving accuracy and user convenience.

Key Words: Expense Tracking, OCR, Voice Recognition, Firebase, React, Personal Finance, AI Insights, Budget Alerts, Data Visualization.

I. INTRODUCTION

In today's digital society, individuals perform a large number of financial transactions across different platforms such as online payments, credit cards, and cash purchases. As a result, managing personal expenses has become more complex than ever before. Maintaining an accurate record of daily expenditures is essential for budgeting, financial planning, and achieving long-term financial stability.

Traditional approaches to expense tracking, such as maintaining paper records or manually entering transactions in spreadsheets, require significant effort and are often susceptible to mistakes. These methods are not only inefficient but also make it difficult for users to maintain consistent and accurate financial records. Many digital expense tracking applications attempt to simplify this task; however, most of them still rely heavily on manual input, which discourages users from consistently recording their transactions. A major challenge arises when users need to manually enter details from physical receipts. This process is repetitive and time-consuming, and it often results in incomplete or inaccurate records. Automating the extraction of receipt information can significantly reduce this burden and improve the reliability of expense data.

To address these issues, this research presents the design and development of a Smart Expense Tracker, an intelligent web-based application designed to simplify personal expense management. The system integrates Optical Character Recognition (OCR) technology to automatically extract

information from receipt images, reducing the need for manual entry. Users can upload a receipt image, and the system processes it to identify key details such as the vendor name, transaction date, and total amount.

In addition to OCR-based automation, the application supports multiple methods of expense entry including manual input and voice commands, allowing users to interact with the system in a more convenient way. The collected data is stored securely and analysed to generate meaningful insights about spending patterns. Visual dashboards display information through graphs and charts, helping users easily understand their financial behaviour.

The application is implemented using modern web technologies such as HTML, CSS, JavaScript, and React, while Firebase provides backend services including authentication and real-time database functionality. The system architecture, OCR processing pipeline, and experimental evaluation are presented in this paper to demonstrate how automation and intelligent analysis can improve the efficiency and usability of personal finance tracking systems.

II. RELATED WORK

A literature review was performed to examine previous research related to expense tracking systems.

Hemalatha et al. [1] proposed a web-based expense tracker that also utilized Tesseract OCR but focused specifically on handwritten text recognition, using Flask for the backend and Chart.js for visualization. Similarly, Jain et al. [2] reviewed expense trackers, highlighting advanced features like automated data entry, budget limits, and forecasting, but also noted challenges in user engagement and data accuracy.

The concept of using OCR for financial management is a common theme. Hegde et al. [3] emphasized that automated tracking with OCR reduces errors and that machine learning can be used to categorize expenses and analyse spending patterns. They also noted challenges in misclassification and the high computational cost of deep learning techniques for text recognition. Saraboji and Lakshmi [4] surveyed applications using OCR libraries like Tesseract and Google Cloud Vision API to reduce manual input errors, with Python and MySQL as common development tools.

Other works have focused on different aspects of automation. Choughule et al. [6] developed a system using a "Donut model" to extract data from receipts without traditional OCR and integrated SMS logging from bank messages. Pawar et al. [8] and Pandey et al. [9] both presented systems using OCR for receipt scanning and ML for AI-powered sorting, categorization, and graphical reports.

Our project builds on this existing research by integrating multiple innovations into a single, cohesive web application:



(1) a multi-modal input system (OCR, voice, and manual), (2) a robust OCR pre-processing pipeline to improve accuracy, and (3) a modern, responsive web interface built on React and Firebase, providing real-time data synchronization and AI-driven insights.

III. PROPOSED SYSTEM

The proposed Smart Expense Tracker is an AI-driven system that automates data capture and analysis. It allows users to record expenses via manual typing, voice commands, or by uploading a receipt image for OCR processing. The system architecture is visualized in the Data Flow Diagram (Fig. 1) and Use Case Diagram (Fig. 2).

The user interacts with the front-end (built in React) to capture, process, or view data. All data is processed and sent to a secure Firebase Database. This database then feeds the dashboard for visualizations and an analytics engine that detects recurring expenses and generates alerts.

