AI Model Project

Bank Check Verification

Presented by:

ABBES Imen

BENSALAH Hatem

CHAMAKH Med Malek

CHABCHOUB Amina

ABDELMOULA Yassine

BENHAMIDA Fadhel

April 18, 2025

Contents

1 Introduction

Bank checks, though increasingly complemented by digital payment systems, remain integral to many financial ecosystems, particularly in business-to-business transactions. Manual check verification, while traditional, is time-consuming and error-prone, leading to security vulnerabilities and operational inefficiencies. This project addresses these challenges through a comprehensive artificial intelligence (AI) solution that integrates computer vision, machine learning, and natural language processing (NLP) to enable automatic bank check verification. Our aim is to provide a system that is both intelligent and practical, capable of enhancing trust, reducing fraud, and optimizing workflow in banking operations.

2 Project Objectives

2.1 Business Objectives

This project directly addresses banking institutions' operational pain points by focusing on:

- Streamlining and automating the check validation pipeline.
- Minimizing fraud by detecting inconsistencies or tampering.
- Reducing dependency on manual inspection, thus improving throughput.
- Boosting customer confidence with faster turnaround and fewer rejections.

2.2 Technical Objectives

To support the business goals, our system was designed to:

- Implement an advanced OCR engine optimized for both printed and handwritten data.
- Build a deep learning model to validate the authenticity of the check's signature.
- Apply NLP techniques to compare written and numeric check amounts.
- Design a modular and user-friendly frontend for non-technical bank employees.
- Ensure scalability and integration capabilities with existing bank infrastructure.

3 Business Understanding

Check rejection often stems from minor discrepancies that could be easily resolved or prevented. These include:

- Handwriting errors or illegibility
- Amount discrepancies between text and digits
- Unverified or forged signatures
- Incomplete or incorrect date formatting

By digitizing and automating these validations, banks can significantly reduce error rates, handle larger volumes, and preserve a high standard of security and compliance.

4 Data Understanding

Our raw dataset consisted of over 3,000 scanned checks collected from anonymized sources. Each image includes several important regions:

- **Payer Information:** Full name and sometimes address
- **Amount (Numeric):** Typically printed in the right middle of the check
- **Amount (Written):** Positioned below the numeric field
- **Date:** Top-right, handwritten or printed

- **Signature:** Bottom-right, always handwritten
- **Check Number & Bank Logo:** Typically top-left and bottom

Initial exploration showed wide variability in layout, style, and scan quality, which influenced our preprocessing strategy.

5 Data Quality Analysis

5.1 Identified Issues

- Low-resolution images leading to unreadable fields
- Skewed or rotated check orientations
- Handwriting variability across different users and languages
- Incomplete datasets with missing labels in ground truth

5.2 Proposed Solutions

- Implement adaptive thresholding and morphological operations for clearer image segmentation.
- Train models on augmented datasets to improve generalization to new handwriting styles.
- Leverage layout detection to realign and deskew images before processing.
- Use synthetic data generation to supplement rare check formats and handwriting samples.

6 System Architecture and Implementation

6.1 Overall Architecture

The system consists of three major modules:

1. **Preprocessing Layer:** Enhances image quality and normalizes formats.
2. **Analysis Layer:** Applies OCR and verification logic.
3. **Presentation Layer:** Interacts with users through an intuitive web interface.

6.2 Region of Interest Detection

We adopted YOLOv5 for object detection to isolate important check fields dynamically, even when layouts vary. This ensures accurate segmentation of:

- Signature box
- Amount (numeric and text)
- Date field
- Payer name

6.3 Image Preprocessing Pipeline

Each image goes through:

1. Conversion to grayscale
2. Contrast enhancement using CLAHE
3. Binarization via adaptive Gaussian threshold
4. Deskewing using Hough Transform
5. Cropping using bounding box detection

6.4 OCR Extraction Workflow

Tesseract OCR (fine-tuned with custom models) handles text recognition. Extracted fields are annotated and stored in JSON format for post-processing. Signature zones are stored as cropped image patches for neural verification.

7 Intelligent Data Correction using AI Models

7.1 Text Vectorization using TF-IDF Embedding

Amount-in-words fields are vectorized using TF-IDF to build similarity maps. This allows us to:

- Detect spelling inconsistencies in OCR output
- Compare against a reference dictionary of amount expressions
- Cluster similar phrases to correct recurring OCR errors

7.2 Cosine Similarity for OCR Correction

Using vector embeddings, we calculate cosine similarity scores between the OCR output and a curated dataset of valid entries. A score threshold determines whether automatic correction is applied.

7.3 Validation Strategy and Thresholding

If the similarity exceeds a confidence threshold (85%), corrections are made automatically. Otherwise, human intervention is requested. This hybrid model ensures high reliability with minimal oversight.

8 User Interface Design

8.1 Gradio Integration

Gradio is used for its rapid prototyping and compatibility with ML models. Our interface allows:

- Drag-and-drop check uploads
- Real-time display of extracted data
- Visual overlays on image fields for verification
- Export of final results as structured JSON or PDF reports

8.2 User Flow and Experience

1. Upload or drag a check image into the UI.
2. The backend processes and displays extracted data.
3. The system flags any suspected errors.
4. User can approve or override suggestions.
5. The final validated version is archived.

9 Results and Evaluation

9.1 Performance Metrics

The final system was tested on a labeled validation set:

- **OCR Accuracy:** 92% on printed text, 86% on handwritten fields
- **Signature Matching Accuracy:** 88%
- **Correction Success Rate:** 90%

- **Overall Verification Accuracy:** 93%
- **User Satisfaction in Pilot Testing:** 4.6/5 average rating

9.2 Error Analysis

Most errors came from highly stylized handwriting or poor-quality scans. Further training with larger and more diverse datasets could mitigate these cases.

10 Conclusion and Future Work

We developed an AI-powered check verification system that significantly enhances banking operations. The system is modular, scalable, and adaptable to different bank formats. Future extensions include:

- **Multi-language and script support** for international usage.
- **Deep learning handwriting models** using LSTM-CNN hybrids.
- **Expansion to other document types**, such as payment slips, utility bills, or contracts.
- **Blockchain-based verification ledger** to prevent tampering or fraud post-processing.