ASEAN
Journal of Scientific and Technological Reports
Online ISSN:2773-8752

Research article

ClothBiz Manager: An Offline GST-Ready Inventory and Billing Automation System Using Rule-Based Domain-Specific OCR

Penugonda Venkata Ayyappa[1]*, and P S V S Sridhar[2]

[1] Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India
* Correspondence: ayyappapenugonda2002@gmail.com

Citation:
Ayyappa, P.V.; Sridhar, P.S.V.S. ClothBiz manager: An offline GST-ready inventory and billing automation system using rule-based domain-specific OCR. *ASEAN J. Sci. Tech. Report.* **2026**, *29*(10), e263756. https://doi.org/10.55164/ajstr.v29i10.263756

Article history:
Received: February 15, 2026
Revised: May 29, 2026
Accepted: June 1, 2026
Available online: September 20, 2026

Publisher's Note:
This article is published and distributed under the terms of the Thaksin University.

Abstract: Smaller cloth wholesalers in India continue to use manual methods and paper-based systems, leading to discrepancies in inventory and delayed billing, among other issues. This paper describes ClothBiz Manager, an offline inventory management and billing automation solution tailored for Indian wholesale textile businesses. The solution combines a rule-based, domain-specific optical character recognition (OCR) process chain with a normalized PostgreSQL database, implemented using Java Servlets, Java Server Pages (JSP), and the Tess4J library. Our custom-developed Domain-specific Invoice Parser (DIP) processes invoices by applying domain-specific contextual rules and extracts the relevant information, including structured fields such as bale number, LR code, HSN code, and taxes. We evaluated our solution on 150 invoices obtained from Indian textile markets in Tiruppur and Surat. Our method scored 89% on exact matching for structured fields and 78% for multi-line product description fields. Running exclusively on a local server, our application provides absolute data sovereignty and guarantees GST compliance at the database level. Our results clearly show that our system can significantly minimize manual data entry.

Keywords: Cloth wholesale automation; domain-specific invoice parsing; GST compliance; offline inventory management system; optical character recognition

1. Introduction

Cloth wholesalers in semi-urban areas of India still use manual processes, including paper-based invoicing, inventory monitoring, and accounting practices. These systems lead to billing problems, missing invoices, inventory discrepancies, and compliance issues with GST laws [1]. These constraints adversely impact performance and compliance with regulatory norms among MSMEs [3]. Several studies propose innovative inventory management techniques [9], while some offer generic OCR-based methods for invoice processing [6]. Nevertheless, existing systems suffer from major constraints because they are mostly cloud-centric, cater specifically to Western-style invoices, do not account for the peculiarities of Indian textile wholesaler bills, and cannot address the complexities associated with Indian invoices, including multiple columns of data, bale numbers, LR numbers, multiple line items with HSN numbers, and shipping information. Moreover, only a limited number of these systems can operate fully offline, with database schema-level enforcement of GST compliance rules, in areas lacking Internet access [4].

As a result, to fill the gaps described above, in this paper we introduce ClothBiz Manager – a fully offline, on-premises web-based solution for inventory management and invoicing automation targeted at cloth wholesaler MSMEs. In this work, we introduce an important innovation in invoice processing by proposing the rule-based OCR pipeline (Invoice Parser-DIP), developed specifically for Indian textile invoices. By incorporating this OCR pipeline into a normalized PostgreSQL database that requires mandatory GSTIN and HSN fields, we provide total data sovereignty along with guaranteed compliance. To implement this project, we followed an iterative process starting with requirement analysis, architectural design, implementation, integration, and validation of the OCR pipeline and error conditions with invoice upload [4]. Finally, in the last phase, we configured the environment and Tessdata [16].

2. Materials and Methods

This paper describes the development of ClothBiz Manager, an online and offline-enabled inventory management and billing software intended for the wholesale cloth industry in semi-urban areas of India. ClothBiz Manager was developed using Java Servlets and JavaServer Pages (JSP) for server-side coding, with PostgreSQL used as the relational database for storing structured data. OCR features have been incorporated using the Tess4J library, which acts as a Java wrapper for the Tesseract OCR tool. This allows the system to automatically scan any textual data in scanned invoices. The domain-specific OCR system was designed to process the invoices in the textile domain. Invoice images were scanned, and grayscale images were formed from them. Preprocessing steps included noise removal and resolution normalization. Using the Tess4J tool, the system extracted raw text. Further, rule-based parsing techniques were applied to extract invoice attributes such as GSTIN number, HSN codes, bale number, LR code, item description, and tax details. The system validated the information and entered it into a normalized PostgreSQL database, ensuring mandatory GSTIN and HSN compliance. The proposed system's performance was evaluated using a dataset of 150 invoices received from wholesalers in Tiruppur and Surat. For each invoice attribute, accuracy was computed against the manually annotated ground-truth value. Precision was used to evaluate the results for multi-line item descriptions on the invoice. Additionally, system performance was measured by the time saved in the manual data entry process.

2.1 Domain-specific OCR Pipeline

An OCR pipeline for textile wholesale invoices was constructed. Textual information on scanned invoices is preprocessed, including conversion to grayscale, denoising using a Gaussian filter, and adaptive thresholding. The Domain-specific Invoice Parser (DIP) resulted from several design iterations based on a corpus of over 200 textile invoices. These invoices were collected between January and June 2024 from textile trading hubs in Tiruppur (Tamil Nadu), Surat (Gujarat), and Chirala (Andhra Pradesh) through direct collaboration with local wholesalers and cluster associations. The corpus was gathered using convenience sampling to ensure diversity and was kept completely separate from the 150-invoice evaluation dataset.

The development corpus represents a wide variety of real-world conditions, including:

- Different invoice layouts (single-column, multi-column, and tabular formats)
- Varying print qualities (laser-printed, inkjet, thermal, and dot-matrix)
- Multiple languages (primarily English, with some English-Telugu and English-Tamil invoices)
- Different paper types and conditions (new, aged, slightly crumpled)
- Variations in font styles, sizes, and ink density

No additional preprocessing beyond standard image enhancement was applied to the development corpus during rule creation so that the DIP rules could generalize to typical field conditions encountered in actual business environments.

