AI/ML Developer Assessment Task - OCR Text Extraction

Task Overview

You are required to develop an OCR-based text extraction system that processes shipping label/waybill images and extracts specific information with high accuracy. This assessment evaluates your ability to implement computer vision solutions, handle real-world document processing challenges, and deliver production-ready code.

Problem Statement

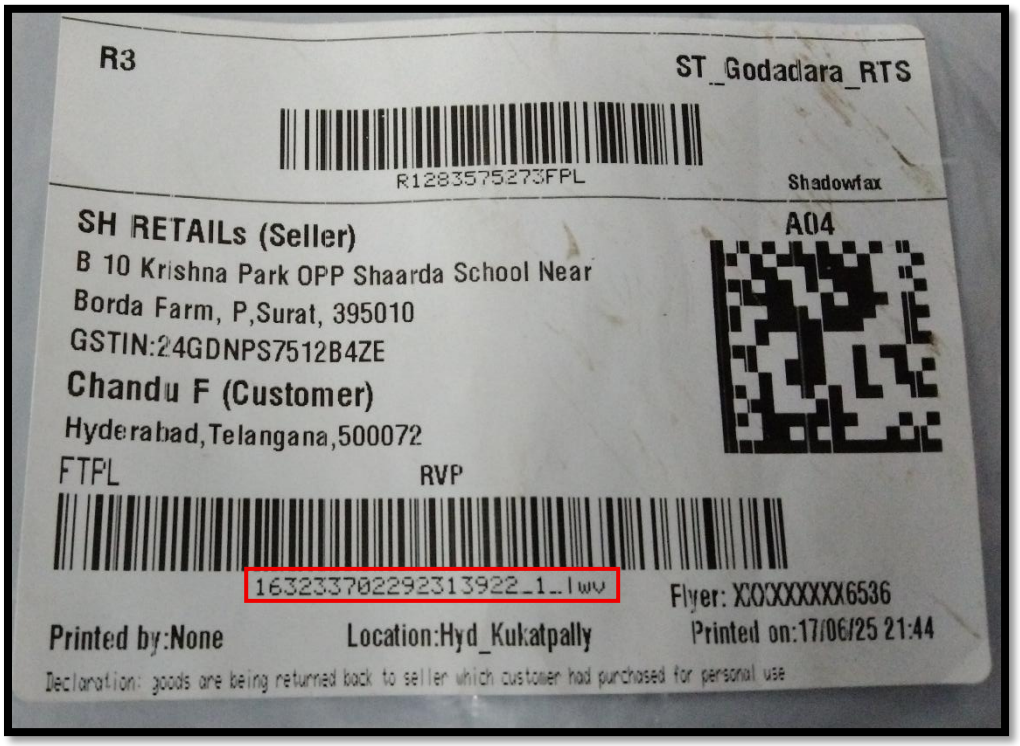
Objective

Build an automated system that:

1. Performs OCR on shipping label/waybill images
2. Extracts the complete text line containing the pattern "*_1_*"
3. Achieves **minimum 75% accuracy** on the test dataset

Dataset

- **Location**: (Click to get Test Dataset)
- **Content**: Shipping labels/waybill images containing various text fields
- **Target**: Text lines containing "*1*" pattern (accuracy will be evaluated solely on this extraction)



Example Scenario

The dataset contains test images of shipping labels, and the task is to extract the specific information listed below.
Your objective is to extract this specific type of text from the images. The text may appear in different positions across the images.

Ultimately, the expected output format should be: "**1632337022923139 22_1_lWV**".

Notes:

The candidate's primary task is to develop or select the best OCR model and implement a robust extraction pipeline that can accurately extract text from the images, specifically the entire line containing the pattern _1_. Since the dataset includes images where characters are partially erased or degraded, the solution must be able to correctly identify and extract each character in that target line. Accuracy will be evaluated solely based on the correctness of this extracted target text.