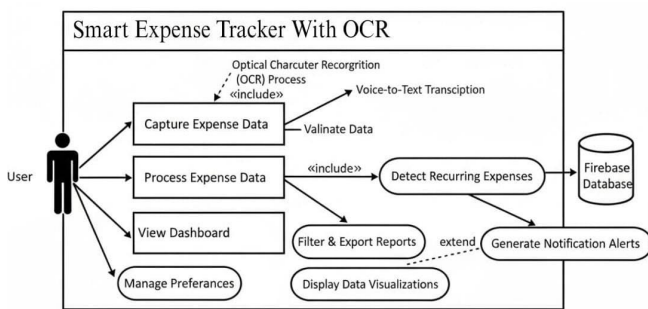


Fig. 1. System Data Flow Diagram.

A. OCR Pre-processing and Data Flow

A critical component of the system is the OCR pre-processing pipeline, which ensures high accuracy from receipt images. When a user uploads an image, it undergoes several automated steps before text extraction:

1. **Grayscale Conversion:** The image is converted to grayscale to reduce colour complexity.
2. **Noise Removal:** Filters are applied to remove artifacts and improve text clarity.
3. **Thresholding:** The image is converted to a binary (black and white) format to create sharp contrast for the text.

After cleaning, the image is passed to the OCR engine (e.g., Tesseract.js), which extracts the raw text. This text is then parsed to identify and filter key data (date, amount, merchant), which is finally structured and stored in the database.



Fig. 2. OCR Pre-processing & Data Flow

B. Detailed Process Flow

The system's operation is divided into three distinct layers:

1. **Data Acquisition Layer:** This layer is responsible for multi-modal expense capture. The primary method is OCR capture, which uses image pre-processing

before calling the OCR API. This is supplemented by a voice input module that uses a real-time speech-to-text service to parse natural language (e.g., "Lunch 25 dollars") into structured data, and a traditional manual input form as a fallback.

2. **Core Processing Layer:** Once data is acquired, it is sent to the API gateway, which validates the user's Firebase authentication token. The data undergoes rigorous validation to check for logical errors (e.g., future dates). Valid data is then passed to an automated categorization engine, which uses a rule-based system and a lightweight machine learning model to assign a category (e.g., 'Food', 'Transport'). Finally, the processed expense data is persisted to the Firebase Database.
3. **Analytical and Output Layer:** This layer transforms the stored data into actionable intelligence. A predictive analytics job runs to identify recurring expenses. The user's dashboard queries the database and performs real-time data aggregation to display visualizations. A proactive alerting service uses the analytical results to send notifications for budget overruns or upcoming bill payments via Firebase Cloud Messaging (FCM).

IV. METHODOLOGY

The research follows a Design Science Research (DSR) approach, focused on creating the application artifact and systematically evaluating its performance to solve the problem of inefficient expense logging. The methodology is structured around system development, detailed OCR pipeline implementation, and rigorous quantitative validation.

A. Three-Tier Serverless Architecture

The Smart Expense Tracker is built upon a modern, three-tier serverless architecture to maximize scalability, reduce latency, and ensure cost-efficiency.

1. **Presentation Tier (Frontend):** The user interface is developed using **React**, ensuring a highly responsive, single-page application (SPA) experience. This tier handles user authentication, data visualization (using charting libraries), and the interface for receipt upload and review. It communicates asynchronously with the Logic Tier via secured API calls.
2. **Logic Tier (Serverless Functions):** All core business logic, including the OCR API calls, image handling, and the data parsing algorithm, resides in **Node.js-based serverless functions** (e.g., Firebase Cloud Functions). This design allows for automated scaling based on user load and isolates the computationally intensive OCR process from the frontend.
3. **Data Tier (Firestore Database):** User data, including individual expense logs, defined categories, user profiles, and application settings, are stored in **Firebase Firestore**. Firestore, a NoSQL, document-based database, was chosen for its real-time synchronization capabilities and native integration with the serverless logic tier, facilitating immediate updates to the user interface upon transaction logging.

B. Intelligent OCR Parsing Pipeline

The success of the application relies entirely on the accuracy of the proprietary parsing logic applied after raw text extraction.