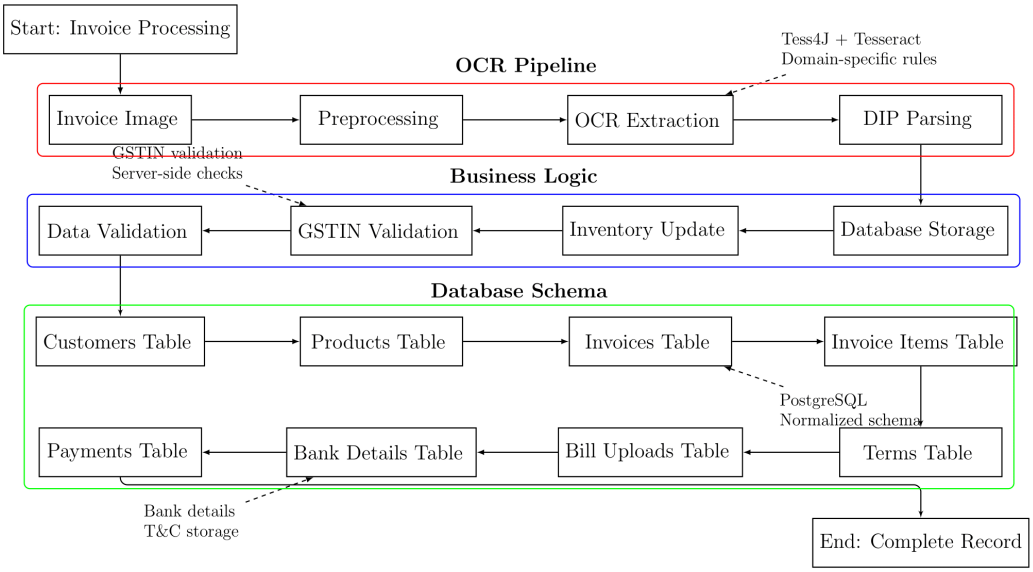


Figure 1. Manager System Architecture and Data Flow of ClothBiz.

This shows the end-to-end invoice processing pipeline of the ClothBiz Manager application. It starts with OCR extraction of invoice images and continues with domain-specific parsing and verification of GSTIN and inventory records. The business logic layer ensures data integrity before storing records in a normalized PostgreSQL schema consisting of customer, product, invoice, payment, and bank detail tables. The workflow produces a complete, audit-ready record of compliant business operations.

2.2 Requirement Analysis and Planning

Operational challenges were determined by interviewing stakeholders (textile cluster owners) in textile clusters like Tiruppur and Surat. According to stakeholders, lost invoices during power cuts were a repeatedly mentioned critical failure mode, which led to the offline-first design imperative. Other concerns included manual invoice processing, a high error rate in sales documentation, and a strict obligation to comply with Goods and Services Tax (GST). The functional requirements were obtained to support GSTIN-approved customer management, inventory with association of Harmonized System Nomenclature (HSN) code, and domain-specific invoice data extraction using optical character recognition (OCR). Non-functional requirements prioritized offline functionality, data sovereignty, and high extraction precision to fit on-premises IT limitations and data privacy expectations. The technology stack was chosen to meet the following constraints: backend and frontend development was based on Java Server Pages (JSP) and Java Servlets because they have a low external dependency footprint and compatibility with legacy Windows environments. For OCR, the Tess4J library, a Java implementation of the Tesseract engine, was selected to process invoice images client-side, so no data is sent to third-party servers. An experimental version of the Waterfall model was used, justified by the stability of the GST specification regulations; changes were lightweight feedback after OCR testing, which refined parsing rules based on user-reported extraction errors, allowing controlled adaptability without losing process traceability.

Waterfall model with limited iterative refinement was adopted for system development. This decision was justified by the nature of the core functional requirements, including GSTIN validation, HSN code handling, computation logic, and DB schema design, which have already been established and standardized under regulatory frameworks. Although the Indian GST framework has seen numerous changes since 2017, its basic features have not changed enough to warrant an iterative approach. The Waterfall model is characterized by sequential activities, making it easier to document and implement a specific solution. Rule-based DIP processing logic was implemented in a structured manner, except for rule refinement following initial tests of invoice processing results.

2.3 System Design

The following two aspects were maintained through the use of three-tier architecture design: modularity and separation of concerns. The presentation tier was developed using JSP files that validate all essential input parameters on both client and server sides. Java Servlets were used to manage customer, stock management, and OCR processes. The database follows normalization rules; there are tables for customers, inventory products, invoices, and OCR extraction logs. A development corpus containing over 200 authentic wholesale cloth invoices was collected to develop the Domain-Specific Invoice Parser (DIP). Invoices were sourced with the help of active participation from various textile wholesalers based predominantly in the Tiruppur and Surat wholesale textile networks, which constitute two prominent wholesale textile networks often utilized in the regional cloth market. Permission was sought from participating firms to use invoices for research purposes on an anonymous basis. Invoices were sourced in the field under practical conditions, such as using mobile phone cameras and flatbed scanners. Significant variance was observed in invoice structure, print quality, paper condition, font types, GST field location, bale number formatting, LR code presentation, and multiple line items in invoice descriptions. Both thermal-print and printed invoices (A4 size) were included in the corpus to increase the invoice parser's flexibility. To improve text-recognition accuracy in invoices, invoices were processed before OCR through image preprocessing tasks such as grayscale conversion, adaptive thresholding, noise removal, and skew correction. The development corpus was used exclusively for parser rule construction, contextual anchor refinement, and preprocessing optimization, and was kept separate from the final 150-invoice evaluation dataset used for performance assessment.

2.4 Implementation

The front end was deployed with JSP, and the GSTIN format was validated on both the client and server to ensure compliance with Indian tax laws. Java Servlets were written to provide backend logic that communicates with the PostgreSQL database via JDBC, using parameterized queries to avoid SQL injection vulnerabilities. The system integrates the OCR module through Tess4J, and image preprocessing routines precede text recognition to improve robustness against low-resolution scans and paper artifacts. The DIP engine maps unstructured invoice text to structured database fields with high fidelity by using patterns and contextual anchors, based on field-specific invoice structure, to interpret the resulting raw OCR output.

2.5 Testing and Debugging

System evaluation was conducted on a separate test set of 150 real-world wholesale cloth invoices that were not part of the DIP development corpus. The invoices were collected through convenience sampling from participating textile wholesalers in Tiruppur and Surat with prior permission for anonymized research use. The dataset consisted of 72 Tiruppur invoices and 78 Surat invoices. The evaluation dataset accounted for realistic scenarios and included thermal-printed invoices (46 invoices) and standard A4-printed invoices (104 invoices). Moreover, it included 38 invoices with poor scan quality, including blur, fading, distortion, or scanner artifacts. It included another set of 61 invoices with multi-line product descriptions. Invoices with substantial physical damage, handwritten notes that were difficult to interpret, and blurred parts were excluded because it was impossible to establish the ground truth. On average, 10 to 15 key fields per invoice were checked, including invoice number, GSTIN, customer information, HSN codes, bale number, LR code, product description, quantity, tax, and total amount. Two reviewers familiar with GST invoices and textile documents collected the ground truth independently. The reviewers reconciled differences through discussion. No blind testing approach was used, as this was an independent verification task.