As part of the project submission, the candidate must provide:

- OCR output screenshots for every test image
- Output results in JSON format for every test image

Technical Requirements

Core Constraints

- **Allowed**: Any open-source LLMs, pre-trained models, OCR engines even you can train your own model.
- **Not Allowed**: Commercial APIs (Google Vision, AWS Textract, Azure OCR, etc.)
- **Programming Language**: Python preferred
- **Accuracy Requirement**: ≥75% on target text extraction

Implementation Guidelines

Backend Requirements

- Implement robust image preprocessing pipeline
- Handle various image qualities and orientations
- Design modular, reusable code architecture
- Include error handling and edge case management
- Optimize for accuracy over speed
- Document your approach and methodology

Frontend Requirements

- Create a basic Streamlit application for demonstration
- Include:
  - Image upload functionality
  - OCR processing trigger
  - Display extracted text with highlighted target line
  - Show confidence scores if applicable

Deliverables

1. GitHub Repository

Your repository must include:

```
project-root/
├── README.md                 # Comprehensive documentation
├── requirements.txt          # All dependencies
├── src/                      # Source code
│   ├── ocr_engine.py        # Core OCR logic
│   ├── preprocessing.py     # Image preprocessing
│   ├── text_extraction.py   # Target text extraction
│   └── utils.py             # Utility functions
├── app.py                    # Streamlit application
├── tests/                    # Test cases
├── notebooks/                # Jupyter notebooks (if used for experimentation)
└── results/                  # Sample outputs and accuracy metrics
```

2. Documentation (README.md)

Must include: - **Project Overview**: Brief description of your approach - **Installation Instructions**: Step-by-step setup guide - **Usage Guide**: How to run the application - **Technical Approach**: - OCR method/model used - Preprocessing techniques - Text extraction logic - Accuracy calculation methodology - **Performance Metrics**: Accuracy achieved on test set - **Challenges & Solutions**: Problems encountered and how you solved them - **Future Improvements**: Potential enhancements

3. Accuracy Report

- Provide accuracy metrics specifically for "*1*" line extraction
- Include confusion matrix or error analysis if applicable
- Document test methodology

Evaluation Criteria

Primary (80% weightage)

- **Accuracy**: Meeting or exceeding 75% threshold
- **Code Quality**: Clean, modular, well-documented code
- **Algorithm Efficiency**: Optimal approach to problem-solving

Secondary (20% weightage)

- **Error Handling**: Robustness to edge cases
- **Documentation**: Clarity and completeness
- **UI/UX**: Streamlit app functionality and usability
- **Innovation**: Creative solutions or optimizations

Submission Guidelines

Format

- Submit via GitHub repository link
- Repository must be public or provide access
- Include all source code, documentation, and test results

Deadline

- **Maximum Duration**: 3 days from task assignment
- **Late Submission**: Will not be evaluated

Required Actions

1. **Upon Starting**: Send WhatsApp message to +91 63526 17754 confirming task commencement
2. **Upon Completion**: Send WhatsApp message to +91 63526 17754 with GitHub repository link

Important Notes

- **Clarifications**: If you have any doubts, feel free to clarify before starting the task
- **Communication**: Additionally, please send us a message on WhatsApp at +91 63526 17754 when you start working on the task and again once you complete it

Next Steps After Submission

Once you submit the task, our technical team will review your work. Upon successful evaluation, you will be invited for an offline interview at our office.

Final Reminders

- Ensure all deliverables are submitted exactly as instructed under the Submission Guidelines section
- Use only the allowed tools and follow the technical constraints specified in the document
- Maintain clean, modular, and lightweight code with high accuracy
- Submit via GitHub Repository Link
- Include clear README and instructions to run the project

If anything is unclear or you need further clarification, feel free to ask before starting. We're happy to guide you through the process. Contact on +916352617754 only on WhatsApp.

Good luck with your assessment! We look forward to reviewing your innovative solution.