1. **Raw Text Acquisition:** Upon receiving the receipt image, the Logic Tier calls the **Google Cloud Vision API** in batch mode, which returns a comprehensive JSON object containing the full text block and coordinates for every word detected on the receipt.
2. **Targeted Feature Extraction:** The parsing engine prioritizes the extraction of three critical fields:
 - **Total Amount:** A complex set of Regular Expressions is applied, targeting keywords such as "TOTAL," "GRAND TOTAL," "BALANCE DUE," and "AMOUNT PAID." The algorithm prioritizes the largest numerical value associated with these keywords, filtered to match standard currency formats.
 - **Transaction Date:** The system scans for patterns matching common date formats (DD/MM/YY, MM-DD-YYYY, etc.), prioritizing dates that appear near the top or bottom of the receipt.
 - **Vendor Name:** The vendor name is often located at the very top of the receipt. The parser uses bounding box information to isolate the top lines of text, comparing these lines against a custom dictionary of common vendor names and filtering out generic information (e.g., "VAT Registration," "Tax No").
3. **Categorization Heuristics:** Initial categorization is achieved using a **keyword-matching heuristic**. Keywords found in the vendor name or line-item descriptions (e.g., "Starbucks," "Coffee," "Latte" map to 'Food & Drink'; "Amazon," "Online" map to 'Shopping') are used to assign a preliminary expense category.

C. Experimental Validation Protocol

The validation process involves two quantitative experiments to demonstrate both the technical efficacy and practical utility of the Smart Expense Tracker.

1) OCR Extraction Accuracy Test

1. **Dataset Creation:** A controlled dataset of unique, real-world receipts was curated. The dataset includes a variety of formats (thermal, inkjet, printed), quality levels (clear, wrinkled, slightly faded), and vendor types.
2. **Ground Truth Establishment:** Each of the 100 receipts was manually logged into a database to establish the Ground Truth for the Total Amount, Transaction Date, and Vendor Name.
3. **Accuracy Calculation:** Each receipt was processed by the Smart Tracker. The results were compared against the Ground Truth to calculate the Extraction Accuracy () for each of the three critical fields.

2) Time Efficiency Comparative Study

1. **Participant Setup:** A group of non-expert users was recruited and divided into two groups: a control group (Manual Entry) and an experimental group (Smart Tracker).

2. **Manual Entry ():** The control group manually entered the three critical fields (Amount, Date, Vendor) for a set of 20 receipts into a standard spreadsheet application. The total time taken for
3. **Smart Tracker Time ():** The experimental group performed the same task using the Smart Tracker (uploading the image and performing any necessary quick edits/confirmations). The total time was measured and averaged per receipt.
4. **Smart Tracker Time ():** The experimental group performed the same task using the Smart Tracker (uploading the image and performing any necessary quick edits/confirmations). The total time was measured and averaged per receipt.
5. **Efficiency Calculation:** The time savings were calculated to demonstrate the increase in efficiency achieved by the automated system. Efficiency Increase (%) = $\frac{T_m - T_s}{T_m} \times 100\%$.

V. RESULTS

The developed system was tested, and the results confirmed its functionality across all specified objectives. The user workflow is demonstrated in Fig. 3 through Fig. 8.

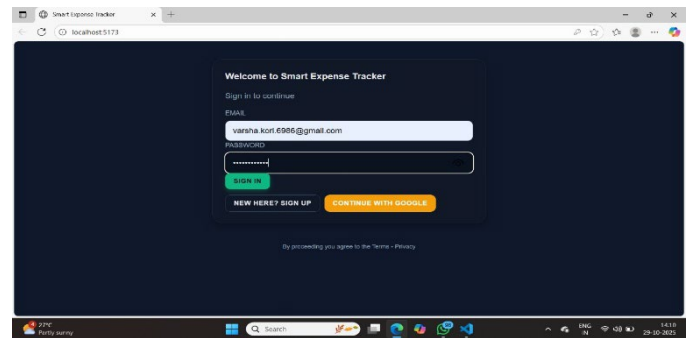


Fig. 3. Home page with secure user authentication

Fig. 3 shows the application's main login and registration page, which is secured using Firebase Authentication.