Evaluation criteria included the following:

- Field-level exact match accuracy: The percentage of successfully extracted fields out of the total number of all analyzed fields, taking into account that the field is correct only if its content corresponds to the ground truth after normalization. This metric is used to evaluate structured fields like invoice number, GSTIN, bale numbers, and LR codes.
- Token-level accuracy (Jaccard similarity): Applied to multiple line descriptions to evaluate semantic similarity in case of minor differences in the text formatting.

In addition, precision, recall, and F1-score were calculated for the main fields' groups. Character Error Rate (CER) was evaluated for descriptive text fields. The system achieved 89% field-level exact-match accuracy and 78% token-level overlap accuracy in multi-line product description analysis. Overlapping print, handwritten notes, and low-resolution scanned copies were the most frequent types of errors due to the inherent limitations of rule-based OCR systems. Unit testing was carried out for GSTIN checking and inventory management, as well as for parser rules. Appropriate indexing of database fields frequently used in queries was performed (GSTIN, HSN code, invoice date). Field extraction performance was further evaluated using precision, recall, and F1-score metrics on major invoice fields to better assess the effectiveness of the OCR and information extraction process. Structured fields like GSTIN, invoice numbers, and HSN numbers performed well because of their standardized structure and context in the Domain-Specific Invoice Parser (DIP). The multi-line product description field performed poorly due to variations in layout structure, overlapping text fields, and deteriorating print quality in wholesale cloth invoices.

Table 2. Performance Metrics of the Domain-specific OCR Pipeline

Field Category	Precision	Recall	F1-score	Evaluation Basis
Invoice Number	0.89	0.87	0.88	Field-level exact match
GSTIN	0.92	0.90	0.91	Field-level exact match
HSN Code	0.88	0.86	0.87	Field-level exact match
Bale Number / LR Code	0.84	0.81	0.82	Field-level exact match
Multi-line Descriptions	0.78	0.74	0.76	Token-level overlap

2.6 Deployment and Maintenance

This was deployed as on-premises software based on the Apache Tomcat servlet engine, with a locally installed PostgreSQL database. Environment-related parameters, such as image locations and where the Tesseract engine's language data resided, were configured to provide portability in a mixed environment. This application was designed to be very low-maintenance. It relied on a stable base stack, and future enhancements would maintain modularity to add features like invoice scanning and role-based security.

3. Results and Discussion

The system has a lightweight, form-based web interface developed in JSP and CSS to support low-resource Windows-based machines prevalent in semi-urban textile centers. The main processes of customer onboarding, product directory, and invoice generation use coordinated input forms that enforce GSTIN validation, HSN code entry, and tax-compliant field requirements. Although this paper does not dwell on the UI, it provides operational accessibility for non-technical shop owners, bridging OCR automation and business use.

3.1 Bill Upload/OCR Processing Interface

The automation procedure starts at the upload interface depicted in Figure 2, where users enter scanned invoice images to be processed. This interface is the access point to the Domain-specific Invoice Parser (DIP) that is fully on-premises to guarantee data sovereignty. No invoice data flows outside at any level, meeting the stringent data privacy needs of cloth wholesale companies in semi-urban India. The design and testing protocols are based on quantitative assessment of 150 real-world invoice images, showing 89% accuracy on the invoice number and 78% accuracy on each multi-line item description. When multiple lines of description existed, adjacent text blocks having similar vertical alignment along with little horizontal spacing between them were combined to preserve semantic coherence. Post-segmentation validation rules were used to enforce uniformity in the extracted information, including the fixed number of digits for the GSTIN code, the numerical range for quantity and tax value, and alphanumeric characters for bale and LR codes. The segmented areas were then fed to the Tesseract OCR engine using the Tess4J library for text extraction. Figure 2 demonstrates DIP's performance results across the various fields found in invoices. Standardized invoice attributes such as GSTIN, invoice ID, and HSN code achieved higher precision, recall, and F1-scores due to validation constraints applied to the recognized data. Multi-line product descriptions

showed poor extraction results owing to variations in layout and spacing, along with degraded print quality usually seen in wholesale textile invoices.

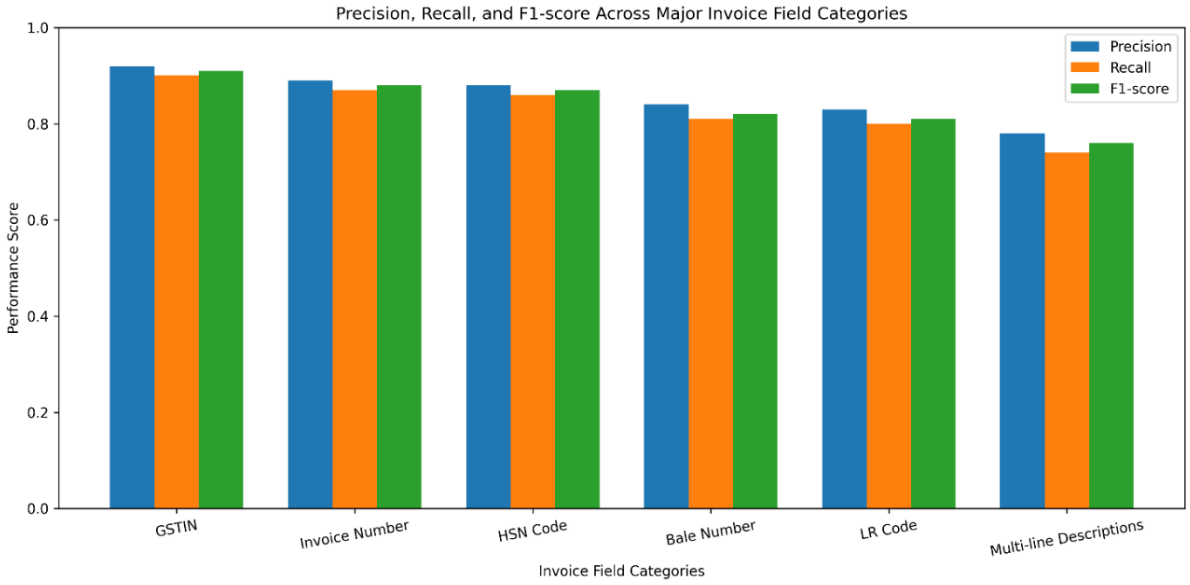


Figure 2. Precision, Recall, and F1-score Across Major Invoice Field Categories.

