RECEIPTVAULT AI: AN OCR-BASED DIGITAL RECEIPT AND WARRANTY MANAGEMENT SYSTEM

[1]Mano Shree S. P., [2]Karthika Kandaswamy, [3]Xavier Mary
[1,2,3]Department of Computer Science and Engineering,
[1,2,3]Adhiyamaan College of Engineering (An Autonomous Institution), Hosur, Tamil Nadu, India
[1]manoshree587@gmail.com, [2]karthikakandaswamy20@gmail.com, [3]marybe.90@gmail.com

Abstract : Paper receipts contain information needed for product returns, repairs, and warranty claims, but they can be lost, damaged, or difficult to retrieve. This paper presents ReceiptVault AI, a web-based digital receipt and warranty management system designed to simplify receipt storage and information retrieval. Users can upload receipt images, which are processed using Optical Character Recognition (OCR). EasyOCR is used to recognize text, while keyword matching and predefined patterns are applied to identify information such as store name, product name, purchase date, price, warranty period, and contact number. The application is developed using Python and Flask for backend processing, HTML, CSS, and JavaScript for the user interface, and MySQL for data storage. Preliminary functional testing was carried out using three receipt samples. The observed outcome was partial field extraction: some target information was identified, while other fields were not consistently extracted. The available test records do not provide field-by-field counts or a quantitative accuracy measurement. The results therefore demonstrate the basic feasibility of the workflow for the tested samples, while highlighting the need for broader receipt-layout testing and systematic evaluation.
IndexTerms - Optical Character Recognition, Digital Receipt Management, EasyOCR, Flask, MySQL, Warranty Management, Keyword-Based Information Extraction.

I. INTRODUCTION

Receipts provide essential information about purchases, including the store, purchased product, transaction date, amount paid, and sometimes warranty details and service contact information. These details may be required when requesting repairs, replacements, or warranty services. However, paper receipts can be misplaced, become difficult to read, or require considerable time to locate.

Digital receipt management provides an alternative by storing purchase documents electronically and making their information easier to retrieve. Optical Character Recognition (OCR) can convert text contained in receipt images into machine-readable text. Nevertheless, extracting specific information from receipts remains challenging because the positions, labels, fonts, and arrangements of fields can differ between receipt designs.

ReceiptVault AI is developed as a web-based application for uploading, processing, and managing digital receipts. It uses OCR to recognize text in receipt images and applies keyword matching and predefined patterns to identify relevant receipt information before storing the available details for later access. The current prototype has been tested with three receipts following a specific layout. Its purpose is to demonstrate the integration of receipt upload, OCR-based extraction, and digital record management rather than to claim universal receipt recognition.

A. Objectives

- To provide a centralized platform for storing digital receipts.
- To extract relevant receipt information using OCR.
- To store and retrieve receipt details through a database.
- To support the storage and retrieval of warranty information recognized from uploaded receipts. □ To evaluate the prototype on a small set of receipts and identify its current limitations.

II. RELATED WORK

A. OCR-Based Text Recognition

Optical Character Recognition has been studied extensively for converting printed documents into machine-readable text. Smith presented an overview of the Tesseract OCR engine in 2007, describing its text-line detection, classification methods, and adaptive recognition techniques [1]. OCR engines can reduce manual transcription by converting printed text into digital text. However, recognizing text does not automatically guarantee that every extracted value is assigned to the correct semantic field.

B. Text Detection in Complex Layouts

Baek et al. introduced Character Region Awareness for Text Detection (CRAFT) in 2019 [2]. The approach detects character regions and relationships between characters to identify text in different shapes and orientations. Such research highlights the importance of text localization when documents contain irregularly positioned or visually complex text.

C. EasyOCR

EasyOCR is an open-source OCR library that provides text detection and recognition for multiple languages [3]. It can return recognized text together with bounding-box coordinates and confidence values. ReceiptVault AI uses EasyOCR as the textrecognition component of its prototype. The application then applies application-level extraction logic to identify fields such as purchase date, price, warranty period, and contact number.

D.Research Gap

OCR systems can recognize text without necessarily understanding the role of each value in a receipt. Receipt information extraction therefore requires additional logic that identifies relevant fields from recognized text. The current prototype addresses this practical requirement for a specific receipt layout. Its limitation is that the extraction rules have not yet been demonstrated to generalize across diverse receipt formats.