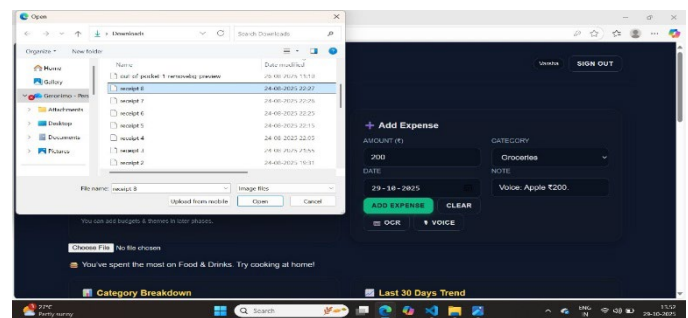


Fig. 4. Selecting a receipt image for OCR processing

As shown in Fig. 4, the user can add an expense by selecting an image file. The system also shows the manual and voice entry options. Upon upload, the OCR engine processes the image and populates the fields.

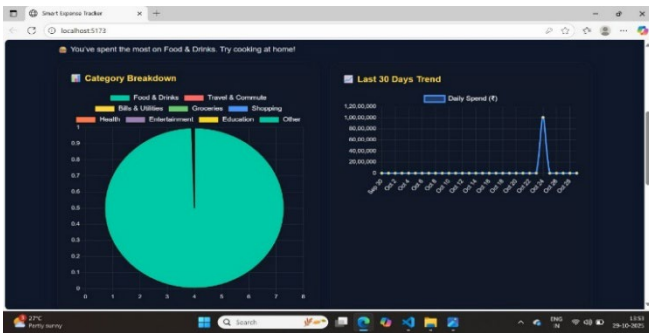


Fig. 5. Dashboard visualization (Category Breakdown and 30-Day Trend)

After data is entered, the dashboard (Fig. 5) provides immediate visual feedback. It displays a pie chart for categorical spending and a line graph for tracking spending trends over time, helping users easily interpret their financial habits.

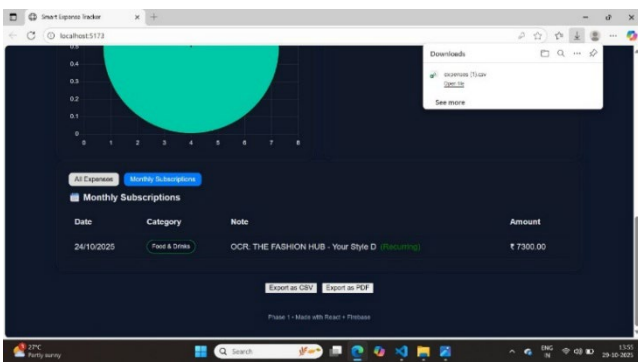


Fig. 6. Recent Expenses and Transaction Auditing Interface

The system provides a detailed list of all recent transactions (Fig. 6), allowing users to review, edit, or delete entries. This interface is crucial for auditing and correcting any potential OCR inaccuracies.

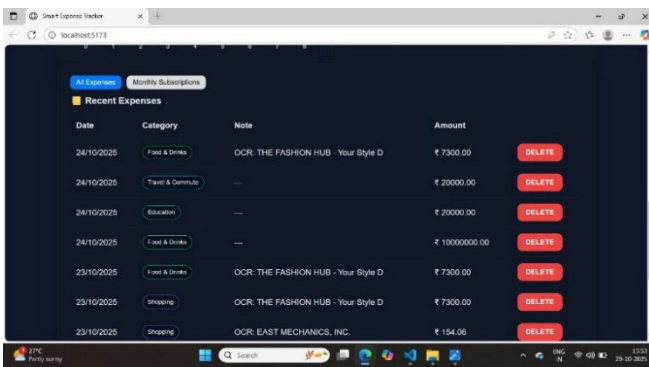


Fig. 7. Monthly Subscriptions detection and Export feature (CSV/PDF)

Fig. 7 demonstrates the system's ability to identify recurring transactions and list them as "Monthly Subscriptions". This screen also shows the export functionality, allowing users to download their data.

A key AI-driven feature is the proactive alert system. Fig. 8 shows a notification warning the user that they have crossed a predefined spending limit, helping to enforce budget adherence.