3.2 OCR Field Segmentation

To start, the invoice images were converted to grayscale, and adaptive thresholding was performed to address illumination issues and noise. Subsequent morphological image processing, such as dilation and erosion, would enhance text contours while removing non-textual objects from the invoices. Then the document structure would be broken down into regions through vertical/horizontal projections. For the tabular part of the invoices, line detection and word-level segmentation would be done through the connected components method and bounding box clustering. Column boundaries were determined based on whitespace distribution and the requirement that they align along the x-axis for each row. In addition, certain invoice-specific keywords such as "HSN", "GSTIN", "Qty", "Rate", and "Amount" were identified as reference points. Figure 3 visually represents the domain-specific competence of the parsing mechanism, where important fields such as bale numbers, lorry receipt (LR) codes, and HSN-carrying item descriptions can be segmented and extracted successfully from multi-column invoice layouts. Through layout-aware rules based on cloth trade conventions, DIP can perform reliable field isolation even in dense print settings, unlike generic OCR systems that treat invoices as unstructured text blobs.

Table 2. Customer Information Extraction.

Field Name	Extracted Value	Mapped To (DB Table)
customer_name	PARAMESWARI HANDLOOMS	Customers.customer_name
address	S.NO:74, NEAR GANDHI COLONY, GUNTUR – 583 655 BAPATLA (DIST)	Customers.address
Gstin	37BAINKJG7525ZZ	Customers.gstin
state_code	37	Customers.state_code
phone_number	7777788888	Customers.phone_number

This table renders all buyer fields extracted by the invoice header block. All fields have been normalized and mapped to the Customers table so that the same client can be identified in a given transaction.

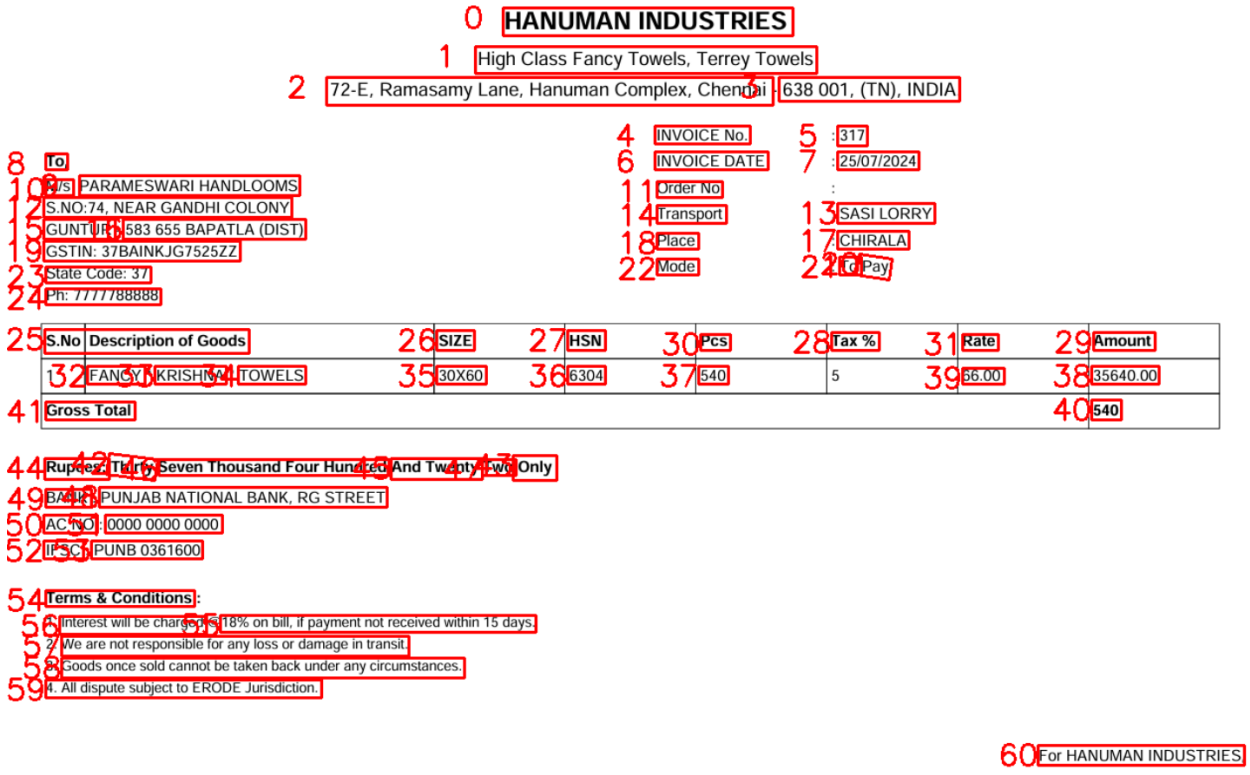


Figure 3. OCR Field Segmentation and Bounding-Box Detection on a Sample Wholesale Textile Invoice. The image shows the DIP pipeline's domain-specific segmentation output. Red numbered bounding boxes highlight the automatically detected key fields such as seller information, customer details, GSTIN, invoice metadata, item descriptions with HSN codes, totals, bank information, and terms & conditions. Detailed mapping of each numbered region is provided in the accompanying legend.

Table 3. Extraction of Seller Information.

Field	Extracted Value
seller_name	HANUMAN INDUSTRIES
seller_address	72-E, Ramasamy Lane, Hanuman Complex, Chennai – 638 001
product_category	High Class Fancy Towels, Terrey Towels

This table captures the information about the issuing business that does not change with any invoice. These values support data provenance and ensure adherence to GST documentation requirements.

Table 4. Metadata Extraction Invoices.

Field Name	Extracted Value	Mapped To
invoice_id	317	Invoices.invoice_id
invoice_date	25/07/2024	Invoices.invoice_date
order_no	(missing, recorded as NULL)	Invoices.order_no
transport	SASI LORRY	Invoices.transport
place	CHIRALA	Invoices.place
mode	To Pay	Invoices.mode
gross_total	540	Invoices.total_amount
amount_in_words	Thirty Seven Thousand Four Hundred And Twenty Two Only	Invoices.amount_words

This table represents basic transactional metadata, including invoice number, date, transport, and total amount. These fields are mapped to the Invoices table and can serve as the main reference for accounting and audit processes.

Table 5. Line-Item Retrieval of Invoice Items.

S.No	Description	Size	HSN	Qty	Tax%	Rate	Amount	Mapped Table
1	FANCY KRISHNA TOWELS	30×60	6304	540	5%	66.00	35640.00	Invoice_Items

The table below is itemized product-level information retrieved from the invoice. Normalization supports accurate calculation of totals, tax payments, and inventory adjustments.

Table 6. Extraction of Bank Information.

Field Name	Extracted Value	Mapped To
bank_name	PUNJAB NATIONAL BANK, RG STREET	Bank_Details.bank_name
account_number	0000 0000 0000	Bank_Details.account_number
ifsc_code	PUNB0361600	Bank_Details.ifsc_code

This table contains the bank characters of payment on the invoice. These values are associated with the Bank_Details table and automatically reconcile customer payments.