III. Proposed System and Methodology

A. System Overview

ReceiptVault AI is a web application that combines receipt-image uploading, OCR processing, field extraction, and database storage. A user uploads a receipt image through the interface. The backend processes the image, obtains recognized text, identifies target fields using keyword matching and predefined patterns, and presents the information identified by the extraction process. The receipt details can then be stored and retrieved through the application. The main workflow is shown in Fig. 1.

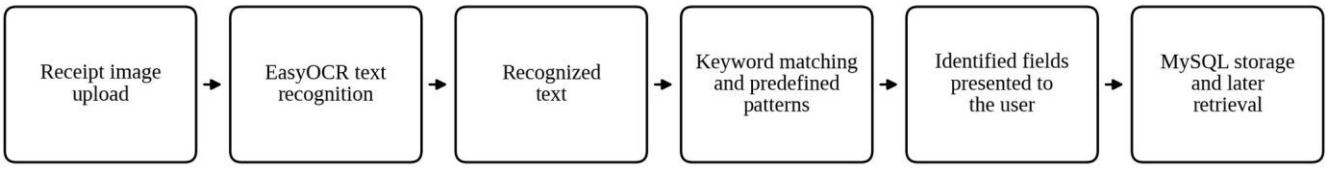


Fig. 1. Workflow of the ReceiptVault AI prototype.

B. System Modules

1) *User Registration and Login:* The web application provides user registration and login functionality. Flask handles the application routes and session-based access to protected pages.

2) *Receipt Image Upload:* Users upload receipt images through the application. The backend receives the selected image and processes it using the configured image-handling workflow.

3) *OCR Processing:* EasyOCR recognizes text from the uploaded receipt image. The recognized text forms the input to the fieldextraction stage.

4) *Receipt Field Extraction:* The application applies keyword matching and predefined patterns to the recognized text to identify relevant information, including store name, product name, purchase date, price, warranty period, and contact number. The extraction outcome may be partial because the rules depend on the wording and layout of the receipt.

5) *Warranty Information:* The system attempts to identify warranty information when it is present and recognized on the receipt. The current application does not provide manual entry of missing warranty details; therefore, warranty information that cannot be extracted may remain unavailable in the saved record.

6) *Database Storage and Retrieval:* MySQL stores the relevant receipt records and associated information. The application provides a way to access saved records through its web interface. *C. Technologies Used*

Table I. Technologies used in ReceiptVault AI

Component	Technology	Purpose
Backend	Python and Flask	Application logic and web routes
OCR	EasyOCR	Recognition of text in receipt inputs
Frontend	HTML, CSS, JavaScript	User interface
Database	MySQL	Storage and retrieval of receipt records
Input formats	Receipt images	Receipt image upload

D.Field Extraction Process

The extraction process consists of the following steps: (1) receive the uploaded receipt image; (2) process the image using the configured OCR workflow; (3) obtain the recognized text; (4) apply keyword matching and predefined patterns to the recognized text; (5) identify the receipt fields that can be matched; (6) present the identified information to the user; and (7) store the available receipt details for later retrieval.

The field-extraction stage depends on both the OCR output and the rules used to identify each field. If a receipt uses different labels or positions, the current rules may fail to identify a value even when some of its text has been recognized correctly.

IV. Implementation and Experimental Results

A. Implementation

The prototype is implemented as a web application using Flask and MySQL. The frontend allows users to interact with the application, upload receipt images, and view the information identified by the extraction process. EasyOCR performs text recognition, while keyword matching and predefined patterns are used to identify target fields from the recognized text.

The information identified by the extraction process is associated with a digital receipt record so that it can be accessed later. The current application does not support manual entry of missing warranty information.

B. Test Setup

The preliminary functional evaluation used three receipt samples. The evaluation examined whether the application could recognize text and identify the target receipt fields using its current keyword-matching and predefined-pattern rules. The observed outcome was partial field extraction: some target fields were identified successfully, while others were not consistently extracted.

This evaluation was a small functional test, not a controlled benchmark of OCR or field-extraction accuracy. The available test notes do not provide field-by-field success counts for each receipt. No independent large dataset, character error rate, word error rate, precision, recall, or cross-layout comparison was used in this preliminary evaluation.