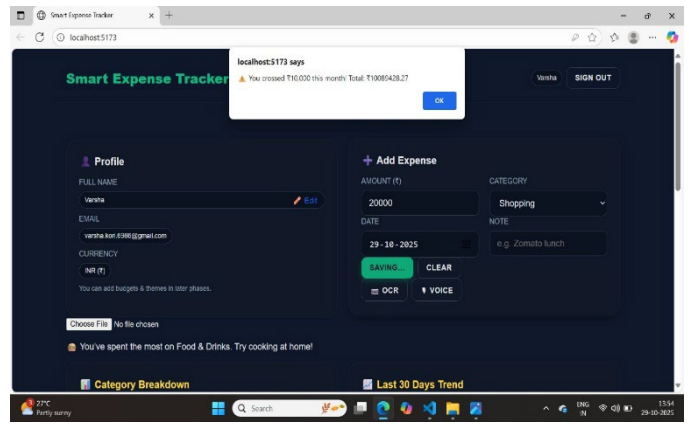


Fig. 8. Proactive Spending Alert and Notification Delivery

VI. CONCLUSION AND FUTURE WORK

A. Conclusion

By integrating a multi-modal input system with a primary focus on OCR, the application significantly reduces manual data entry, thereby enhancing user efficiency and data accuracy. The system successfully stores and validates data using Firebase, and the front-end reporting module provides users with valuable, actionable insights through visualizations, automated summaries, and proactive spending alerts. This work validates the feasibility of using integrated AI and OCR technologies to bridge the gap between physical receipts and digital financial records, providing a scalable and intelligent companion for personal finance management.

B. Future Work

While the current system provides a robust foundation, several enhancements are planned for future iterations:

- **Native Mobile Applications:** Develop dedicated iOS and Android apps for a more seamless, on-the-go OCR capture experience.
- **Bank Synchronization:** Integrate with financial APIs (e.g., Plaid) to automatically sync transactions from bank and credit card accounts for a holistic financial overview.
- **Offline Functionality:** Implement local storage to allow users to capture expenses even without an internet connection, syncing the data when connectivity is restored.
- **Shared Budgets:** Introduce collaborative features for couples or teams to manage shared finances and track group expenses.
- **Advanced OCR:** Further refine the OCR engine to improve accuracy on low-quality images and handwritten receipts.

REFERENCES

- [1] S. Hemalatha, T. Harish, M. Jagadeeshwaran, and S. Maniraja, "Web-Based Expense Tracker with Handwritten Text Recognition using Tesseract OCR," International Journal of Advanced Research in Science, Communication and Technology, Jan. 2025.
- [2] N. Jain, D. Mishra, A. Sahani, and H. Prajapati, "Expense Trackers," International Journal of Advanced Research in Science, Communication and Technology, Jan. 2025.
- [3] S. R. Hegde, H. Gangatkar, P. V. G., Hayavadana M. B, and Sudha M, "Automated Expense Tracking with OCR: Enhancing Financial

Management through Technology," International Advanced Research Journal in Science, Engineering and Technology, vol. 12, no. 1, Jan. 2025.

- [4] A. Saraboji and S. Lakshmi, "Expense Tracker Application," International Advanced Research Journal in Science, Engineering and Technology, vol. 11, no. 6, pp. 144–146, June 2024.
- [5] M. Mistry, S. Dhomne, and P. Gomase, "An Expense Tracker," International Research Journal of Engineering and Technology (IRJET), vol. 11, no. 4, April 2024.
- [6] R. Choughule, A. Parmar, K. Devasi, C. D. Nagane, M. Wasnik, and I. Bharambe, "Expense Tracking System," International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol. 9, no. 3, June 2023.
- [7] M. Sakthivel, P. Roshini, K. Roja, P. Maha Lakshmi, and V. Keerthi, "Personal Expense Tracker Application," International Journal of Creative Research Thoughts (IJCRT), vol. 11, no. 5, May 2023.
- [8] A. Pawar, V. Nikam, Y. Markad, C. Pandkar, and S. Bong, "Expense Management System Using OCR," International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 11, no. 5, May 2023.