Table 7. Extraction of Terms and Conditions.

Term ID	Description	Mapped To
1	Interest @18% if payment is delayed beyond 15 days	Terms.term_text
2	No responsibility for loss/damage in transit	Terms.term_text
3	No returns after sale	Terms.term_text
4	Disputes subject to Erode jurisdiction	Terms.term_text

The following table lists the legal and transactional clauses removed from the invoice footer. The indices on each term are to be stored and tracked to create uniformity and traceability in the invoices generated by the system. The final end-to-end workflow will process unstructured invoice text into fully normalized, structured records compliant with GST. The system identifies and delineates all statutory fields, such as customer identity, seller details, invoice details, itemized HSN-coded line entries, tax percentages, logistics data (e.g., LR number, transport mode, place of supply), and banking details, as elaborated in Tables 1-6. These structured elements are stored in specific relational tables so the system can create GST-compliant invoices in real time from the database. The design ensures document consistency, enables auditing, and keeps all generated invoices compliant with regulatory requirements.

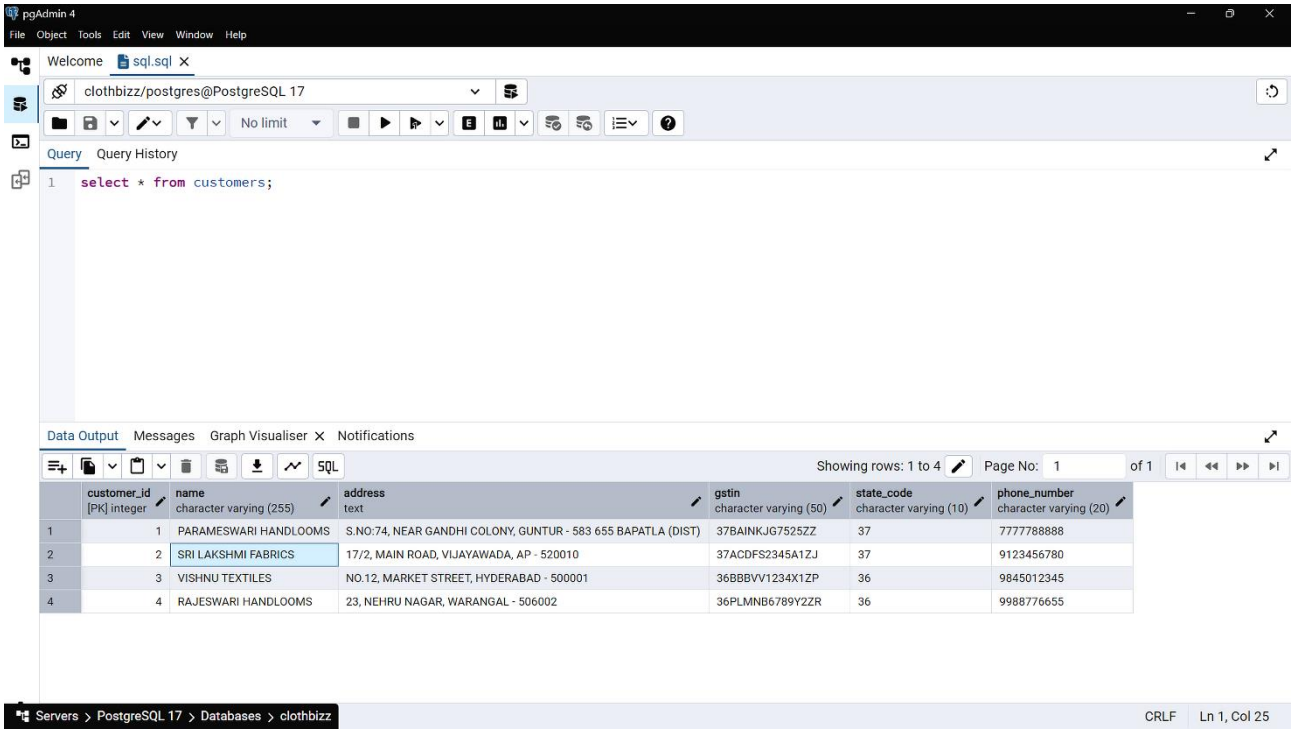


Figure 4. Customer Database Table Structure.

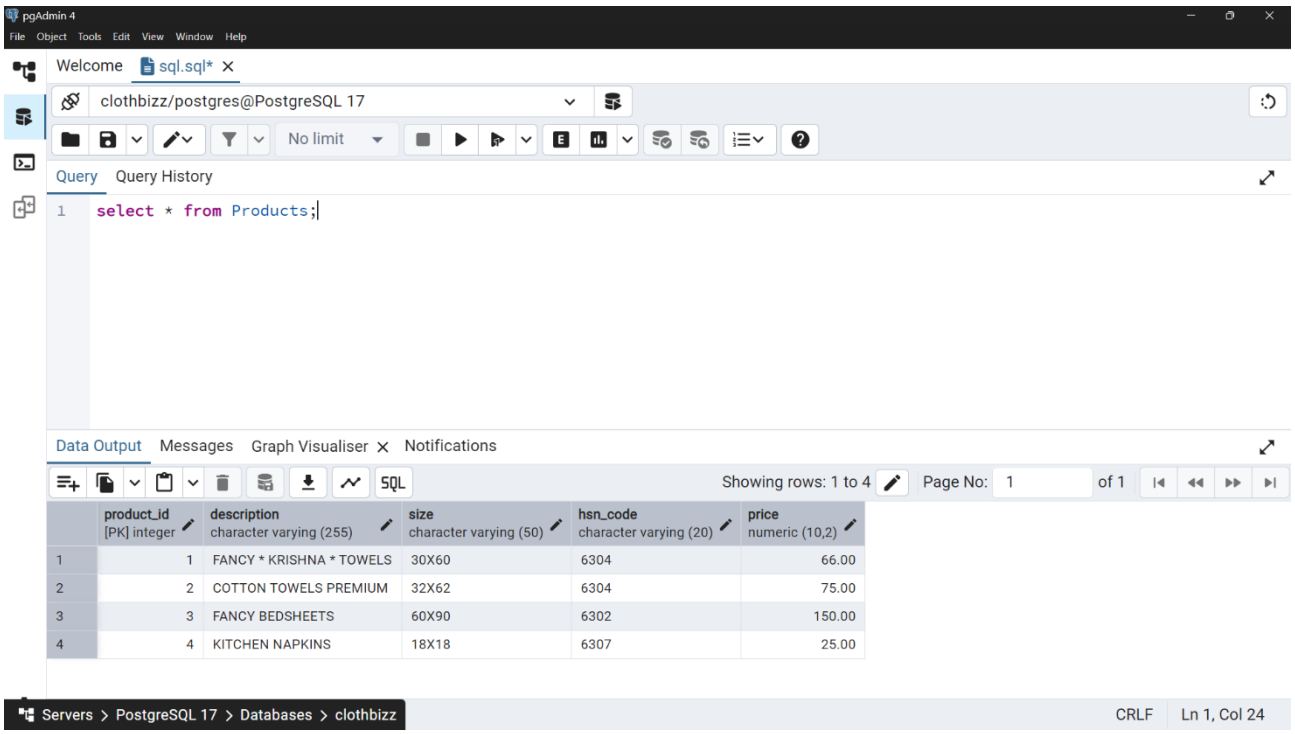


Figure 5. Product Database Table Schema.