C. Observed Results

Table II. Reported field-extraction observations for the tested receipt layout

S. No.	Evaluation Item	Reported Observation
1	Receipt samples	Three receipt samples were tested
2	Text recognition and field extraction	Some target fields were identified; extraction was partial
3	Field-level results	Individual field success counts for each receipt were not recorded in the available test notes
4	Quantitative accuracy	Not calculated; verified field-by-field ground-truth results were unavailable

The preliminary tests show that the prototype can identify some target receipt fields from the three tested samples. However, extraction was partial, and the available records do not establish which fields were successfully identified in each individual receipt or how consistently they were extracted across repeated runs. The results also do not establish performance on receipts with different layouts.

No numerical accuracy percentage is reported because field-by-field ground-truth values and corresponding extraction results have not been documented in a form that supports quantitative calculation.

D.Discussion

The prototype integrates OCR with application-level keyword matching and predefined patterns to convert recognized receipt text into structured information. This is useful because recognized text alone does not indicate which values represent the store, product, date, price, warranty, or contact information. The preliminary tests indicate that the current rules identify some fields, but do not consistently extract all target information.

The principal limitations are partial field extraction and dependence on the wording and layout of the tested receipts. Different receipts may use different labels or place the total price, date, product description, warranty information, or contact details in different positions. Variations in image quality, orientation, font size, and printing quality may also affect OCR output. In addition, the application does not currently allow users to enter missing warranty details. The results should therefore be interpreted as preliminary functional observations rather than evidence of general-purpose receipt understanding.

V. Limitations and Future Enhancements

The current prototype has several limitations:

- *Limited Test Dataset:* Only three receipts were tested, which is insufficient to establish robust performance.
- *Layout Dependency:* The field-extraction logic is designed around a specific receipt format and may not work reliably with other layouts.
- *Limited Performance Measurement:* A formal accuracy evaluation based on manually verified reference values has not yet been completed.
- *OCR Errors:* Poor image quality, unusual fonts, and unclear printing may cause recognition errors.
- *Missing Warranty Information:* Warranty details may be absent from a receipt or may not be recognized. The current application does not provide a manual-entry option for missing warranty information.

Future work will focus on testing a larger collection of receipt images from different stores, improving keyword-matching and predefined-pattern rules, and introducing more flexible methods for identifying receipt fields. A manually verified test dataset should be prepared to calculate field-level precision, recall, and accuracy. Additional work could investigate image preprocessing, confidence-based validation, and a user-correction mechanism for incorrectly extracted values. A manual warranty-entry feature could also be considered as a future enhancement.

VI. Conclusion

ReceiptVault AI is a web-based digital receipt and warranty management prototype that combines receipt-image uploading, OCRbased text recognition, keyword- and pattern-based field extraction, and MySQL storage. The application is developed using Python, Flask, HTML, CSS, JavaScript, and EasyOCR. It attempts to identify store name, product name, purchase date, price, warranty period, and contact number from uploaded receipt images. The current application does not support manual entry of missing warranty information.

Preliminary functional testing was conducted using three receipt samples and resulted in partial field extraction: some target information was identified, while other fields were not consistently extracted. Because field-by-field success counts and a larger verified dataset are not available, no numerical accuracy claim is made. The prototype should not yet be considered a general solution for arbitrary receipt formats. Future work will focus on expanding the test dataset, improving extraction rules, adding support for missing information where appropriate, and conducting a formal quantitative evaluation.

ACKNOWLEDGMENT

The authors would like to acknowledge the support and guidance provided by the Department of Computer Science and Engineering, Adhiyamaan College of Engineering, Hosur, Tamil Nadu, India, during the development of the ReceiptVault AI prototype.

REFERENCES

[1] R. Smith, "An overview of the Tesseract OCR engine," in Proc. 9th Int. Conf. Document Analysis and Recognition (ICDAR), 2007, pp. 629–633, doi: 10.1109/ICDAR.2007.4376991.

[2] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2019, pp. 9365–9374, doi: 10.1109/CVPR.2019.00959.

[3] JaidedAI, "EasyOCR: Ready-to-use OCR with 80+ supported languages." [Online]. Available: https://github.com/JaidedAI/EasyOCR. Accessed: Oct. 29, 2026.

Copyright & License:
© Authors retain the copyright of this article. This work is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.