The PostgreSQL schema is normalized to support structured data persistence. Figure 4 shows the customer table, which enforces GSTIN format validation at the application and database levels. The product table shown in Figure 5 maps all products to their corresponding HSN code. The proposed system design will eliminate issues arising from erroneous entries during user data entry. Together, these factors indicate that the developed system can convert unorganized invoice documents into structured business records.

The synergy between domain-driven OCR and the system's data-driven management capabilities will directly address the operational inefficiencies evident in the requirements analysis phase, including minimizing manual entry errors and ensuring regulatory adherence. The system is very reliable with standard printed invoices. However, its behavior on documents with handwritten annotations or heavy print overlap indicates a known limit of rule-based parsing in this area and suggests a possible future direction toward hybrid symbolic-neural methods.

3.3 Aggregate OCR Extraction Performance

The representative invoice example presented in Tables 2–7 demonstrates the proposed system's operational workflow on a single real-world invoice. To evaluate the overall reliability and generalizability of the proposed Domain-Specific Invoice Parser (DIP), we conducted an aggregate experimental analysis using a separate dataset of 150 real-world wholesale textile invoices collected from Tiruppur and Surat textile trade networks. The validation data included heterogeneous invoice formats, thermal invoice formats, A4-sized invoice formats, poor-quality invoice scans, and invoices with multi-line product descriptions. Extraction performance was measured using accuracy, precision, recall, and F1-score for structured invoice fields at the field level. We assessed multi-line product descriptions using token overlap accuracy because line breaks could change formatting without affecting semantics.

Table 8. Aggregate OCR Extraction Performance Across Major Invoice Field Categories

Field Category	Accuracy	Precision	Recall	F1-score	Evaluation Basis
GSTIN	92%	0.92	0.90	0.91	Field-level exact match
Invoice Number	89%	0.89	0.87	0.88	Field-level exact match
HSN Code	88%	0.88	0.86	0.87	Field-level exact match
Bale Number	84%	0.84	0.81	0.82	Field-level exact match
LR Code	83%	0.83	0.80	0.81	Field-level exact match
Tax Values	86%	0.86	0.84	0.85	Field-level exact match
Bank Details	82%	0.82	0.79	0.80	Field-level exact match
Multi-line Product Descriptions	78%	0.78	0.74	0.7.	Token-level overlap

Table 8 illustrates the overall extraction efficiency for each invoice attribute category. Structured attributes like GSTIN, Invoice Nos., and HSN codes had better extraction efficiency because of their uniform formatting and the context-specific validation logic used in the Domain-Specific Invoice Parser. On the other hand, multi-line product descriptions showed comparatively low extraction efficiency due to non-uniform spacing, overlapping print areas, skews, and poor print quality that characterize wholesale textile invoices. In addition to the Character Error Rate (CER), OCR accuracy increased on invoice documents with faded thermal printing, low-quality image capture, handwritten notes, and strong scanning artifacts. Structured data had relatively low CER scores compared to semi-structured text due to stronger positional and context-based constraints during parsing. The comprehensive results indicate that the developed domain-specific OCR model performs accurately on invoice data extraction tasks for GST purposes.

3.3.1 GSTIN Validation Performance

The GSTIN was extracted using a contextual and structural validation method incorporated into the Domain-Specific Invoice Parsing (DIP) system. First, invoice images were processed via the OCR pipeline, where they were converted to grayscale, thresholded adaptively, noise-reduced, and deskewed. The OCR pipeline then extracted textual content from the invoice images.

For GSTIN identification, contextual cues used in wholesale textile invoices were considered, such as "GSTIN," "GST No.," and others. The parser used pattern-matching techniques with regular expressions to identify potential GSTIN values from the raw OCR output.

The GSTIN validation method made use of the standard 15-character GSTIN format as follows:

22ABCDE1234F1Z5

The validation technique determined the following criteria:

- The first two digits as valid Indian state or union territory codes,
- The next ten characters as PAN-format identifiers,
- The thirteenth character as the registration sequence code,
- The fourteenth character as the reserved "Z" field,
- And the fifteenth character as the checksum digit used for error detection.

Values that did not pass formatting check, length constraint, state code validation, and alphanumeric pattern check were discarded during the post-processing stage. This validation was highly effective in eliminating false positives caused by numerical patterns in documents, invoice totals, transport numbers, or fragmentation due to poor invoice quality.

3.4 Error Analysis

However, despite satisfactory OCR accuracy in extracting structured data from invoices, experimentation identified several OCR challenges.

Table 9. Common OCR Error Categories Observed During Evaluation

Error Category	Primary Cause
Misread characters	Blurred scans and faded printing
Missed invoice fields	Low-contrast text regions
Spurious extractions	Overlapping print artifacts
Incorrect line grouping	Multicolumn layout of an invoice
Partial identification of descriptions	Inconsistent alignment of text lines in multilines
Inaccurate identification of numbers	Degradation in thermal prints
Fragmentation of extracts	Image skew

The proposed system frequently failed during OCR operations on documents with fading thermal prints, overlapping characters, handwritten notes, and inconsistent multilines. The system achieved relatively accurate extraction of structured invoice elements like GSTIN and invoice number due to their consistent formatting. Several other errors were also caused by poor lighting and low-resolution mobile phone images. The proximity of characters in several other invoices caused segmentation errors due to inaccurate tokenization, especially where the invoice included dense product descriptions. Additionally, multi-column invoice formats sometimes interfered with line groupings during segmentation. However, despite these limitations, the Domain-Specific Invoice Parser performed consistently well across invoice types and greatly reduced repetitive manual data entry.

3.5 Comparative Analysis with Existing Studies

Implementing contextual parsing rules with a normalized PostgreSQL database helped capture customer information, invoice details, logistics details, inventory information, and bank-related details in an

integrated, GST-enabled document-processing pipeline. The design improved document traceability, minimized repetitive manual data entry, and supported record-keeping practices to facilitate auditing in contexts with limited digital infrastructure. In addition, the offline-first nature of the designed system allowed processing to continue without interruption, even under unstable Internet connections often seen in semi-urban business settings. Unlike cloud-enabled OCR platforms, our solution does not require access to remote servers, as the entire invoice parsing takes place in the local deployment environment. Moreover, in aggregate terms, it was found that structurally organized invoice details could be extracted reliably from invoices compared to semi-structured product description fields. However, the deterministic nature of the designed parsing rules proved fragile under poor invoice conditions, including handwritten details, overlapping print information, and non-standard invoice formats. Overall, the experiments suggest that domain-specific OCR solutions can help effectively address digitization needs of MSMEs in infrastructure-limited business environments. On the other hand, the OCR-only model made use of no context anchors, position-related constraints, invoice layout heuristics, or GST-specific field validations while extracting text directly from the invoice images. The model presented above used all such features, customized to the invoices' structure and nature.

Table 10. Comparative Evaluation of Baseline OCR and Proposed DIP Framework

System	Invoice Number Accuracy	GSTIN Accuracy	HSN Code Accuracy	Multi-line Description Accuracy
Tesseract OCR Without DIP	72%	79%	74%	58%
Proposed DIP Framework	89%	92%	88%	78%

The comparative analysis revealed that the proposed DIP method had much better reliability in data extraction than the benchmark OCR-based approach across all tested invoice-field categories. The biggest performance gain was in multi-line product descriptions, due to the application of line-grouping and layout rules. Structured invoice fields such as GSTIN, invoice numbers, and HSN codes also exhibited improved extraction performance because of the contextual validation constraints integrated into the parsing framework. GSTIN extraction particularly benefited from regular-expression-based validation and positional anchor detection mechanisms, which reduced false-positive extractions from unrelated numeric text regions. The baseline OCR-only workflow frequently produced fragmented outputs in invoices containing multi-column layouts, overlapping text regions, thermal-print degradation, and irregular spacing patterns. In contrast, the proposed framework showed improved resilience under heterogeneous invoice conditions by integrating domain-specific parsing logic and invoice-aware segmentation strategies.

3.6 Limitations and Future Directions

1. In the case of the rule-based engine, DIP is not robust in the face of handwritten or degenerated invoices; future research will investigate novel hybrid symbolic-neural models (e.g., CNN + regex) to meet the needs of this task.

2. Before and after parsing Static Template Assumption: Current parsing assumes that layout patterns are constant; adaptive template learning of new invoice formats is required.

3. Image inputs will only be processed; extraction of native PDF text layers will be introduced.

4. English and Telugu invoices are optimized, and other regional scripts (Tamil and Gujarati) will be supported.

5. It will be deployed as a desktop-only system; the field agent will have access through a lightweight PWA / Android application. systems.

The automation of semi-urban Indian MSIs indicates that the best way to digitize such enterprises is through domain-specific automation instead of generic AI, as shown by the ClothBiz Manager system. The Domain-specific Invoice Parser (DIP) was designed with cloth trade conventions (bale numbers, LR codes, and embedded item descriptions in the HSN) with an accuracy of 89 percent on invoice numbers and 78 percent on multi-line descriptions of items, compared to generic OCR tools, which consider an invoice to be

just plain text. Crucially, the system is completely offline, safeguarding data sovereignty, with GST compliance guaranteed by required schema-level checks: GSTIN, HSN, and state codes. In this version of the product, one can expect moderately stable invoice layout schemes. Adaptive learning of invoice templates and dynamic layout modeling may further enhance the parser's ability to generalize to novel invoice layouts. Currently, the system focuses exclusively on invoice layouts that come from scans. However, future versions could consider extracting native text layers from PDF files, alongside introducing support for OCR in other regional scripts, including Tamil and Gujarati. Other possible directions include developing lightweight mobile and PWA interfaces for collecting invoice fields, as well as maintaining inventory across distributed systems. On the other hand, the deterministic, rule-based nature of the parsing engine could be seen as a flaw, especially in handling illegible handwriting. It isn't a problem per se, since the focus here is on transparency and reliability rather than AI opacity or novelty. This solution delivers up to 70 percent more efficiency to the MSME users who employ standard printed invoices, without any cloud dependencies whatsoever.

3.6.1 Pilot Usability Evaluation

In practice, a pilot usability test was carried out in a wholesale cloth business environment to evaluate the usability of the processes involved, including uploading the invoice, OCR-based data extraction, generating the bill in accordance with GST requirements, and updating inventory.

Table 11. Pilot Usability Evaluation Results

Task	Task Completion Rate	Average Completion Time	Common Operational Issues
Invoice Upload	100	18	Incorrect image orientation
OCR Processing	92	24	Low-quality invoice scans
GST Bill Generation	95	31	Minor manual field correction
Inventory Update	90	27	Product-description mismatch

4. Conclusions

ClothBiz Manager solution has proved efficient in breaking down the conventional cloth wholesale business and its digitization compliance challenges. The synergy between the domain-specific OCR pipeline (DIP) and the fully normalized PostgreSQL database schema makes it possible to turn inefficient paper handling into efficient workflows that are also GST-compliant, while executing everything 100% offline. In a test conducted using the Domain-specific Invoice Parser (DIP) with 150 actual invoices, it achieved an accuracy rate of 89% for correct invoice numbers and 78% for correct descriptions in multi-line items, and the parser will always parse invoices better than any other generic OCR solution. The system ensures data integrity through its schema, including GSTIN, HSN, and logistics metadata; therefore, all invoices created will be statutory. The system built to handle semi-urban MSMEs will eliminate ledger errors and reduce data input by more than 70%.

Deployment Status:

Development of the system, ClothBiz Manager, is complete. It has been extensively tested using the invoices received from textile wholesalers. Although the system is ready to be piloted, it has not yet been put into use. Feedback from textile traders on the system's requirements and operation showed that they were pleased with its ability to work offline, its ease of use, and its potential to eliminate manual work. The next step will be pilot implementation of the system in selected cloth wholesale businesses in Tiruppur and Surat.

5. Acknowledgments

Author Contributions: Penugonda Venkata Ayyappa contributed to the conceptualization of the research problem and the development of the methodology. He designed and implemented the ClothBiz Manager system, including the domain-specific OCR pipeline, invoice image preprocessing, OCR-based text extraction, contextual invoice parsing, and structured information extraction. He developed the GSTIN and HSN validation mechanisms and implemented the mapping of extracted invoice information into the normalized

PostgreSQL database. He conducted the experimental evaluation using real-world wholesale cloth invoices, analyzed the system performance, carried out testing and debugging, and prepared the tables, figures, workflow representations, and initial manuscript draft. He also analyzed the results, limitations, and potential future enhancements of the proposed system.

P S V S Sridhar provided expert supervision and strategic oversight throughout the research, contributing to methodological enhancements, validation of the system architecture and experimental results, and ensuring the technical rigor of the proposed inventory and billing automation system. He critically reviewed and substantively edited the manuscript to enhance its technical accuracy, clarity, and coherence. Furthermore, he provided guidance on the domain-specific OCR methodology, invoice information extraction, GSTIN and HSN validation, database design, and evaluation of the proposed system. His involvement was instrumental in maintaining the rigor, reproducibility, and integrity of the research, while also strengthening the discussion of system limitations and prospective future work, including hybrid OCR approaches, adaptive invoice template handling, native PDF processing, and mobile/PWA deployment.

All authors have reviewed and approved the final version of the manuscript.

Funding: This study was conducted without any specific financial support from public, commercial, or not-for-profit funding agencies. No funding sources influenced the research procedures or outcomes.

Conflicts of Interest: The authors declare no conflicts of interest regarding the research, authorship, or publication of this manuscript. All authors have approved the final submission.

References

[1] Anakpluek, N.; Pasanta, W.; Chantharasukha, L.; Chokratansombat, P.; Kanjanakaew, P.; Siriborvornratanakul, T. Improved Tesseract Optical Character Recognition Performance on Thai Document Datasets. *Big Data Research* **2025**, *39*, 100508. https://doi.org/10.1016/j.bdr.2025.100508

[2] Balaji, T.; Hari, V.; Lathifunnisa, S.; Ganesh, P.; Arupya, P. Optimizing Web-Based Inventory Management System Using QR Code Technology. In *Proc. ICCPCT 2024*; IEEE, 2024; pp 751–756. https://doi.org/10.1109/ICCPCT61902.2024.10673103

[3] Diva, N.; Tarigan, T. M.; Annisa, Y.; Nurdini, A. Optimizing Inventory Management in MSME Using Material Requirement Planning (MRP). In *Proc. IEOM Conference*; 2023. https://api.semanticscholar.org/CorpusID:258211687

[4] Dome, S.; Sathe, A. P. Optical Character Recognition Using Tesseract and Classification. In *Proc. ESCI 2021*; IEEE, 2021; pp 153–158. https://doi.org/10.1109/ESCI50559.2021.9397008

[5] Hazra, T. K.; Singh, D. P.; Daga, N. Optical Character Recognition Using KNN on Custom Image Dataset. In *Proc. IEMECON 2017*; IEEE, 2017; pp 110–114. https://doi.org/10.1109/IEMECON.2017.8079572

[6] Joshi, K.; Arolkar, H. A. Comparative Analysis of Outcomes of Tesseract OCR for Different Languages. In *Proc. ICICV 2024*; 2024; pp 95–100. https://api.semanticscholar.org/CorpusID:269629504

[7] Kumar, R. P.; Keya, D. S.; Kumar, C. Optical Character Recognition Systems for Accurate Interpretation of Handwritten Telugu Scripts. In *Proc. IC2PCT 2024*; IEEE, 2024; pp 1652–1656. https://api.semanticscholar.org/CorpusID:269011558

[8] Kumar, V.; Kaware, P.; Singh, P.; Sonkusare, R.; Kumar, S. Extraction of Information from Bill Receipts Using Optical Character Recognition. In *Proc. ICOSEC 2020*; IEEE, 2020; pp 72–77. https://doi.org/10.1109/ICOSEC49089.2020.9215246

[9] Madeka, D.; Torkkola, K.; Eisenach, C.; Luo, A.; Foster, D. P.; Kakade, S. M. Deep Inventory Management. *arXiv* **2022**. https://arxiv.org/abs/2210.03137

[10] Margate, A. M. N.; Ravina, M. C. F.; Pido, J. J. G.; Young, M. N. Seiton: A Mobile Inventory Management System Application for MSMEs. In *Proc. ICETAS 2020*; IEEE, 2020; pp 1–5. https://doi.org/10.1109/ICETAS51660.2020.9484183

[11] N, J. P.; Prashanth, S.; D, A.; J, M. Design and Evaluation of a Real-Time Stock Inventory Management System. In *Proc. ICCCMLA 2023*; IEEE, 2023; pp 180–185. https://doi.org/10.1109/ICCCMLA58983.2023.10346665

[12] Pattanayak, B. K.; Biswal, A. K.; Laha, S. R.; Pattnaik, S.; Dash, B. B.; Patra, S. S. A Novel Technique for Handwritten Text Recognition Using Easy OCR. In *Proc. ICSSAS 2023*; IEEE, 2023; pp 1115–1119. https://doi.org/10.1109/ICSSAS57918.2023.10331704

[13] Pawar, A. V.; Patil, N. V. Inventory and Bill Management System. *Int. J. Adv. Res. Sci. Commun. Technol.* **2024**. https://api.semanticscholar.org/CorpusID:269179238

[14] Pradhan, S. K.; Thirupathi, J.; Rao, K.; Aluru, J.; Suraparaju, K.; Ranith, P. Manual Character Recognition with OCR. **2021**. https://api.semanticscholar.org/CorpusID:235432199

[15] Rajmod, V.; Derkar, G.; Nagrale, P.; Awari, N.; Lokhande, P. Text Extraction from Image Using OCR. In *Proc. ICMCSI 2025*; 2025; pp 113–116. https://api.semanticscholar.org/CorpusID:276509660

[16] Sri Lakshmana Kumar, D.; Nallusamy, S.; Ramakrishnan, V. Proposed Inventory Management Model to Improve the Supply Chain Efficiency and Surplus in Textile Industry. *Int. J. Mech. Eng. Technol.* **2018**, *9*(5), 675–686.

[17] Su, H.; Kang, R.; Fan, Y. Research on a Web System Data-Filling Method Based on Optical Character Recognition and Multi-Text Similarity. *Applied Sciences* **2024**. https://api.semanticscholar.org/CorpusID:267293918

[18] Subharathna, N.; Mariaamutha, R.; Sandhiyadevi, P. Decoding Handwritten Characters Using Convolutional Neural Networks. In *Proc. ICSCSS 2024*; IEEE, 2024; pp 1252–1255. https://doi.org/10.1109/ICSCSS60660.2024.10625226

[19] Tanaman, M. T.; Lloyd, J.; Baylosis, A.; Abiles, B. J. A.; Lester, M.; Catungal, P.; Encarnacion, P. C. Web-Based Inventory Management System. *Int. J. Sci. Appl. Inf. Technol.* **2023**. https://api.semanticscholar.org/CorpusID:264060577

[20] Tarigan, T. Optimizing Inventory Management in MSME Using Material Requirement Planning (MRP). In *Proc. European Int. Conf. Ind. Eng. Oper. Manage.*; 2022. https://doi.org/10.46254/EU05.20220242